An Internet of Things Ecosystem for Atmospheric Water Generators

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 24 March 2025, accepted 14 April 2025, date of publication 21 April 2025, date of current version 8 May 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3562742

RESEARCH ARTICLE

An Internet of Things Ecosystem for Atmospheric Water Generators

FRANCESCO GAGGINI[1], RAFAEL NATALIO FONTANA CRESPO[1], (Member, IEEE), MATTEO CALÒ[2], VINCENZO MARIA GENTILE[2], ALBERTO MACII[1], (Senior Member, IEEE), AND EDOARDO PATTI[1], (Member, IEEE)

[1]Department of Control and Computer Engineering (DAUIN), Politecnico di Torino, 10129 Turin, Italy
[2]TEBE Group, Department of Energy "G. Ferraris" (DENERG), Politecnico di Torino, 10129 Turin, Italy

Corresponding author: Vincenzo Maria Gentile (vincenzo.gentile@polito.it)

ABSTRACT In an era of diminishing water resources, Atmospheric Water Generators (AWG) offer an innovative solution to address the global water crisis by harnessing alternative local water sources. Recent advances in AWG technology have demonstrated the feasibility of converting atmospheric moisture into potable water, making AWGs suitable for use as domestic appliances and positioning them as key players in tackling the water-energy nexus. In this context, efficient data collection and management from AWG systems are critical for improving technological capabilities and performance optimization. This work presents an Internet of Things (IoT) ecosystem that integrates IoT-enabled AWG machines, utilizing a biopolymer for water generation, and an IoT platform to collect, store, and manage data from these devices. We enhance existing AWG machines with IoT functionalities, allowing for remote monitoring and control of a fleet of units. Additionally, we demonstrate the implementation of this ecosystem in a real-world scenario, enabling the management and oversight of operational AWG machines. Communication tests were conducted to evaluate transmission performance. Since AWGs may be located in remote regions, we tested the scenario in which AWGs are connected to the Internet via Low Earth Orbit Satellites Internet Service Providers (e.g. Starlink). The results showed that latency reached values around 100 ms, and less than 1 s even for the extreme scenario of total network congestion, guaranteeing that the data transmission latency is within acceptable limits for this application, ensuring effective monitoring and control of the AWG systems.

INDEX TERMS Internet of Things, atmospheric water generator, autonomous remote control, decentralized water production, IoT platform.

I. INTRODUCTION

Atmoshperic Water Generators (AWG) emerge as a promising remedy for regions facing water scarcity or logistical constraints, offering a pathway to convert local atmospheric humidity into freshwater [1], [2], [3], [4]. Many studies have demonstrated that AWG devices, which require energy input for the moisture-to-water conversion process, can enhance sustainability when powered by renewable energy sources [5], [6], [7], [8]. Given that the residential sector accounts for over 25% of water consumption, integrating AWG systems into buildings offers considerable potential for improving sustainability and operational efficiency, addressing a substantial portion of global water demand. Envisioning AWG devices as an appliance of dwellings could shift the focus from water scarcity to the energy-water nexus, necessitating the coordination of an energy-intensive fleet of AWG devices exploiting renewable energy sources, with the capability to provide demand-response flexibility in urban settings [9], [10], [11]. In this scenario, the development of a distributed software platform leveraging upon Internet of Things (IoT) technologies represents a promising solution to foster different distributed services to coordinate the water needs of dwellings, the energy consumption of AWG machines, and the local or regional production of renewable energy [12], [13], [14].

The associate editor coordinating the review of this manuscript and approving it for publication was Liang-Bi Chen.

Having a set of machines that continuously produce data can bring several challenges that need to be addressed. On one hand, each machine has numerous sensors and actuators that generate a considerable amount of diverse data, such as numerous temperature and humidity sensors, or fans used to move the air in different parts of the machine that need to be turned on or off. On the other hand, remote control of these actuators is crucial for (near)-real-time management, ensuring precise control over each machine while also providing full visibility into its current state so that the operators can have a full understanding of the fleet's state. In recent years, there have been growing efforts towards enhancing AWG machines with IoT technology to enable (near)-real-time monitoring of a fleet of these machines [12], [13], [15], [16]. However, these solutions focused only on the AWG machine's IoT enhancement without considering a robust framework that enables data collection and the machine's management, as well as the challenges related to the communications between the IoT-enhanced machines and the platform. This framework must ensure scalability, which is essential for managing an expanding number of AWG machines and adapting to increased data volumes and evolving requirements, such as new types of sensors, actuators or new types of AWG machines.

Our work introduces an innovative IoT ecosystem that enhances existing AWG machines and includes a vendor-agnostic, distributed platform. This system enables the remote monitoring and control of a fleet of AWG units, addressing the limitations of machines operating independently and offline. Sensor data collection is fully automated, eliminating the need for manual input. A purpose-built distributed platform efficiently gathers, stores, and manages both metadata and telemetry data from the fleet. The platform is designed using a microservices architecture, supporting both asynchronous and synchronous communication. Each service in the platform handles specific tasks and can be created, deployed, and scaled independently, ensuring flexibility and adaptability. The platform's plug-and-play design allows for easy extension of functionalities to meet evolving requirements, while it seamlessly integrates new AWG machines, automatically supporting new data types and achieving horizontal scalability through additional microservices. In terms of actuator management and control logic, the IoT ecosystem enables AWG machines to operate based on remotely defined strategies, eliminating the need for on-site human interaction. This approach is particularly beneficial for managing a large fleet of AWG machines, offering real-time monitoring of key performance indicators. By integrating IoT technology, operators gain a comprehensive view of each machine's operational status, allowing for quick, data-driven decisions and minimizing manual intervention. This leads to streamlined operations and improved overall efficiency. Moreover, this system paves the way for community-level strategies, such as demand-response models that align machine operation with renewable energy availability or grid excess, promoting both resource efficiency and sustainability.

In addition to the platform design, we tested its implementation in a real-world scenario for monitoring and controlling AWGs. To this end, we enhance an existing AWG with IoT technology [4]. The device comprises multiple sensors for monitoring temperature and humidity both within the machine and in the outdoor air to manage operations of the device. It also integrates sensors and actuators that can track and control the current absorbed by various internal components, including the fan, water circulators, motors driving automation valves, and the energy source used for the moisture-to-water conversion. These sensors and actuators enable the device's (near)-real-time remote monitoring and control. Furthermore, we carried out realistic communication tests to guarantee the platform effectiveness and reliability for transmitting data under expected workloads within the presented IoT ecosystem. Given that AWGs may be situated in remote areas, we evaluated the scenario in which AWGs are connected to the Internet through Low Earth Orbit (LEO) satellite-based Internet Service Providers, such as Starlink. These communication tests demonstrated the robustness of the proposed system to operate in limited-connectivity scenarios (i.e., via satellite links), thereby enabling the broader deployment of AWGs in off-grid locations [17].

In a nutshell, the scientific contribution can be resumed as follows:

- Developed a comprehensive IoT ecosystem that comprises the IoT-enabled enhancement of AWG machines alongside a robust infrastructure for device management. This infrastructure facilitates the monitoring and control of AWG devices in (near)-real-time.
- The platform architecture enables the extension of its functionalities in a modular, plug-and-play fashion, promoting future integration of advanced services, like automated control of IoT-enabled AWG actuators and machine learning models that process and analyse collected data. Furthermore, the proposed IoT ecosystem allows the dynamic regulation of individual AWG machines in a fleet based on, for example, predictive meteorological data, like temperature, relative humidity and solar radiation.
- Executed realistic communication tests to validate the platform's effectiveness and appropriateness for data transmission under specific workloads within the proposed IoT ecosystem. Given that AWGs may be deployed in remote or isolated areas, we investigated the scenario where AWGs were connected to the Internet through Low Earth Orbit (LEO) satellite-based connectivity solutions, such as Starlink. These communication tests showcased the robustness of the proposed ecosystem to operate in limited-connectivity scenarios (i.e., via satellite links), thereby enabling the broader deployment of AWGs in off-grid locations.

The rest of this article is organized as follows. Section II reviews different AWG technological solutions for moisture-to-water conversion and proposed approaches for enhancing AWG machines with IoT technology. Section III first introduces the proposed distributed platform for monitoring and managing Atmospheric Water Generators. Afterward, it reports the IoT-enhanced AWG device, highlighting the control logic for efficient operation and the communication process with the platform. Section IV describes the case study and the experimental setup we employed to evaluate the proposed platform's communication performance and tests performed on the IoT-enhanced AWG device in a real environment. Section V presents the experimental results. Finally, Section VI reports closing remarks.

II. STATE OF THE ART

AWGs are emerging as a sustainable alternative to address the looming global water security challenges [6], [18]. Traditional AWG systems typically use refrigeration cycles to cool air below its dew point, condensing moisture from the atmosphere [19]. However, this method proved inefficiencies in arid environments, often leading to energy consumption levels exceeding 1 kWh per liter of water produced [2], [8]. To overcome this limitation, material scientists are developing advanced sorbent materials, including Metal-Organic Frameworks (MOFs) [20], and polymers [5], [21], [22], [23], which efficiently adsorb and desorb water vapour without the need for energy-intensive refrigeration. The most effective sorbent not only possesses high moisture uptake capacities but also exhibits efficient thermal cycling, continuously alternating between dry and hot conditions and wet and mild states [2], [8]. Striking this equilibrium is fundamental for maximizing water productivity, as well as their integration in a machine operating this cycle control [6], [8], [24]. For this reason, it has been developed a setup that integrated an innovative biopolymer, with improved properties fostering both the material capacity and the rapidity of the cycles, with reduced energy requirements compatible with the exploitation of renewable energy sources [4], [21]. The use of alginate-based polymers is not only advantageous in terms of hygroscopic properties [21], which make them particularly suitable for AWG applications, but also offers significant environmental benefits. The primary components of this biopolymer are environmentally friendly, as they do not rely on hydrocarbons derived from fossil fuels. Instead, bioderived products, such as Sodium Alginate (SA), are utilized. SA is used in small quantities as a precursor and cross-linking agent. Other studies have explored the life cycle assessment (LCA) of this material [25], [26], [27]. In particular, alginate-derived polymers have gained attention in both the market and research sectors due to the role of seaweed as a sustainable raw material for biorefinery purposes. Sodium alginate, one of the products derived from the seaweed refining process, is a key element in this approach. Notably, the purification of extracted sodium alginate is a critical step in the value chain, and it avoids the use of organic solvents, unlike conventional methods of producing hydrocarbon-based polymers. Sodium alginate microspheres are considered a more environmentally friendly alternative to conventional flocculant agents used in water clarification, with a reported greenhouse gas emissions of 0.773 kg CO_2-eq [28]. However, this technology must adapt to varying weather conditions, raising the need for control rules that can respond to future dynamics of ambient temperature and relative humidity. Key parameters such as the polymer regeneration temperature, the cycling frequency between drying and wetting stages, and the system's airflow rate all play a crucial role in optimizing performance and water productivity [21]. Moreover, enabling remote interaction with the device introduces the potential for demand-response building application, which could be highly valuable in a future where renewable energy integration in the grid goes beyond energy storage, allowing excess energy to be converted into water and alleviating water demand in buildings [29], [30]. A well-designed control strategy needs to operate at the device level with reliable input-output communication to transmit control logic tailored to specific needs. Furthermore, this strategy should coordinate at a community level, envisioning a fleet of devices working in synchronization with demand, rather than focusing solely on individual needs. With the increasing integration of IoT in appliances within the building sector and daily life [31], [32], [33], AWGs based on IoT can establish their role through a robust control system, enabling these devices to adjust dynamically to, for example, fluctuating environmental parameters and/or user requirements [34]. This system would take into account the operational requirements of the machine to ensure consistent cycling, the variability of weather conditions, the availability of excess renewable energy, and the demand for fresh water — a fundamental necessity for individuals and daily household activities. Enabling (near)-real-time monitoring and management of a fleet of AWGs is one of the main advantages IoT can provide in this scenario. However, this enhancement must be coupled with a robust framework that can collect, store, and manage the data coming from these devices.

