energies

MDPI

Article

An Alertness-Adjustable Cloud/Fog IoT Solution for Timely Environmental Monitoring Based on Wildfire Risk Forecasting

Athanasios Tsipis [1,*], Asterios Papamichail [1], Ioannis Angelis [1], George Koufoudakis [1], Georgios Tsoumanis [2] and Konstantinos Oikonomou [1]

[1] Department of Informatics, Ionian University, 49100 Corfu, Greece; aspapa@ionio.gr (A.P.); iangelis@ionio.gr (I.A.); gkoufoud@ionio.gr (G.K.); okon@ionio.gr (K.O.)
[2] Department of Informatics and Telecommunications, University of Ioannina, 45110 Arta, Greece; gtsoum@uoi.gr
* Correspondence: atsipis@ionio.gr; Tel.: +30-26610-87734

Received:15 June 2020; Accepted:14 July 2020; Published: 17 July 2020

Abstract: Internet of Things (IoT) appliances, especially those realized through wireless sensor networks (WSNs), have been a dominant subject for heavy research in the environmental and agricultural sectors. To address the ever-increasing demands for real-time monitoring and sufficiently handle the growing volumes of raw data, the cloud/fog computing paradigm is deemed a highly promising solution. This paper presents a WSN-based IoT system that seamlessly integrates all aforementioned technologies, having at its core the cloud/fog hybrid network architecture. The system was intensively validated using a demo prototype in the Ionian University facilities, focusing on response time, an important metric of future smart applications. Further, the developed prototype is able to autonomously adjust its sensing behavior based on the criticality of the prevailing environmental conditions, regarding one of the most notable climate hazards, wildfires. Extensive experimentation verified its efficiency and reported on its alertness and highly conforming characteristics considering the use-case scenario of Corfu Island's 2019 fire risk severity. In all presented cases, it is shown that through fog leveraging it is feasible to contrive significant delay reduction, with high precision and throughput, whilst controlling the energy consumption levels. Finally, a user-driven web interface is highlighted to accompany the system; it is capable of augmenting the data curation and visualization, and offering real-time wildfire risk forecasting based on Chandler's burning index scoring.

Keywords: environmental monitoring; precision agriculture; Internet of Things; wireless sensor networks; cloud/fog computing; fire risk forecasting; Chandler burning index; wildfires

1. Introduction

There has been a recent spike of research activity in precision agriculture and environmental sustainability [1]. One popular direction, focuses on the advances of *cloud computing* enablers [2] and *wireless sensor networks* (WSNs) [3]. These systems embed pioneering wireless technologies, such as ZigBee [4,5], to monitor field conditions in secure and credible ways [6].

The current trend, however, especially when considering the wide proliferation of the *Internet-of-Things* (IoT) applications [7,8], and the opportunities that they bring [9], is to employ a *cloud/fog computing* environment [10–12], offering computing, networking, and storage support near the end user, minimizing response times and greatly improving operational capacity and scalability. Cloud servers (typically located in large data centers) are expected to have increased computational capabilities, whereas *fog devices* located in close proximity to the end users will obtain similar, though less-in-power attributes [13],

and transmute them into suitable candidates for offloading cloud elastic resources and alleviating the communication overhead and traffic burden.

1.1. Challenges and Motivation

Despite the turn towards these information communication technologies (ICT), current practices focus heavily on standalone and trivial data logger systems, whose main tasks are data acquisition, regarding specific environmental facets, from spatially distributed sensors [14]. These initiatives, although useful, lack sophistication and do not take advantage of the full potential brought about by the assimilation of cloud/fog IoT platforms.

That being said, the design, implementation, and large-scale installation of a fully functional IoT platform will provide the involved parties with instant decision-making models, yielding an analytical understanding of natural systems. When combined with other visualization and knowledge tools, such as area maps or hazard scale metrics, they can become highly effective actuators for adaptive field administration, targeted interventions, and personalized notification procedures. Nevertheless, their programming is considered a devious process that requires expertise and extensive know-how, since in most cases it relies on custom-designed equipment and non-standard hardware solutions [15]. Therefore, the procurement of a complete design methodology across all system elements is imperative and the main motivation behind the current work, which attempts to address these issues with the proposition of a complete field-to-stakeholder IoT solution.

Meanwhile, the challenges that arise are numerous, especially when it comes to monitoring catastrophic events in nature [16], e.g., floods [17], earthquake activity [18], and volcanic eruptions [19]; or biological dangers to the plantation itself, e.g., vegetation deceases and pest infestations, which deeply distort the health of the farms, forests, and wild-lands, thereby requiring clear and fast tracking of the agents that cause them, and a continuous stream of data regarding the conditions that drive their spreading. For example, consider the work found in [20], which, much like the study presented here, reports on the development of a WSN-based IoT system, coupled with cloud/fog liturgies, to address time-sensitive agents that affect the health of olive groves.

Still, one of the major challenging environmental hazards to confront is the wildfire, mainly due to the large number of different variables affecting this complex phenomenon, which can wreak havoc on vast areas of land [21]. With that said, the main causal factors remain high temperature and low relative humidity, especially in prolonged drought during summer seasons, which is a common sight in regions such as the Mediterranean basin [22]. However, in conjunction with heterogeneous geographic and micro-climate habitats, such as the ones found in the island regions of Greece, their appearance becomes increasingly hard to predict. In such circumstances, in order to offer targeted fire suppression techniques, the WSNs must be deployed in various landscapes with potentially diversified prevailing conditions (e.g., under different altitudes, wind conditions, temperature/humidity levels, etc.), making the system complex and hard to manage. Consequently, the procurement of an ergonomic, adaptable, and easy-to-use IoT system, able to promptly manage data from various input sources, along with suitable visualization, forecasting, and decision-support functionality, is of the utmost importance for understanding habitat inter-dependencies