Enhancing AWGs with IoT technology is garnering increasing attention in the research community. Dar et al. [15] introduced a sophisticated weather prediction model driven by solar energy data. This interdisciplinary initiative aimed to integrate meteorological insights with solar energy analytics, facilitating the development of a more responsive and adaptive AWG. By predicting essential weather parameters, the model sought to proactively enable AWG devices to adjust their operational strategies. The AWG employed an Arduino and an ESP32 microcontroller. Nonetheless, to the best of our knowledge, how the AWG machines are monitored and remotely controlled in terms of both sensors and communication protocols was not mentioned. In [16], the authors presented an AWG utilizing a thermoelectric Peltier module, which was employed to establish the necessary conditions for achieving water condensation temperature,

also known as the dew point. This AWG employed an ESP8266 microcontroller for control purposes, which can be connected to Internet and be controlled remotely using the Blink App [35]. Similarly, Kadam et al. [12] introduced an AWG equipped with an IoT device designed to manage the machine's operational status and power consumption. This IoT device is capable of monitoring water levels, operational modes, and On/Off states, transmitting this data to the Blink server. In [13], the authors proposed an AWG utilizing a spiral wind turbine driven by airflow to direct air into a copper condenser, where the air is dehumidified and converted into water droplets. This solution incorporated an IoT device based on a Raspberry Pi (RPi) to monitor airflow. In cases of insufficient airflow, the IoT device autonomously activates a DC motor powered by a solar panel to rotate the turbine. The RPi also tracks water availability, temperature, and location. However, the methods for transmitting and collecting this information are not detailed.

To the best of our knowledge, the existing solutions remain limited in scope. They primarily focus on AWG solutions integrating IoT technologies for monitoring and control purposes without addressing the need for a robust framework to manage and coordinate a large fleet of devices, facilitating also the communication between them and the remote software infrastructure. Moreover, solutions in [12] and [16] relied on Blink, a proprietary license software.

In contrast to existing literature, our proposed solution introduces a novel element by offering a comprehensive IoT ecosystem that encompasses both the IoT-enabled enhancement of AWG machines and a robust framework for managing these devices. Our platform facilitates (near)-real-time monitoring and control of AWGs, giving operators a bird-eye overview of the entire fleet and enabling quick and real-time adjustments to the AWGs' actuators. Additionally, the platform is designed to scale its functionalities in a plug-and-play manner horizontally. This will support the future integration of advanced services, such as automated control of IoT-enabled AWG actuators and the implementation of machine learning algorithms to analyse collected data. Moreover, the proposed IoT ecosystem enables the dynamic regulation of each AWG machine in a fleet based on, e.g. predictive meteorological data, such as temperature, relative humidity and solar radiation. Last but not least, we conducted realistic communication tests to ensure the platform's effectiveness and suitability in transmitting data under expected workloads within the proposed IoT ecosystem. Considering that AWGs may be located in remote regions, we studied the scenario in which AWGs are connected to the Internet via Low Earth Orbit (LEO) satellite-based Internet Service Providers, such as Starlink.

III. METHODOLOGY

In this section, we describe the proposed IoT ecosystem. On one hand, we present the platform for collecting and managing AWG data, the architecture of which is depicted in Fig. 1. On the other hand, we will provide an overview of the AWG we used, its IoT enhancement, control logic for efficient operation, and the communication process with the platform.

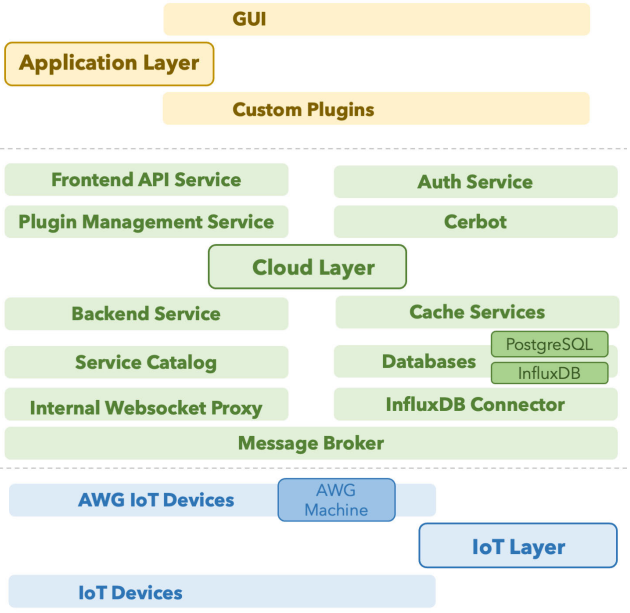


FIGURE 1. Schema of platform layered architecture.

A. PLATFORM ARCHITECTURE

We built an IoT platform based on the microservices approach that ensures: i) *flexibility* - the ability to integrate services with heterogeneous attributes and requirements; ii) *reliability* - to perform a required function under stated conditions for a stated time; iii) *modularity* - design the entire system as a set of services performing particular functions communicating through lightweight mechanisms; iv) *scalability* - the ability to handle and process a large number of users, sensors, devices, information exchanged and processed, data stored and to be able to grow during the system runtime; v) *interoperability* - the capability to integrate heterogeneous devices (e.g. different manufacturers), technologies and systems; vi) *decentralization* - the implementation of different components, each of them providing particular functionalities autonomously making use of the most appropriate technology; vii) *extendibility* - the ability to support software and permissions update, add new functionalities and bugs fix; viii) *security* - protection against unauthorized access and threats; ix) *standardization* - to provide uniform interfaces for data exchange following open data formats; x) *synchronous communication* - the possibility of browsing and managing users and devices as well as exploiting different services functionalities using the request/response approach; and xi) *asynchronous communication* - support (near)-real-time data transmission for event-based communication enabling low latency and scalability, exploiting the publish/subscribe mechanism.

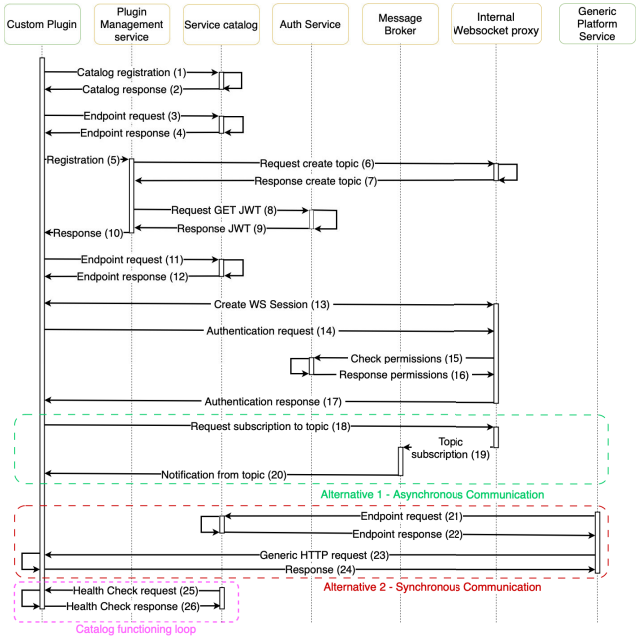


FIGURE 2. Main operational steps of the connection process of a custom plugin.

The architecture schematized in Fig. 1 obeys the microservices design pattern that involves *developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms* [36]. These microservices can be "containerised" using Docker [37], which means encapsulating individual components within portable, self-sufficient runtime environments, facilitating their deployment, orchestration and scalability. The platform supports two main communication paradigms: a synchronous approach, implemented with the Representation State Transfer (REST) [38] protocol and functioning through a *request/response* mechanism, and an asynchronous approach implemented with the Message Queuing Telemetry Transfer (MQTT) [39] protocol, based on a *publish/subscribe* mechanism [40]. The architecture's design enables the integration of novel components in a plug-and-play manner. These new components can seamlessly interact with other platform modules synchronously (via REST) and asynchronously (via MQTT).

Fig. 1 illustrates the layers of the platform architecture: i) *IoT Layer*, ii) *Core Components Layer*, and iii) *Application Layer*. The first layer, the *IoT Layer*, includes the *AWG IoT Devices*. AWG machines, by themselves, do not have IoT capabilities. A device connector is a hardware component that acts a "bridge", providing Internet connection. The *AWG IoT Device* is a device connector that enables AWG machines to connect to the Internet and communicate with the platform. Further details of the *AWG IoT Device* will be provided in Section III-B. Moreover, the platform design enables the integration of other *IoT devices* that could be considered for this use case in the future.

The *Core Component Layer* is the central core of the platform and contains several components. The *Message Broker* enables and manages asynchronous communication (MQTT) between the devices in the *IoT layer* and the platform components. To secure the data transmitted from the devices in the *IoT layer* and the platform components, the platform implements Transport Layer Security (TLS) [41]. Employing TLS guarantees the encryption of data transmission, mitigating the risk of unauthorized users accessing confidential information. Additionally, the *Certbot* [42] microservice enables Hypertext Transfer Protocol Secure (HTTPS) communications with the platform by automatically generating and renewing the corresponding certificates. The *Internal Websocket proxy* is responsible for the synchronous (REST) communication between the platform components and balancing the workload among them. Communication among platform components also implements TLS. Moreover, the access to the different platform resources (e.g. management of *AWG IoT Devices* and other users) is role-based, which means that the users need to authenticate to have access to different resources and can perform different actions within the platform based on their authentication credentials. We implemented authentication based on JSON Web Tokens (JWT) [43], [44]. The *Auth microservice* is responsible for platform security by checking and authenticating the JWT provided by users, services and devices. This service provides RESTful APIs to handle JWT generation, validation and renewal. The authentication process is crucial to protect data transmissions and ensure security. The *Frontend API Service* manages initial validations, forwards requests to the *Backend Service*, and asynchronously delivers the responses. On the other hand, the *Backend Service* is responsible for the internal management of databases and devices. Moreover, it controls the data transmission of *PostgreSQL* (working as a *PostgreSQL connector*) and the *Cache Service*. The *Cache Service* provides quick access to recent data for a specific time interval, after which it is deleted. The platform employs two databases: i) *PostgreSQL* [45], which handles metadata of users and machines, and ii) *InfluxDB* [46], which collects telemetry data with its corresponding timestamps. The *InfluxDB Connector* manages the data exchange with the *InfluxDB*. This module "listens" to all the telemetry data sent via *Message Broker* and writes it into the *InfluxDB* database. Additionally, the *InfluxDB Connector* enables the retrieval of the telemetry data using REST APIs. Furthermore, the *Plugin Management Service* manages *Custom Plugins*, which are lightweight applications that can extend the platform functionalities in a plug-and-play fashion (more details will be introduced when describing the *Application Layer*). All services must register to the *Service Catalog*, which is responsible for service registration and discovery. Moreover, the *Service Catalog* checks the health of registered services by periodically pinging them, i.e. performing an HTTP request, and checking that the request was successfully received, understood and accepted (HTTP status code 2xx [47]). Otherwise, the service will be marked as offline.

The last platform layer is the *Application Layer*. The *GUI* provides a rich interface that allows users to interact with the platform's features. The *GUI* encapsulates functionalities for displaying and downloading data, deleting AWGs, adding or deleting users, and checking if the other services in the platform are working correctly.

The platform design allows the extension of its features in a plug-and-play manner with the development of *Custom Plugins*. *Custom Plugins* are lightweight applications that can communicate implementing either REST or MQTT, thus supporting both the *request/response* and *publish/subscribe* communication paradigms. Fig. 2 depicts the connection process of a *Custom Plugin* to the platform:

- *(1)-(2)*: The *Custom Plugin* first registers to the *Service Catalog*, providing its name, description, registration time, and eventual endpoints.
- *(3)-(4)*: Afterwards, the *Custom Plugin* requests the endpoint of the *Plugin Management Service* to register as a new plugin.
- *(5)-(10)*: The *Custom Plugin* registers to the *Plugin Management Service*, providing the appropriate user credentials for authentication (either user JWT or password). This means that, during registration, the plugin must provide credentials with the appropriate access level of the resources it will have access to. The *Plugin Management Service* creates a topic for the *Internal Websocket Proxy (6-7)* and retrieves a plugin JWT for the *Custom Plugin* from the *Auth Service (8-9)*. Then, the *Plugin Management Service* returns the response to the *Custom Plugin* with the plugin JWT *(10)*.
- *(11)-(17)*: The *Custom Plugin* retrieves the endpoint of the *Auth Service* and the *Internal Websocket Proxy* from the *Service Catalog (11-12)*. Then, the *Custom Plugin* creates a WebSocket session with the *Internal Websocket Proxy (13)* and makes an authentication request *(14)* providing its own plugin JWT. The *Internal Websocket Proxy* requests the *Auth Service* to authenticate the JWT credentials provided by the *Custom Plugin (15)*. After the *Auth Service* confirmed the credentials *(16)*, the *Internal Websocket Proxy* responds to the *Custom Plugin (17)*.
- Once the *Custom Plugin* has established the WebSocket session and has authenticated, the *Custom Plugin* can communicate in both two ways:
  - *Alternative 1 (Asynchronous Communication)*
    - *(18)-(19)*: the *Custom Plugin* performs the request to be subscribed to a topic.
    - *(20)*: The *Custom Plugin* receives notifications of the subscribed topic.
  - *Alternative 2 (Synchronous Communication)*
    - *(21)-(22)*: another platform service, *Generic Platform Service*, requests the endpoint of the *Custom Plugin* to the *Service Catalog*.
    - *(23)-(24)*: the *Generic Platform Service* makes an HTTP request to the *Custom Plugin*
- *(25)-(26)*: The *Service Catalog* periodically checks the health of the *Custom Plugin*.

B. AWG IoT DEVICE

Our study utilizes the AWG schematized in Fig. 3. The figure outlines the two primary operating modes—*Adsorption* (depicted in Fig. 3a) and *Regeneration* (Fig. 3b)—as well as the status of the actuators and sensors involved in these processes (Fig. 3c). This prototype incorporates an advanced biopolymeric coating made from hydrophilic alginate hydrogel, which has demonstrated exceptional performance even in arid climates.

The AWG machine relies on various components to control its different operating modes, some of which are managed through on/off control, while others utilize Pulse Width Modulation (PWM). Specifically, the system includes a fan controlled via PWM to regulate airflow through the polymeric coating, two on/off valves to alternate between ventilation during *Adsorption* and recirculation during *Regeneration*, and two on/off water pumps. One pump is used for rejecting condensation heat (which also requires an on/off fan), while the other collects condensed water and transports it outside the machine for use. An electrical heater, controlled by PWM, ensures a constant temperature to regenerate the hygroscopic polymer when saturated with atmospheric moisture.

The system is monitored through a multitude of sensors: six for the temperature of air streams, and of condensed water; two for relative humidity inside and outside the AWG machine; two hall sensors give feedback information for the opening/closing of the upper and bottom valve once switching between the *Adsorption* to *Regeneration* mode and vice-versa. All these sensors provide real-time feedback on key operational parameters, which is particularly important during MODE 2 (*Regeneration* - see Fig. 3b).

All of these components and sensors are coordinated through a custom-designed electronic shield, referred to as the *AWG IoT Device*. This device, based on the SAMD21 Cortex-M0+ 32-bit microcontroller with a Wi-Fi antenna, enables seamless communication over the internet, integrating the AWG machine into our IoT ecosystem. Fig. 3c illustrates a simplified representation of the internal control logic employed within the *AWG IoT Device*. Upon powering on, the *AWG IoT Device* initializes its variables and progresses through various states defined within the operational setup. These states include *Idle*, *Adsorption* (*Ads*), *Regeneration* (*Reg*), *Automatic* (*Auto*), and *Stop*. During the *Idle* state, the *AWG IoT Device* primarily awaits commands. After receiving a command from the platform (e.g. issue by a user via GUI or from a *Custom Plugin*), the *AWG IoT Device* transitions to the corresponding operational mode (*MODE*) accordingly. For instance, if the *AWG IoT Device* is in the *Idle* state and receives a command to switch to the *Reg* mode (*MODE* = *2*), it transitions to the *Reg* state and initiates alternating *Regeneration* and *Adsorption* processes. This process continues until a command is received to halt

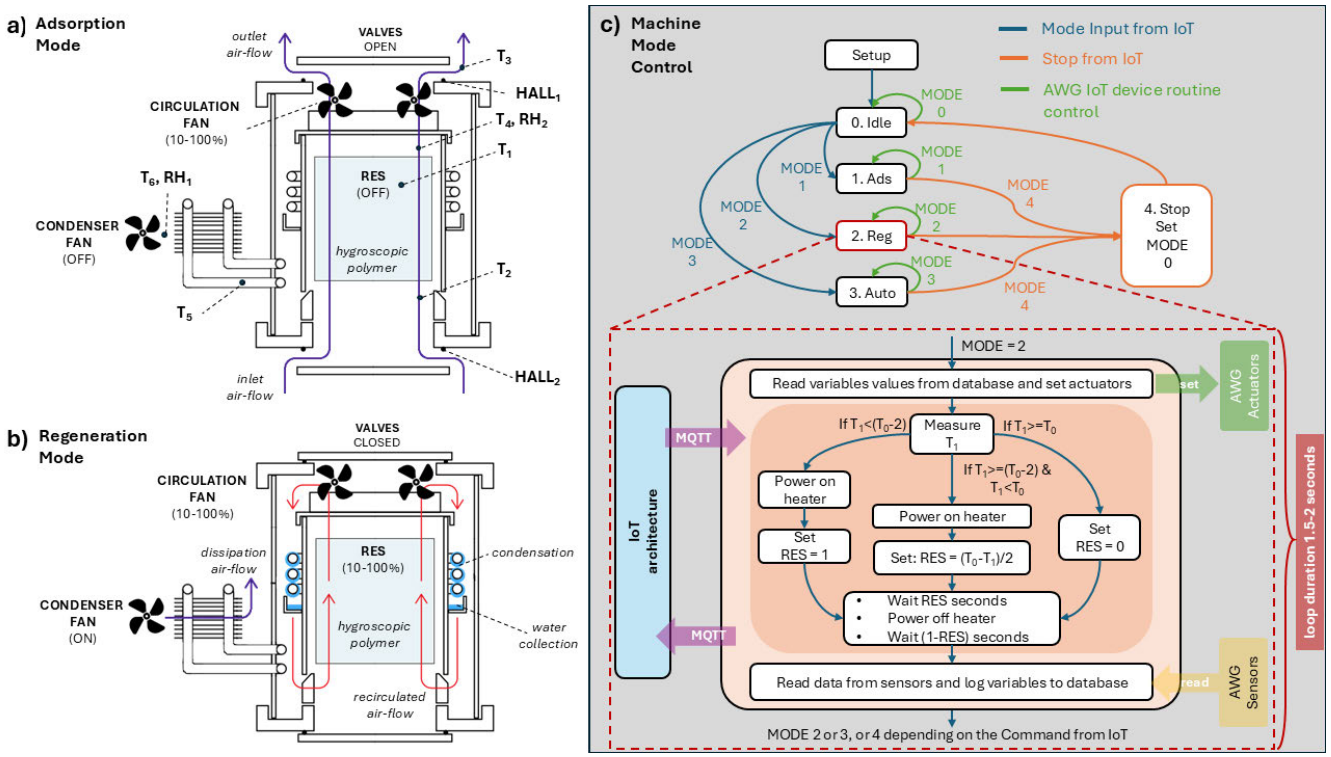


FIGURE 3. Prototype of atmospheric water generator [4]. (a) Schematic of *Adsorption* mode (MODE = 1), showing moisture capture by the hygroscopic polymer from the outdoor environment. (b) Schematic of *Regeneration* mode (MODE = 2), illustrating moisture release and water condensation. (c) Diagram of the system's operational modes, highlighting MODE = 2 as crucial for AWG functionality when integrated with IoT architecture.

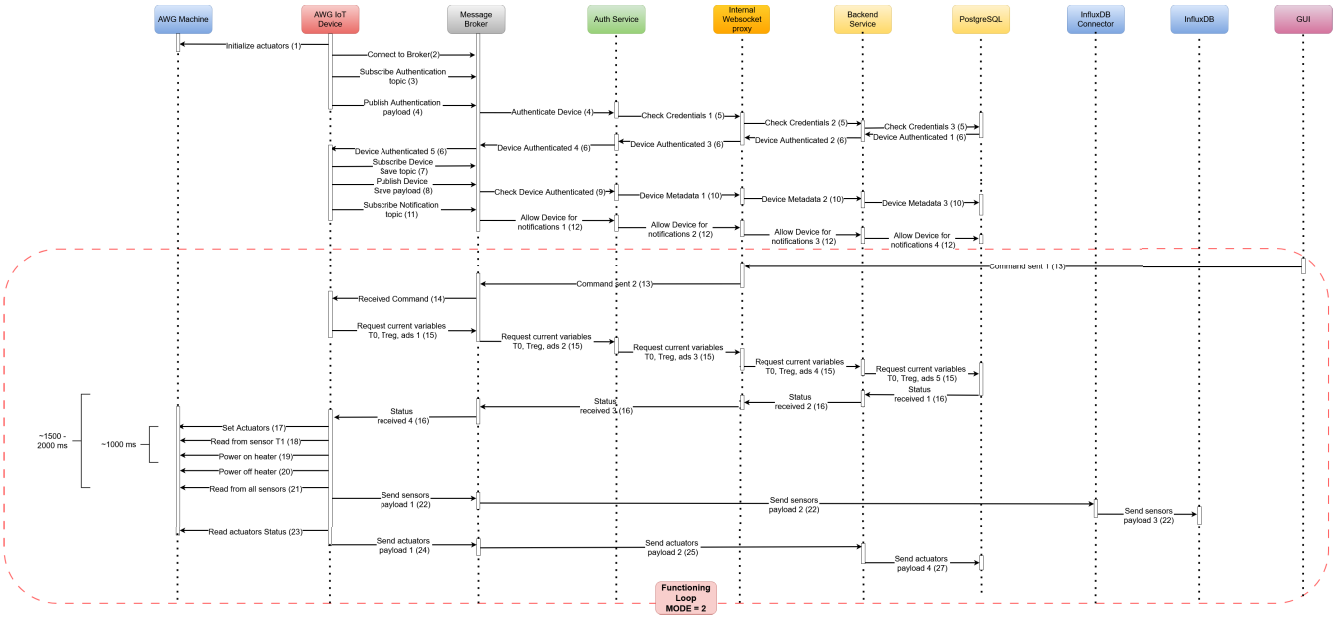


FIGURE 4. Communication flow between AWG IoT device and the platform: Main operational steps for *Reg* state.

operations (*MODE = 4*), leading the *AWG machine* to turn off and return to the *Idle* state. This systematic approach ensures efficient utilization of resources and adherence to user-defined parameters throughout the *AWG machine*'s operation.

Fig. 4 illustrates a complete communication sequence diagram between the *AWG IoT Device* and the platform, elaborating on the operational intricacies of *Reg* state (depicted in the lower part of Fig. 3c), focusing on the *Regeneration* phase of the sorbent material and the subsequent collection of water. Indeed, this is the most critical mode of operation from the view point of communicating information via the IoT platform, and translating coherently in command and actions oriented towards the goal of optimizing water production.

To begin with, the *AWG IoT Device* initialises the actuators (*1*), and connects to the *Message broker* (*2*). The *AWG IoT Device* subscribes to the authentication topic (3). The device then publishes an MQTT message with its corresponding JWT credentials, which the *Message Broker* delivers to the *Auth Service* (4). The *Auth Service*, to authenticate the credentials, performs a request to the *Backend Service* (via *Internal Websocket Proxy*), which in turn, communicates with *PostgreSQL* to verify the information (5). The *Backend Service* returns the device authentication response to the *Auth Service*, which publishes a message under the device authentication topic (6), to which the device subscribed in (3). Afterwards, the *AWG IoT Device* is added to the platform, saving its data (*7-10*). Subsequently, the *AWG IoT Device* is authorised to send measurements to the platform (*11-12*).

Operations (*13-16*) demonstrate the functioning loop of the communications between the *AWG IoT Device* and the platform, which will repeat during the device's lifetime. In contrast, the operations highlighted before will be performed only once during the device's startup. In the *IDLE state* (*MODE = 0*), the *AWG IoT Device* awaits for commands. Upon receiving a command to switch to *Reg* state (*13-14*) (e.g. from a *Custom Plugin* or from users via *GUI*), the device undertakes a sequence of actions (illustrated in the lower part of Fig. 3c). Initially, the *AWG IoT Device* retrieves the desired current variables stored in the *PostgreSQL* database (*15-16*), encompassing parameters like desired temperature (T_0), adsorption duration (t_{ads}), and regeneration duration (t_{reg}). Afterwards, it set the actuators following the information received in the command (*17*). Thereafter, an internal control loop (*18-20*) is initiated to incrementally adjust the temperature of the exchanger within the machine, aligning it with the predetermined set value of T_0. As depicted in the lower part of Fig. 3c, after reading the temperature T_1 (operation (*18*) in Fig. 4), the device executes a series of operations based on the comparison between T_1 and T_0. For example, if T_1 is lower than $T_0 - 2$ it may *Power On heater*, set $RES = 1$ and then *Power Off heater* (operations (*19-20*) in Fig. 4). Subsequently, data from all temperature and humidity sensors are read and transmitted to the database (*21-22*). Afterwards, the actuators status are read and sent to the platform (*23-24*). The complete functioning loop of the *Reg* mode operates over a duration of 1.5-2 seconds. The machine then assesses the *MODE* value: if the user has specified a *MODE* other than 2, the machine transitions to a different state, such as *Stop* (*MODE = 4*); otherwise, it continues the *Regeneration* process until internal criteria for cessation, such as reaching the maximum regeneration time specified in t_{rig}, are met.

IV. CASE STUDY AND EXPERIMENTAL SETUP

This section describes the tests performed to evaluate the proposed *IoT ecosystem*, including the platform and the IoT-enhanced AWG device. First, we describe the tests conducted to evaluate the transmission performance of the proposed framework. To this end, different communication tests were performed to assess the time required by the various entities to communicate over the Internet. Moreover, we will present tests performed on the *AWG IoT Device* in a real-world environment.

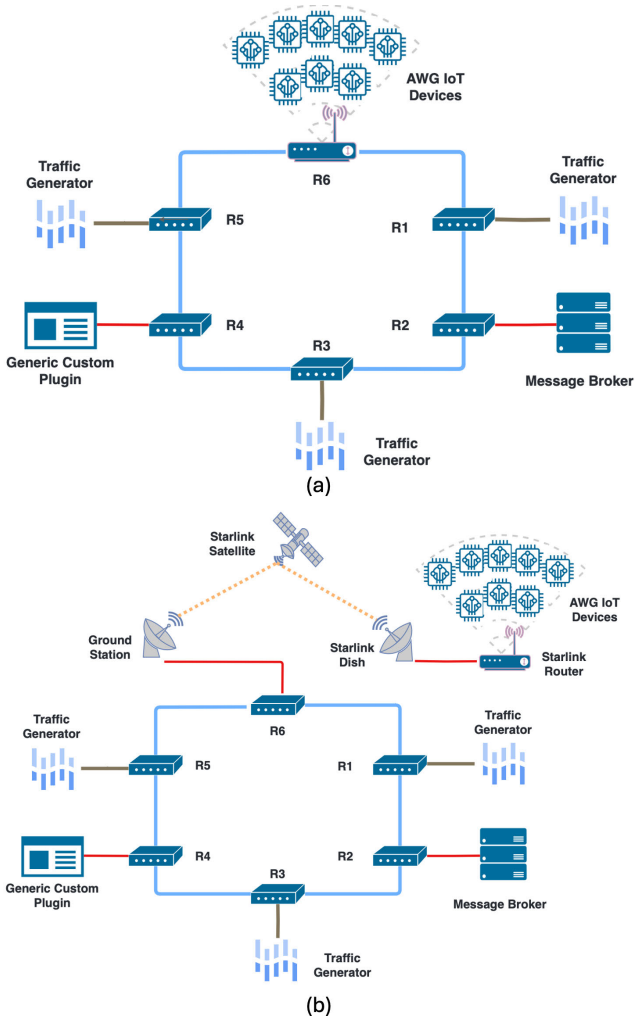


FIGURE 5. Implemented MAN topology for experimental tests in a) Wired Case and b) Starlink Case.

A. COMMUNICATION TESTS SETUP

To study the transmission performance of the proposed infrastructure, we carried out a series of communication tests to evaluate the duration necessary for the various entities to transmit and receive data over the Internet employing the MQTT protocol. We employed Mininet [48] to realistically simulate the communication between the involved entities (i.e. *AWG IoT Devices*, *Message Broker* and *Generic Custom Plugin*) within a Metropolitan Area Network (MAN). Fig. 5a illustrates the structure of our MAN backbone model, which we will refer to as *Wired Case*. The *AWG IoT Devices* and the *Generic Custom Plugin* work both as MQTT publishers and subscribers. In this MAN scenario, the publishers, *Message Broker*, and subscribers are located across various locations in the same metropolitan area and communicate via

the Internet. Typically, MAN backbones utilize fiber-optic links deployed throughout the cities to interconnect various backbone routers in a ring configuration. These fiber-optic links commonly ensure 230-Mb/s connections and, in our model, are configured as full-duplex with a maximum distance of approximately 10 km and zero signal losses (depicted as light blue edges in Fig. 5a). Fig. 5a depicts the assumption we made that all *AWG IoT Devices* are connected to the same backbone router R6. To study the behaviour of the platform with larger fleets, we carried out communication tests employing 1, 100, and 200 *AWG IoT Devices*.

The *Message Broker* is linked to router R2 while the *Generic Custom Plugin* is connected to router R4. R2 and R4 are connected to their respective sub-networks via 150-Mb/s full-duplex links with a maximum distance of about 1 km and zero signal losses (represented as red edges in Fig. 5). Moreover, the *AWG IoT Devices* are connected to R6 via Wi-Fi with a 72-Mb/s connection (in line with the hardware specifications of the NinaW102 Wi-Fi module compatibility with the IEEE 802.11n standard). Additionally, routers R1, R3, and R5 serve three distinct sub-networks generating background traffic at various rates to realistically congest the MAN using iPerf [49]. To prevent bottlenecks, the links between these routers and their traffic generators are configured as 500-Mb/s full-duplex, with a maximum distance of approximately 1 km and zero signal losses (shown as brown edges in Fig. 5a). Thus, in our simulations, the *AWG IoT Devices*, *Message Broker*, and the *Generic Custom Plugin* are situated in different locations within the same metropolitan area and communicate via the Internet. Table 1 presents the parameters utilized for simulating the MAN backbone architecture.

Nonetheless, Atmospheric Water Generators may be employed in remote regions, where there might not be a wired internet connection. In recent years, Low Earth Orbit (LEO) satellites have emerged as a new kind of Internet Service Provider that enables end-users to be connected to the internet directly from space [50]. LEO satellites operate at altitudes ranging from approximately 180 km to 2000 km above the Earth's surface. In contrast to traditional Geosynchronous Orbit (GEO) satellites, which are positioned at around 35780 km, LEO satellites facilitate reduced latency and higher throughput for space-to-ground communications [51]. However, this lower operational altitude has the drawback that each LEO satellite has less coverage than a GEO satellite. LEO satellites can collectively form a LEO Satellite Network (LSN) constellation to deliver high-quality services to ground users with truly global coverage [52]. An inherent challenge with these LSN constellations is the dynamics resulting from the satellites' mobility due to their rapid orbiting velocity, and associated handovers between satellites and the topology dynamics. These can lead to various instabilities across the multiple layers of the Internet network, causing for example, a higher latency and also network interruptions [53]. Moreover, environmental factors such as rain, cloud and temperature can heavily affect the Internet service supply by

TABLE 1. Simulated network main parameters - wired case.

Type of Node	Type of Connection	Transfer Rate
Ring Routers Connection	full-duplex - max length of 10 km - zero losses	up to 230Mbp
R1, R3, R5 towards traffic generators	full-duplex - max length of 1 km - zero losses	up to 500Mbp
R2, R4 towards users	full-duplex - max length of 1 km - zero losses	up to 150Mbp
R6 towards AWG IoT Devices	wireless	up to 72Mbp

TABLE 2. Simulated network main parameters - Starlink Case.

Type of Node	Type of Connection	Transfer Rate
Ring Routers Connection	full-duplex - max length of 10 km - zero losses	up to 230Mbp
R1, R3, R5 towards traffic generators	full-duplex - max length of 1 km - zero losses	up to 500Mbp
R2, R4 towards users	full-duplex - max length of 1 km - zero losses	up to 150Mbp
R6 towards Ground station	full-duplex - max length of 1 km - zero losses	up to 150Mbp
Ground Station, Starlink Satellite and Starlink Dish	wireless	up to 150Mbp
Starlink Dish towards Starlink Router	full-duplex - max length of 1 km - zero losses	up to 150Mbp
Starlink Router towards AWG IoT Devices	wireless	up to 72Mbp

LEO satellites [51], [52], [54]. For example, the throughput rate of the network can be diminished by 5% due to cloudy sky, 17% due to light rain (i.e. less than 4mm per hour) and 30% due to heavy rain (i.e. more than 4mm per hour) [54].

For this reason, we decided to extend our communication tests to a realistic scenario in which the AWGs are located in a remote location and have Internet access provided by SpaceX's Starlink [55] broadband service, one of the industrial leaders of LEO satellites constellations, with more than 5000 operational satellites [56]. Fig. 5b illustrates the network topology implemented for the experimental tests under this scenario, which we will refer to as *Starlink Case*. As can be seen, in the proposed scenario, the Starlink service is employing a bent-pipe strategy [51], which consists of the *Starlink Router* to which all the *AWG IoT Devices* are wireless connected to via Wi-Fi; the *Starlink Dish* which transmits data to and from a *Starlink Satellite*; and finally the *Ground Station* which routes the traffic towards the backbone router R6. We employed LeoEM real-time emulator [53] to capture the intricate dynamics of LEO satellite network. We realistically simulate the network topology (which includes the LEO satnet) using Mininet. The *AWG IoT Devices* are connected to the *Starlink Router* with a 72-Mb/s connection. On the other hand, the *Starlink Router* and the *Starlink Dish* are connected via a 150-Mb/s full duplex link, with a maximum distance of about 1 km and zero signal losses (represented as red edges in Fig. 5b). Moreover, the link properties between the *Starlink Dish*, *Starlink Satellite*

and *Ground Station* (depicted as yellow edges in Fig. 5b) are dynamically changed during the simulation according to LeoEM emulator, such as variations in propagation latency and network disruptions due to handovers between satellites. Initially, these links are set with a connection bandwidth of 150-Mb/s to be compliant with Starlink's claimed network throughput [53]. Moreover, to study the behaviour of the network topology under different weather conditions, i.e. light rain and heavy rain as done in [54], the bandwidth of the Starlink connections (i.e. yellow links in Fig. 5b) were diminished 17% (i.e. 120-Mb/s) for the light rain case, and 30% (i.e. 83-Mb/s) for heavy rain conditions. Regarding the weather influence on the Wi-Fi connection, adverse weather conditions mainly influence the Received Signal Strength Indicator, which is a metric that quantifies the power level of a received radio signal [57]. However, empirical data showed that the impact of adverse weather conditions (like cloudy or even with rainstorms of 15.2 cm/hr) with devices connected within less than 800 m can be neglected [58]. Table 2 presents the parameters utilized for simulating the MAN backbone architecture using Starlink as Internet Service Provider for the *AWG IoT Devices* illustrated in Fig. 5b.

Within this framework, in both cases (i.e. *Wired* and *Starlink Case*) we test the following scenario: the message sequence is launched by an MQTT publish action initiated by each *AWG IoT Device* sending the AWG machine sensors measurements (MQTT publisher), which are received by *Generic Custom Plugin* (MQTT subscriber). Since we are interested in the communication performance, the *Generic Custom Plugin* just receives the message containing the sensors measurements and publishes a command (MQTT publisher) to the corresponding AWG that issue the message, which is later received by this device (MQTT subscriber). We conducted the tests with the *AWG IoT Device* transmitting the sensors' measurements periodically every 1s.

B. PROTOTYPE TESTING IN A REAL ENVIRONMENT

The AWG machine shown in Fig. 6 was tested in a climate typical of dry and remote regions. It produces water without refrigeration by condensing it at ambient temperature (T_{amb}). During the *Adsorption* phase, ambient air passes through the machine, allowing the sorbent polymer (97.5 gr of alginate based composite hydrogel) to capture moisture. Once this phase is complete, two valves seal the regeneration chamber where moisture is converted into liquid water (as shown in the schematics of Fig. 3a and Fig. 3c). Heat for regeneration is provided by an electric resistor (250 W), and condensation heat is rejected using a dry cooler facing the outdoor environment. A water circulator pumps water through the dry cooler loop, maintaining the coil temperature close to the outdoor value. Detailed schematics and system functionality are provided in [4]. The prototype is equipped with three sensors to measure temperature and relative humidity of the air circulating inside the machine and of the outdoor environment, along with one temperature sensor to monitor

FIGURE 6. Prototype of atmospheric water generator during testing in texas, palo duro state park [4].

the polymer. The temperature sensors had an accuracy of $\pm$ 0.3°C, while relative humidity $\pm$ 1.5%. This sensor controls the thermostatic loop, adjusting the resistor's power via a solid-state relay based on the temperature feedback. The hysteresis in the thermostatic loop to adjust regeneration temperature had an interval of $\pm$ 0.75°C. An axial fan (20W) moves air during both operational modes (*Ads* and *Reg*), controlled by a PWM signal from the *AWG IoT Device* to adjust the airflow rate according to the water production needs. A water circulator (10W max) flows water through the dry cooler loop for the rejection of the condensation heat. Two 24V DC motors drive the aeraulic valves that open and close the regeneration chamber air loop, and a peristaltic pump collects the liquid water converted from atmospheric moisture into the final collection tank. Tests are carried out using a t_{reg} equal to 80 min, t_{ads} of 40 min, while the regeneration temperature is equal to 75 °C. These settings result from a previous analysis carried out in a previous experience in laboratory tests [21].

V. EXPERIMENTAL RESULTS

In this section, we present the results obtained for the tests previously described in Section IV-A and Section IV-B.

A. COMMUNICATION RESULTS

Firstly, we will examine and analyze the results of the experiments with Mininet [48], focusing on the timing of the message delivery process. The message sequence involved an MQTT publish action initiated by the *AWG IoT Devices*, which sent the sensor measurements. Since our focus is on the communication performance, the *Generic Custom Plugin* received the message, and published a hypothetical command back to the corresponding *AWG IoT device* i.e. the one that sent the message.

As variables, we chose to have the two different proposed scenarios (*Wired* and *Starlink Case* depicted in Fig. 5), each with different working conditions of the network. As network conditions, we imposed various traffic levels, simulating how

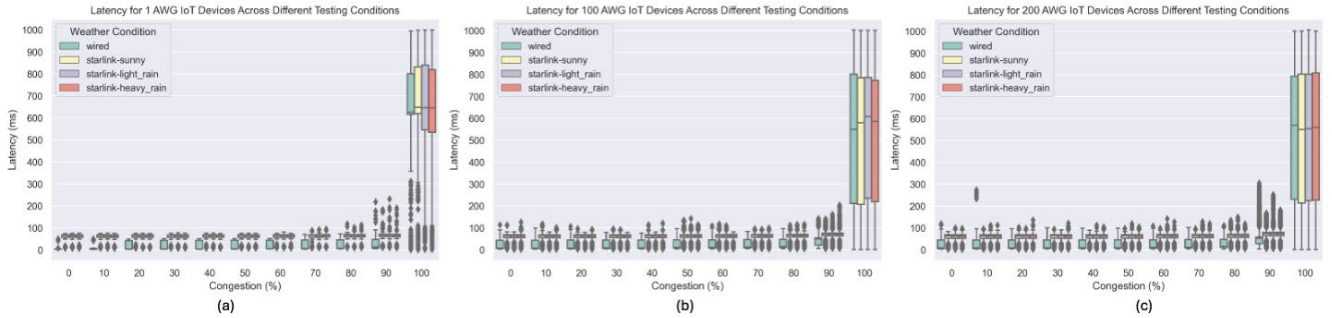


FIGURE 7. Communication latency as a function of the level of traffic over the network for (a) 1, (b) 100, and (c) 200 AWG IoT devices.

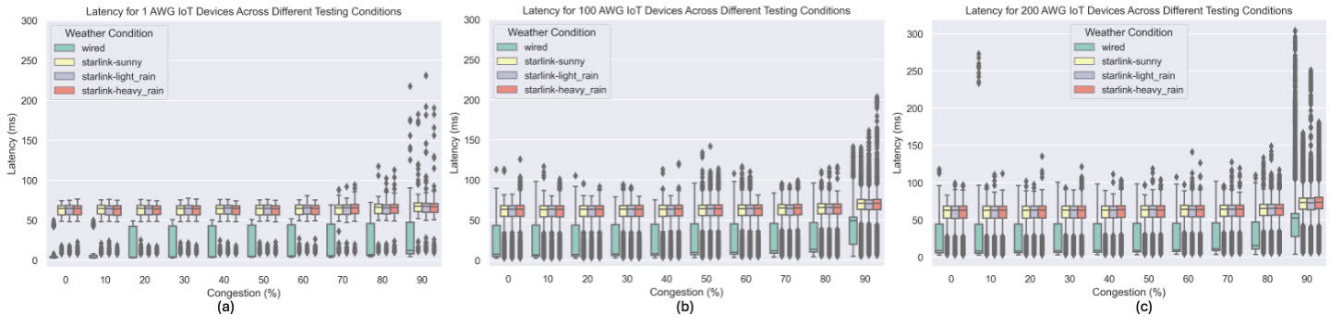


FIGURE 8. Communication Latency as a function of the level of traffic (except 100%) over the network for (a) 1, (b) 100, and (c) 200 AWG IoT devices.

the platform would perform in different clogged situations. Moreover, for each of the proposed scenarios and traffic levels, we tested the behaviour of the ecosystem with 1, 100 and 200 *AWG IoT Devices*. Furthermore, for the Starlink Case, we set different weather conditions (i.e. sunny, light rain, and heavy rain) which affect the Starlink bandwidth connection (depicted as yellow edges in Fig. 5b), i.e. 150-Mb/s, 120-Mb/s and 83-Mb/s, respectively. Regarding the test itself, we considered two different payloads, both in JSON format: one sent by the *AWG IoT Device* to the platform and one sent by the *Generic Custom Plugin* to the corresponding *AWG IoT Device*, both transmitted via MQTT. In the first case, the payload was 865 bytes and contained the sensor measurements. There are three types of sensors: i) *relative humidity*, with one sensor measuring the outdoor environment's relative humidity and one employed inside the *AWG machine*; ii) *temperature*, with six sensors placed in different parts of the *AWG machine*; and iii) two sensors measuring *hall voltage*. On the other hand, the payload sent by the *Generic Custom Plugin* was 285 bytes and comprised commands for the LED, the water gathering pump, the capacitor's power, the fan's power, the *MODE*, temperature T_0, adsorption duration (T_{ads}), and regeneration duration (t_{reg}).

Fig. 7 reports the latency for the tested scenarios, i.e. *Wired* and *Starlink* cases with Sunny, Light Rain, and Heavy Rain conditions. Instead, Fig. 8 illustrates the results without the 100% congestion level. As depicted in Fig. 8, the median latency in the *Wired Case* is lower than in the *Starlink* one, with values below 50 ms in the first case and 65 ms in the second one. This expected behaviour is because data in the *Starlink Case* must be transmitted from the *Starlink Dish* to the *Starlink Satellite*, to finally reach the *Ground Station* (as shown in Fig. 5b). On the other hand, as shown in Fig. 8, there can be seen a slight increase in the communication latency for network congestions of 90% compared to lower congestions. Furthermore, there is almost a negligible impact of the weather on the transmission times, with quite similar behaviour observed for the three weather conditions studied (i.e., Sunny, Light Rain, and Heavy Rain). This can be due to the high bandwidth speed (i.e., higher than 83-Mb/s) with respect to the "lightweight" payload, i.e. 865 bytes for the message with the sensor measurements, and 285 bytes for the commands sent to the *AWG IoT Device*. Regarding the platform behaviour under larger fleets, as observed in Fig. 8, there is a small impact in the observed latency in the *Wired Case* for network congestion of 90%, with a median value lower than 25 ms for 1 AWG, while the latency is close to 50 ms for 100 and 200 AWGs. In contrast, there is almost no influence in the *Starlink Case*.

As can be seen in Fig. 7, the latency in all the studied cases is not problematic since, at most, it reaches a value of around 100 ms (300 ms if we consider the outliers). The only exception is observed in Fig. 7 for total network congestion (network congestion of 100%). This network congestion is caused by the generated background traffic (router R1, R3, and R5 in Fig. 5) and not by the *AWG IoT Devices*. Nonetheless, Internet service providers try to prevent such

severe network congestion since prolonged periods could result in the collapse of the MAN network itself. Moreover, the fact that for the use case the AWG functioning loop for the *Reg* state (illustrated in Fig. 3c and Fig. 4 *operations (15-24)*) is around 1.5-2 seconds, together with the frequency which the *AWG IoT Device* sends the sensors measurements (i.e. every 10s) in a real-scenario, demonstrates that these latencies do not represent a problem at all, and guarantee that the platform is more than appropriate for handling the expected workload.

B. RESULTS OF PROTOTYPE TESTS

Fig. 9 shows the monitored parameters during multiple consecutive thermal cycles over 12 hours of continuous operation. The graph shows the dry air temperature (T_1), outdoor air dry temperature (T_{amb}), and dew points within the regeneration chamber ($T_{dew-chamber}$) and the outdoor air ($T_{dew-ambient}$), calculated using humidity sensors. During the regeneration period, the alginate-based polymer reached a surface temperature of 75°C, with the maximum air temperature (T_1) at 65°C. When the electric heater is activated, the internal dew point rises from 12°C (matching the outdoor $T_{dew-ambient}$) to 35°C. Water vapor condenses spontaneously in the condensation chamber as $T_{dew-chamber}$ exceeds $T_{dew-ambient}$, which was approximately 23°C during the experiment. In contrast, conventional refrigeration systems would need temperatures below 12°C to condense moisture from the air. During the adsorption phase, the system's relative humidity varied between 32% and 50%, increasing as the ADS-HX absorbed moisture, while the ambient relative humidity remained around 50%. After five consecutive cycles, corresponding to half a day of operation (aligned with solar radiation availability), the average water collected per day was 2.1 litres per kilogram of sorbent polymer. The device had a daily electricity consumption of 0.7 kWh (with an average absorbed power of 58W) to operate all actuators, the fan, and the electric resistor for sorbent regeneration, which was the main source of electricity consumption.

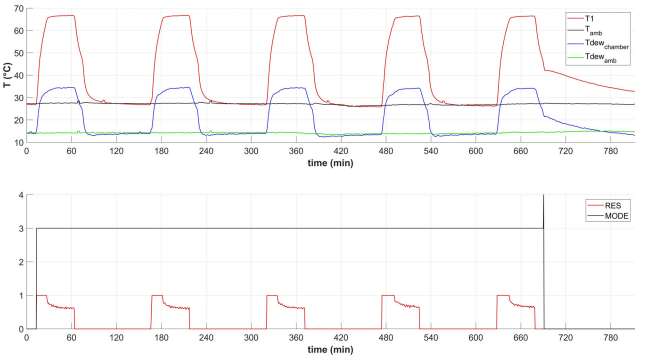


FIGURE 9. The top graphic shows the relevant physical parameters monitored and transmitted through the IoT platform related to the *regeneration* operational mode monitored while testing the machine in the real environment. The bottom graphic shows the Mode signal (MODE = 3) together with the PWM control transmitted through the IoT platform in order to achieve thermostatic regeneration of the sorbent polymer.

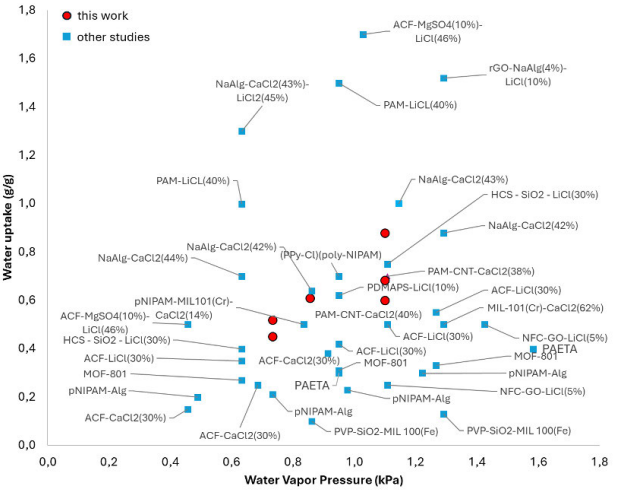


FIGURE 10. Comparison of the sorbent used in this work in terms of quantity of water adsorbed per kilos of dry material (water uptake g/g) at different water vapor pressure corresponding to dry and arid areas. The material references are taken from Table 3.

C. DISCUSSIONS

Unlike conventional AWG systems, which rely on energy-intensive vapor compression to cool air to the dew point, the proposed sorption-based approach offers a more efficient alternative. Traditional AWG technologies become ineffective when relative humidity drops below 30% and typically require more than 0.6 kWh of electricity per liter of produced water, often exceeding 1 kWh/L in hot and dry conditions. In contrast, the proposed sorption-based system utilizes heat as the primary energy source, with an energy consumption of approximately 0.7 kWh of thermal energy per liter of produced water [8]. Additionally, refrigeration-based AWG systems struggle in arid environments where the dew point falls below freezing, making direct condensation impractical. By leveraging advanced hydrogel sorbents, the sorption-based approach enhances water uptake efficiency across a broad range of temperature and humidity conditions, maintaining effectiveness even at low water vapor pressures (<1 kPa), where conventional technologies fail. This makes it particularly well-suited for arid environments.

Table 3 and Fig. 10 highlight the performance of the technology used in this study, focusing on the moisture uptake of the materials and the daily water productivity of the system compared to other studies in the literature. Fig. 10 presents various performance data retrieved from the literature (full list in Table 3, which also includes the corresponding operating conditions. If not reported by the authors, the respective cell is left blank). Each point is labeled with the acronym of the material studied in the respective research. The acronym typically represents the type of sorbent or polymeric medium used, along with the salt concentration for impregnation. The biopolymer

TABLE 3. Summary of various studies and sorbent materials used in the AWG sector, along with the demonstrated daily water productivity under specific environmental conditions, including ambient temperature (T_{amb}), relative humidity (RH_{amb}), and the regeneration temperature (T_{reg}) required to drive the atmospheric water harvesting cycle.

Reference	Material	T_{amb} °C	T_{reg} °C	Sorbent mass	Daily water yield
This work	CaAlg with graphite	20-30	60	∼100 g	1.6-2.9 $L/kg/day$ with $RH_{amb} < 30\%$
[59]	pNIPAM-Alg	21	60	-	-
[5]	PAM-CNT-CaCl2(38%)	25	80	-	-
[60]	ACF-LiCl(30%)	25	94	10 kg	0.39 $L/kg/day$ with 70% RH_{amb}
[61]	NFC-GO-LiCl(5%)	25	-	30 g	-
[62]	ACF-MgSO4(10%)-LiCl(46%)	20	120	-	0.9 $L/kg/day$ with 70% RH_{amb}
[63]	PVP-SiO2-LiCl	20	70	-	-
[64]	PVP-SiO2-MIL 100(Fe)	30	120	0.2 g	-
[65]	PAM/Alg-CNF-CaCl2(30%)	23	60	-	0.9 $L/kg/day$ with 70% RH_{amb}
[66]	(PPy-Cl)(poly-NIPAM)	25	80	-	0.8-14.8 $L/m^2/day$ with 30-60% RH_{amb}
[67]	rGO-NaAlg(4%)-LiCl(10%)	30	90	-	1.4-10 $L/m^2/day$ (unknown RH_{amb})
[68]	PAETA	25	50	-	-
[69]	PAM-CNT-CaCl2(40%)	25	57	-	-
[70]	pNIPAM-MIL101(Cr)-CaCl2(14%)	23	65	-	4.2 $L/kg/day$ with 90% RH_{amb}
[71]	NaAlg-CaCl2(43%)	28	150	-	-
[72]	NaAlg-CaCl2(44%)	25	80	-	-
[72]	NaAlg-LiCl(46%)	25	80	-	-
[72]	NaAlg-CaCl2(43%)-LiCl2(45%)	25	70	-	-
[3]	MOF-801	25	85	-	1.25 $L/kg/day$ with 30% RH_{amb}
[73]	PAM-LiCL(10%)	25	70	-	-
[73]	PAM-LiCL(40%)	25	70	-	2.6 $L/kg/day$ with 20% RH_{amb}
[74]	HCS - SiO2 - LiCl(30%)	25	80	-	2.5 $L/kg/day$ with 60% RH_{amb}
[75]	NaAlg-CaCl2(42%)	30	140	-	-
[76]	ACF-CaCl2(30%)	20	120	-	-
[77]	MIL-101(Cr)-CaCl2(62%)	30	80	-	-
[78]	PDMAPS-LiCl(10%)	25	80	-	5.87 $L/kg/day$ with 20% RH_{amb}

analyzed in this study (red circles in Fig. 10) demonstrates a higher moisture uptake (y-axis) than most other sorbents (light blue squares) at equivalent vapor pressure (x-axis, in kPa). However, a few alternative solutions demonstrate slightly better performance. In particular, some studies utilize composite sorbents mixed with lithium chloride (*LiCl*) to enhance hygroscopic properties. However, lithium chloride is highly toxic to humans. Since this study focuses on producing drinking water, such solutions were not considered viable for a safe and sustainable real-world application. By contrast, the materials used in this study are bio-derived and composed of edible compounds, ensuring both safety and sustainability. For this reason, despite its lower moisture uptake in the same operating range, we chose to use the alginate-based biopolymer impregnated with $CaCl_2$

The results obtained in the communication tests carried out highlighted the platform's robustness and reliability, ensuring that, even under near-optimal conditions and with fleets of hundreds of *AWG IoT Devices* as shown, the proposed framework can maintain efficient performance and meet operational demands without significant latency issues. For increasing number of *AWG IoT Devices*, the *Message Broker* might result in a resource bottleneck. Nonetheless, it is possible and straightforward to replace it with a scalable broker, like RabbitMQ [79] that enables horizontal scalability, i.e. clustering, by providing multiple nodes working together as a single logical message broker. Instead, the other platform services might not face any issue since they can be horizontally scaled during runtime thanks to the platform architecture employed.

For unlocking the full potential of AWG machines and enable the energy water-nexus, it is required to coordinate operations across fleets of *AWG IoT Devices* powered by renewable energy to minimize energy usage at a community level. This coordinated management might enable demand-response flexibility in urban environments' multi-energy systems (i.e. electrical, thermal, gas and other energy vectors) [9], [10], [11]. Nonetheless, the integration of renewable energy generation units (e.g. solar and wind energy), coupled with the unpredictability of users' energy consumption patterns and the rise of active loads such as electric vehicles, introduces heightened uncertainty into sustainable energy and multi-energy systems [80]. This increased complexity renders optimising multi-energy systems operations more challenging and intricate. In this scenario, Reinforcement Learning (RL) [81] can play a key role.

RL represents the challenge faced by an autonomous agent that learns the optimal behaviour of a dynamic and evolving environment through iterative trial-and-error interactions [81]. RL operates independently of a predefined model of the system dynamics and it is not sensitive to the physical model of the object/system under investigation. RL can be employed to optimise the operation of AWGs in a multi-energy system scenario, focusing on the energy supply side, on the user side, and on the whole system [80]. On the energy supply side, if we consider AWGs as Energy Storage Systems (ESS), RL can be used to optimize the operation of AWGs to control these devices to operate when there is a surplus of electric power, contributing

to balance energy production and demand. Regarding the optimization from the user side, AWGs have great potential for residential demand side management since they can be seen as time scaling and shifting electrical loads according to [82]. Furthermore, from the entire system point of view, AWGs can be seen as autonomous agents involved in the complex interaction of the whole system, modelling users and energy suppliers as autonomous agents too, where RL can be employed to optimize the overall cooperative operation of the entire system [80]. Last but not least, RL unlocks the possibility of scaling the presented ecosystem for diverse geographical regions with varying humidity levels. This is because maximizing water productivity and optimizing AWG energy consumption for varying relative humidity is possible by adjusting the regeneration temperature of AWGs (for more details, see [4], [21]). Therefore, RL can be employed to adjust the regeneration temperature accordingly (i.e., by sending control commands to AWG machines) based on the varying humidity levels of different geographical regions by learning in an iterative trial-and-error basis from previous observations.

Regarding the integration of Reinforcement Learning in the proposed IoT ecosystem, it can be seen as centralized control, completely decentralized control or a hybrid one. In the case of centralized control, RL can be integrated into the ecosystem as a *Custom Plugin* (as highlighted in Fig. 1). In this scenario, a single system optimises the operation of AWGs, (it can be only one or an entire fleet of AWGs) based on certain objectives. For example, the Centralized Control can be employed to optimise each AWG machine to work efficiently with the humidity conditions that the device is facing, which will dynamically adapt from one region to another. Moreover, the Centralized control can be used to minimize energy consumption at a community level by activating the regeneration phase based on weather forecasts retrieved from third party APIs and water demand forecasts. Another example can be that the Centralized Control activates the regeneration phase of AWGs to work as ESS based on the power forecasts of the Grid Management Entity.

However, centralized control depends on a singular point of decision, which can give rise to scalability challenges and increased vulnerability to system failures. Additionally, centralized control might lead to higher communications costs, e.g. in terms of memory requirements or processing capabilities [83], [84]. To overcome these challenges, recent works have focused on decentralized control approaches. These approaches rely on the computational power of the devices in the IoT layer, known as Edge Computing [85]. For example, [83] employed a Reinforcement Learning (RL) methodology based on Q-learning to decentralized planning for truck platoons in road transportation systems. In the proposed IoT ecosystem for AWGs, the decentralized control can be seen to optimise the water production of a fleet of AWGs, coordinating their operations, based on their current load, external moisture, etc. Last but not least, an hybrid control system can be employed, where there is the RL *Custom Plugin*, while others decisions are taken based on the decentralized algorithms among *AWG IoT Devices*. The decision of whether to employ a centralized control or a decentralized one might depend on CPU requirements, energy consumption, latency, and task size. Reference [84] implemented a RL approach based on Federated Dueling Deep Q network to optimise and balanced energy consumption and latency of IoT devices. In the proposed framework, each IoT device has the possibility to choose whether to employ local computation, cloud computation or edge computation.

The presented IoT ecosystem represents a significant step towards achieving sustainability by enabling the remote monitoring and control of a fleet of AWG machines. By transforming conventional AWGs into smart connected systems capable of autonomous operation and remote management, this integration is particularly valuable for arid and remote regions where water scarcity demands efficient, self-regulating, and low-maintenance solutions. While the proposed ecosystem does not yet integrate algorithms (like Reinforcement Learning) for coordinating AWGs operations, its architecture paves the way for the integration of these algorithms in a plug-and-play manner. Its architecture combined with the system's ability to process and analyse data at scale, while ensuring security and reliability, establishes a robust foundation for advancing sustainable water production technologies unlocking the full potential of AWGs to overcome the global water crisis and achieve the energy-water nexus.

VI. CONCLUSION

In this article, we presented a novel IoT ecosystem which enables the remote monitoring and control of a fleet of Atmospheric Water Generators. The ecosystem comprises a distributed platform and IoT-enhanced Atmospheric Water Generators through *AWG IoT Devices*. The platform follows the microservices design pattern for handling data collection, storage, and management from AWGs. The platform design guarantees scalability for a growing number of devices and the management of heterogeneous data. On the other hand, we add IoT capabilities through the *AWG IoT Device* to an innovative *AWG machine* which is able to generate water from atmospheric moisture by means of a biopolymer.

We tested the complete IoT ecosystem in a real-world scenario. On one hand, to evaluate the transmission performance of the proposed ecosystem, we carried out a series of communication tests to assess the latency of the communication between various platform entities over the Internet. Therefore, we simulated a realistic scenario of the proposed IoT ecosystem in a Metropolitan Area Network where these AWG machines were used as building appliance for the local production of drinkable water. Given that AWGs may be situated in remote areas, we also studied the scenario in which AWGs are connected to the Internet via Starlink, an industrial leader in the Low Earth Orbit (LEO) satellite-based Internet Service Providers. The results proved that

the transmission latencies are compliant with the studied *AWG machine* operating modes. Furthermore, these tests demonstrated the suitability of the implemented platform for properly monitoring and managing AWGs. On the other hand, we showcased the implementation of the *AWG IoT Device* for the operation of the *AWG machine* for water generation in a real environment, demonstrating the correct functioning of the *AWG machine* together with the monitoring of the relevant parameters and control of the actuators.

The presented IoT ecosystem represents a significant step towards achieving sustainability by enabling the remote monitoring and control of a fleet of AWG machines. By transforming conventional AWGs into smart connected systems capable of autonomous operation and remote management (i.e. *AWG IoT Devices*), this integration is particularly valuable for arid and remote regions where water scarcity demands efficient, self-regulating, and low-maintenance solutions. Its architecture unlocks the possibility of integrating advanced algorithms (like Reinforcement Learning) for coordinating AWGs operations in a plug-and-play manner as Custom Plugins. The ecosystem architecture combined with its ability to process and analyse data at scale, while ensuring security and reliability, establishes a robust foundation for advancing sustainable water production technologies unlocking the full potential of AWGs to overcome the global water crisis and achieve the energy-water nexus.

Future works may include the development and integration of post-processing services, leveraging on the stored data. For example, machine learning algorithms for predictive maintenance can be implemented by analysing and linking the different monitored variables. Furthermore, Reinforcement Learning algorithms can play a key role by enabling the cooperative operation of a fleet of AWGs powered by renewable energy sources, unlocking demand-response flexibility in multi-energy systems, aligning water production with fluctuations in renewable energy availability. By modulating AWG operation in response to excess renewable energy generation, the ecosystem can effectively convert surplus electrical energy into stored water, enhancing grid stability and promoting sustainable water-energy management at a community level. Moreover, RL can enable the dynamic regulation of AWG units boosting the scalability of the presented ecosystem to diverse geographical regions with varying humidity levels as well as to adapt to varying environmental conditions based on predictive meteorological data, including temperature, relative humidity and solar radiation. The dynamic regulation of AWGs optimizes moisture uptake and regeneration timing, ensuring efficient water production while adapting to environmental constraints.

REFERENCES

[1] Z. Shao, H. Lv, P. Poredoš, S. Su, R. Sun, H. Wang, S. Du, and R. Wang, "Scaled solar-driven atmospheric water harvester with low-cost composite sorbent," *Energy*, vol. 302, Sep. 2024, Art. no. 131917.

[2] Y. Tu, R. Wang, Y. Zhang, and J. Wang, "Progress and expectation of atmospheric water harvesting," *Joule*, vol. 2, no. 8, pp. 1452–1475, Aug. 2018.

[3] H. Kim, S. Yang, S. R. Rao, S. Narayanan, E. A. Kapustin, H. Furukawa, A. S. Umans, O. M. Yaghi, and E. N. Wang, "Water harvesting from air with metal-organic frameworks powered by natural sunlight," *Science*, vol. 356, no. 6336, pp. 430–434, Apr. 2017.

[4] V. Gentile, M. Bozlar, M. Caló, M. Simonetti, and F. Meggers, "Alginate biopolymeric coated heat exchanger for atmospheric water harvesting," *ACS ES T Water*, vol. 4, no. 4, pp. 1874–1882, Apr. 2024.

[5] R. Li, Y. Shi, M. Alsaedi, M. Wu, L. Shi, and P. Wang, "Hybrid hydrogel with high water vapor harvesting capacity for deployable solar-driven atmospheric water generator," *Environ. Sci. Technol.*, vol. 52, no. 19, pp. 11367–11377, Oct. 2018.

[6] H. Kim, S. R. Rao, E. A. Kapustin, L. Zhao, S. Yang, O. M. Yaghi, and E. N. Wang, "Adsorption-based atmospheric water harvesting device for arid climates," *Nature Commun.*, vol. 9, no. 1, p. 1191, Mar. 2018.

[7] H. Qi, T. Wei, W. Zhao, B. Zhu, G. Liu, P. Wang, Z. Lin, X. Wang, X. Li, X. Zhang, and J. Zhu, "An interfacial solar-driven atmospheric water generator based on a liquid sorbent with simultaneous Adsorption–Desorption," *Adv. Mater.*, vol. 31, no. 43, Oct. 2019, Art. no. 1903378.

[8] V. Gentile, M. Bozlar, F. Meggers, and M. Simonetti, "Liter-scale atmospheric water harvesting for dry climates driven by low temperature solar heat," *Energy*, vol. 254, Sep. 2022, Art. no. 124295.

[9] A. Entezari, O. C. Esan, X. Yan, R. Wang, and L. An, "Sorption-based atmospheric water harvesting: Materials, components, systems, and applications," *Adv. Mater.*, vol. 35, no. 40, Oct. 2023, Art. no. 2210957.

[10] A. M. Omer, "Energy, environment and sustainable development," *Renew. Sustain. Energy Rev.*, vol. 12, no. 9, pp. 2265–2300, 2008.

[11] Ş. Kılkış, "Sustainable development of energy, water and environment systems index for Southeast European cities," *J. Cleaner Prod.*, vol. 130, pp. 222–234, Sep. 2016.

[12] S. Kadam, P. Kadam, P. Jadhav, S. Gurav, V. Patil, and A. Prabhakar, "Atmospheric water generator using smart automation systems," *Int. J. Electr. Electron. Eng.*, vol. 11, no. 4, pp. 206–216, Apr. 2024.

[13] E. Sudarshan, S. N. Korra, K. Prof. Rajasekharaiah, S. Venkatesulu, and A. Harshavardhan, "IoT based smart solar atmospheric water harvesting system," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 981, Dec. 2020, Art. no. 042004.

[14] R. Munsin, A. Wannachai, N. Chongbun, S. Karnpian, N. Sumankant, K. Sanwong, S. Pinta, J. Panya, P. Yeunyongkul, and N. Nuntapap, "Development of microclimate control room using IoT system for atmospheric water harvesting research," *Eng. Access*, vol. 8, no. 2, pp. 336–344, 2022.

[15] E. Dar, K. Rana, M. Z. Hussain, M. Z. Hasan, A. Khalid, A. Javaid, J. Altaf, B. Sattar, S. H. Chuhan, M. A. Yaqub, A. M. Qureshi, and Z. Mubarak, "Optimizing atmospheric water generation through advanced weather prediction using solar energy data," *IEEE 9th Int. Conf. for Converg. Technol. (I2CT)*, pp. 1–8, Apr. 2024.

[16] M. A. K. Jana, "IoT based smart irrigation system with water generator," *Int. J. for Res. Appl. Sci. Eng. Technol.*, vol. 8, no. 6, pp. 260–267, Jun. 2020.

[17] E. O. Amuta, H. Orovwode, S. T. Wara, A. F. Agbetuyi, S. Matthew, and E. F. Esisio, "Hybrid power microgrid optimization and assessment for an off-grid location in Nigeria," *Mater. Today, Proc.*, vol. 105, pp. 155–161, 2024.

[18] X. Zhou, H. Lu, F. Zhao, and G. Yu, "Atmospheric water harvesting: A review of material and structural designs," *ACS Mater. Lett.*, vol. 2, no. 7, pp. 671–684, Jul. 2020.

[19] D. Milani, A. Abbas, A. Vassallo, M. Chiesa, and D. A. Bakri, "Evaluation of using thermoelectric coolers in a dehumidification system to generate freshwater from ambient air," *Chem. Eng. Sci.*, vol. 66, no. 12, pp. 2491–2501, Jun. 2011.

[20] N. Hanikel, M. S. Prévot, and O. M. Yaghi, "MOF water harvesters," *Nature Nanotechnol.*, vol. 15, no. 5, pp. 348–355, May 2020.

[21] V. Gentile, M. Calò, M. Bozlar, M. Simonetti, and F. Meggers, "Water vapor mass transfer in alginate–graphite bio-based hydrogel for atmospheric water harvesting," *Int. J. Heat Mass Transf.*, vol. 219, Oct. 2023, Art. no. 124794.

[22] H. Mittal, A. Al Alili, S. M. Alhassan, and M. Naushad, "Advances in the role of natural gums-based hydrogels in water purification, desalination and atmospheric-water harvesting," *Int. J. Biol. Macromolecules*, vol. 222, pp. 2888–2921, Dec. 2022.

[23] Y. Guo, J. Bae, Z. Fang, P. Li, F. Zhao, and G. Yu, "Hydrogels and hydrogel-derived materials for energy and water sustainability," *Chem. Rev.*, vol. 120, no. 15, pp. 7642–7707, Aug. 2020.

[24] X.-T. Jin, L. Shao, M. Liu, J. Zhao, C. Xue, S.-X. Zhang, P. Feng, and Y.-H. Luo, "Smart coating materials of buildings for atmospheric water harvesting and response," *Mater. Today Sustainability*, vol. 27, Sep. 2024, Art. no. 100848.

[25] G. A. Tsalidis, D. Dias, A. Martins, V. Vasilaki, J. M. Ribeiro, and E. Katsou, "Assessing the ISO hierarchy validity in circular wastewater treatment life cycle assessments: A Portuguese case study," *Resour., Conservation Recycling*, vol. 215, Apr. 2025, Art. no. 108146.

[26] S. Wagle, B. Kovacevic, C. Ionescu, D. Walker, M. Jones, L. Carey, R. Takechi, M. Mikov, A. Mooranian, and H. Al-Salami, "Pharmacological and biological study of microencapsulated probucol-secondary bile acid in a diseased mouse model," *Pharmaceutics*, vol. 13, no. 8, p. 1223, Aug. 2021.

[27] A. E. Nilsson, K. Bergman, L. P. Gomez Barrio, E. M. Cabral, and B. K. Tiwari, "Life cycle assessment of a seaweed-based biorefinery concept for production of food, materials, and energy," *Algal Res.*, vol. 65, Jun. 2022, Art. no. 102725.

[28] Y. Wang, H. Wen, M. Wu, X. Liu, H. Yin, W. Qin, X. Zheng, J. He, K. Wei, X. Kong, and S. Liang, "Assessing the environmental footprint of microalgae biofuel production: A comparative analysis of cultivation and harvesting scenarios," *Biochem. Eng. J.*, vol. 213, Jan. 2025, Art. no. 109571.

[29] P. Palensky and D. Dietrich, "Demand side management: Demand response, intelligent energy systems, and smart loads," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 381–388, Aug. 2011.

[30] F. Brahman, M. Honarmand, and S. Jadid, "Optimal electrical and thermal energy management of a residential energy hub, integrating demand response and energy storage system," *Energy Buildings*, vol. 90, pp. 65–75, Mar. 2015.

[31] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[32] B. L. Risteska Stojkoska and K. V. Trivodaliev, "A review of Internet of Things for smart home: Challenges and solutions," *J. Cleaner Prod.*, vol. 140, pp. 1454–1464, Jan. 2017.

[33] A. R. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. Alikarar, "A smart home energy management system using IoT and big data analytics approach," *IEEE Trans. Consum. Electron.*, vol. 63, no. 4, pp. 426–434, Nov. 2017.

[34] B. Menin, "Innovative technologies for large-scale water production in arid regions: Strategies for sustainable development," *J. Appl. Math. Phys.*, vol. 12, no. 7, pp. 2506–2558, 2024.

[35] *Blink*. Accessed: Feb. 25, 2025. [Online]. Available: https://blinkforhome.com

[36] M. Fowler and J. Lewis. (2014). *Microservices*. [Online]. Available: https://Martinfowler.com/articles/microservices.html

[37] *Docker*. Accessed: Oct. 15, 2024. [Online]. Available: https://www.docker.com

[38] R. T. Fielding, "Architectural styles and the design of network-based software architectures," University of California, Irvine, 2000.

[39] *Message Queuing Telemetry Transport (MQTT)*. Accessed: Oct. 17, 2024. [Online]. Available: https://mqtt.org

[40] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM Comput. Surveys*, vol. 35, no. 2, pp. 114–131, Jun. 2003.

[41] T. Dierks and E. Rescorla, "The transport layer security (TLS) protocol version 1.2," RFC Editor, Tech. Rep. 5246, 2008. [Online]. Available:https://www.rfc-editor.org/info/rfc5246

[42] *Certbot*. Accessed: Oct. 8, 2024. [Online]. Available: https://certbot.eff.org/

[43] K. Shingala, "JSON Web token (JWT) based client authentication in message queuing telemetry transport (MQTT)," 2019, *arXiv:1903.02895*.

[44] F. A. Shodiq, R. R. Pahlevi, and P. Sukarno, "Secure MQTT authentication and message exchange methods for IoT constrained device," in *Proc. Int. Conf. Intell. Cybern. Technol. Appl. (ICICyTA)*, Dec. 2021, pp. 70–74.

[45] *PostgreSQL*. Accessed: Oct. 9, 2024. [Online]. Available: https://www.postgresql.org

[46] *InfluxDB*. Accessed: Oct. 9, 2024. [Online]. Available: https://www.influxdata.com

[47] *Http Status Codes*. Accessed: Oct. 15, 2024. [Online]. Available: https://http.dev/status

[48] *Mininet*. Accessed: Sep. 17, 2024. [Online]. Available: http://mininet.org/

[49] V. Gueant. *Iperf*. Accessed: Sep. 17, 2024. [Online]. Available: https://iperf.fr

[50] M. M. Kassem, A. Raman, D. Perino, and N. Sastry, "A browser-side view of starlink connectivity," in *Proc. 22nd ACM Internet Meas. Conf.*, Oct. 2022, pp. 151–158.

[51] S. Ma, Y. C. Chou, H. Zhao, L. Chen, M. Xiao-qiang, and J. Liu, "Network characteristics of LEO satellite constellations: A starlink-based measurement from end users," in *Proc. IEEE Conf. Comput. Commun.*, May 2023, pp. 1–10.

[52] S. Ma, Y. C. Chou, M. Zhang, H. Fang, H. Zhao, J. Liu, and W. I. Atlas, "LEO satellite network access in the wild: Potentials, experiences, and challenges," *IEEE Netw.*, vol. 38, no. 6, pp. 396–403, Nov. 2024.

[53] X. Cao and X. Zhang, "SaTCP: Link-layer informed TCP adaptation for highly dynamic LEO satellite networks," in *Proc. IEEE Conf. Comput. Commun.*, May 2023, pp. 1–10.

[54] D. Laniewski, E. Lanfer, B. Meijerink, R. Van Rijswijk-Deij, and N. Aschenbruck, "WetLinks: A large-scale longitudinal starlink dataset with contiguous weather data," in *Proc. 8th Netw. Traffic Meas. Anal. Conf. (TMA)*, May 2024, pp. 1–9.

[55] Starlink. (2025). *SpaceX*. [Online]. Available: https://www.starlink.com

[56] J. McDowell. *Enormous ('MEGA') Satellite Constellations*. Accessed: Feb. 25, 2025. [Online]. Available: https://planet4589.org/space/con/conlist.html

[57] J. Luomala and I. Hakala, "Effects of temperature and humidity on radio signal strength in outdoor wireless sensor networks," *Federated Conf. Comput. Sci. Inf. Syst. (FedCSIS)*, pp. 1247–1255, Oct. 2015.

[58] E. P. Management. (2010). *Impact of Weather on Smart Wireless Networks*. [Online]. Available: https://www.emerson.com/documents/automation/white-paper-impact-of-weather-on-smart-wireless-networks-en-42650.pdf

[59] K. Matsumoto, N. Sakikawa, and T. Miyata, "Thermo-responsive gels that absorb moisture and ooze water," *Nature Commun.*, vol. 9, no. 1, p. 2315, Jun. 2018.

[60] W. Wang, Q. Pan, Z. Xing, X. Liu, Y. Dai, R. Wang, and T. Ge, "Viability of a practical multicyclic sorption-based water harvester with improved water yield," *Water Res.*, vol. 211, Mar. 2022, Art. no. 118029.

[61] M. Wang, T. Sun, D. Wan, M. Dai, S. Ling, J. Wang, Y. Liu, Y. Fang, S. Xu, J. Yeo, H. Yu, S. Liu, Q. Wang, J. Li, Y. Yang, Z. Fan, and W. Chen, "Solar-powered nanostructured biopolymer hygroscopic aerogels for atmospheric water harvesting," *Nano Energy*, vol. 80, Feb. 2021, Art. no. 105569.

[62] M. Ejeian, A. Entezari, and R. Z. Wang, "Solar powered atmospheric water harvesting with enhanced LiCl /MgSO4/ACF composite," *Appl. Thermal Eng.*, vol. 176, Jul. 2020, Art. no. 115396.

[63] A. Entezari, M. Ejeian, and R. Wang, "Modifying water sorption properties with polymer additives for atmospheric water harvesting applications," *Appl. Thermal Eng.*, vol. 161, Oct. 2019, Art. no. 114109.

[64] H. Maher, T. H. Rupam, K. A. Rocky, R. Bassiouny, and B. B. Saha, "Silica gel-MIL 100(Fe) composite adsorbents for ultra-low heat-driven atmospheric water harvester," *Energy*, vol. 238, Jan. 2022, Art. no. 121741.

[65] H. Park, I. Haechler, G. Schnoering, M. D. Ponte, T. M. Schutzius, and D. Poulikakos, "Enhanced atmospheric water harvesting with sunlight-activated sorption ratcheting," *ACS Appl. Mater. Interfaces*, vol. 14, no. 1, pp. 2237–2245, Jan. 2022.

[66] F. Zhao, X. Zhou, Y. Liu, Y. Shi, Y. Dai, and G. Yu, "Super moisture-absorbent gels for all-weather atmospheric water harvesting," *Adv. Mater.*, vol. 31, no. 10, Mar. 2019, Art. no. 1806446.

[67] J. Xu, T. Li, T. Yan, S. Wu, M. Wu, J. Chao, X. Huo, P. Wang, and R. Wang, "Ultrahigh solar-driven atmospheric water production enabled by scalable rapid-cycling water harvester with vertically aligned nanocomposite sorbent," *Energy Environ. Sci.*, vol. 14, no. 11, pp. 5979–5994, 2021.

[68] M. Wu, R. Li, Y. Shi, M. Altunkaya, S. Aleid, C. Zhang, W. Wang, and P. Wang, "Metal- and halide-free, solid-state polymeric water vapor sorbents for efficient water-sorption-driven cooling and atmospheric water harvesting," *Mater. Horizons*, vol. 8, no. 5, pp. 1518–1527, 2021.

[69] R. Li, Y. Shi, M. Wu, S. Hong, and P. Wang, "Photovoltaic panel cooling by atmospheric water sorption–evaporation cycle," *Nature Sustainability*, vol. 3, no. 8, pp. 636–643, May 2020.

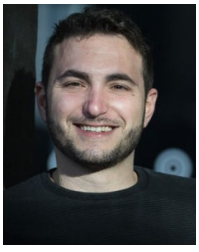
[70] G. Yilmaz, F. L. Meng, W. Lu, J. Abed, C. K. N. Peh, M. Gao, E. H. Sargent, and G. W. Ho, "Autonomous atmospheric water seeping MOF matrix," *Sci. Adv.*, vol. 6, no. 42, p. 8605, Oct. 2020.

[71] P. A. Kallenberger and M. Fröba, "Water harvesting from air with a hygroscopic salt in a hydrogel–derived matrix," *Commun. Chem.*, vol. 1, no. 1, p. 28, May 2018.
[72] A. Entezari, M. Ejeian, and R. Wang, "Super atmospheric water harvesting hydrogel with alginate chains modified with binary salts," *ACS Mater. Lett.*, vol. 2, no. 5, pp. 471–477, May 2020.
[73] H. Lu, W. Shi, J. H. Zhang, A. C. Chen, W. Guan, C. Lei, J. R. Greer, S. V. Boriskina, and G. Yu, "Tailoring the desorption behavior of hygroscopic gels for atmospheric water harvesting in arid climates," *Adv. Mater.*, vol. 34, no. 37, Sep. 2022, Art. no. 2205344.
[74] R. Li, Y. Shi, M. Wu, S. Hong, and P. Wang, "Improving atmospheric water production yield: Enabling multiple water harvesting cycles with nano sorbent," *Nano Energy*, vol. 67, Jan. 2020, Art. no. 104255.
[75] P. A. Kallenberger, K. Posern, K. Linnow, F. J. Brieler, M. Steiger, and M. Fröba, "Alginate-derived salt/polymer composites for thermochemical heat storage," *Adv. Sustain. Syst.*, vol. 2, no. 7, Jul. 2018, Art. no. 1700160.
[76] J. Y. Wang, J. Y. Liu, R. Z. Wang, and L. W. Wang, "Experimental research of composite solid sorbents for fresh water production driven by solar energy," *Appl. Thermal Eng.*, vol. 121, pp. 941–950, Jul. 2017.
[77] A. Permyakova, S. Wang, E. Courbon, F. Nouar, N. Heymans, P. D'Ans, N. Barrier, P. Billemont, G. De Weireld, N. Steunou, M. Frère, and C. Serre, "Design of salt–metal organic framework composites for seasonal heat storage applications," *J. Mater. Chem. A*, vol. 5, no. 25, pp. 12889–12898, 2017.
[78] C. Lei, Y. Guo, W. Guan, H. Lu, W. Shi, and G. Yu, "Polyzwitterionic hydrogels for efficient atmospheric water harvesting," *Angew. Chem.*, vol. 134, no. 13, Mar. 2022, Art. no. 202200271.
[79] J. Williams, "RabbitMQ in Action: distributed messaging for everyone," Simon and Schuster, New York, NY, USA, Tech. Rep., 2012.
[80] T. Yang, L. Zhao, W. Li, and A. Y. Zomaya, "Reinforcement learning in sustainable energy and electric systems: A survey," *Annu. Rev. Control*, vol. 49, pp. 145–163, 2020.
[81] L. P. Kaelbling, M. L. Littman, and A. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, pp. 237–285, May 1996.
[82] E. Mocanu, D. C. Mocanu, P. H. Nguyen, A. Liotta, M. E. Webber, M. Gibescu, and J. G. Slootweg, "On-line building energy optimization using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3698–3708, Jul. 2019.
[83] I. D. Zarzà, J. D. Curtò, and C. T. Calafate, "Decentralized planning of platoons in road transport using reinforcement learning," in *Proc. IEEE 43rd Int. Conf. Distrib. Comput. Syst. Workshops (ICDCSW)*, Jul. 2023, pp. 133–138.
[84] Shristy and K. Kumar, "Reinforcement learning-based strategies for achieving long-term sustainability in IoT," in *Proc. Int. Conf. Integr. Circuits, Commun., Comput. Syst. (ICIC3S)*, Jun. 2024, pp. 1–6.
[85] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

FRANCESCO GAGGINI received the B.S. degree in biomedical engineering and the M.Sc. degree in mechatronics engineering from the Politecnico di Torino, Italy. He is currently a Research Fellow with the Politecnico di Torino. His research interests include studying and developing solutions for cloud platforms.

RAFAEL NATALIO FONTANA CRESPO (Member, IEEE) received the M.Sc. degree in mechatronics engineering from the Politecnico di Torino, in 2022, where he is currently pursuing the Ph.D. degree in computer engineering. His main research interests include the Internet of Things, smart city, cloud technologies, and the design and optimization of machine learning methodologies for the prediction of time-series data in a smart city scenario.

MATTEO CALÒ received the master's degree in mechanical engineering from the Politecnico di Torino, where he is currently pursuing the Ph.D. degree with the Energy Department. He is a Research and Development Engineer with Sorption Technologies.



VINCENZO MARIA GENTILE received the M.Sc. degree in energy and nuclear engineering and the Ph.D. degree in energetics from the Politecnico di Torino, in 2015 and 2020, respectively. In 2022, he was a Visiting Postdoctoral Research Associate with Princeton University. He is currently an Assistant Professor (RTDa) with the Politecnico di Torino. His research interests include sorption-based technologies for air dehumidification and atmospheric water harvesting, air quality and ventilation low-cost sensors, and personal comfort environmental systems technologies, with a focus on the IoT integration and application.

ALBERTO MACII (Senior Member, IEEE) received the Laurea and Ph.D. degrees in computer engineering from the Politecnico di Torino, Turin, Italy. He is currently a Full Professor of computer engineering with the Politecnico di Torino. His research interests include the design of electronic digital circuits and systems, with a particular emphasis on low-power consumption aspects. In the field above, he has authored around 180 scientific publications.

EDOARDO PATTI (Member, IEEE) received the M.Sc. and Ph.D. degrees in computer engineering from the Politecnico di Torino, in 2010 and 2014, respectively. From 2014 to 2015, he held an academic visiting position with The University of Manchester. He is currently an Associate Professor with the Politecnico di Torino. His research interests include ubiquitous computing, the Internet of Things, smart systems and cities, and software architectures with a particular emphasis on infrastructure for ambient intelligence, software solutions for simulating and optimizing energy systems, and software solutions for energy data visualization to increase user awareness.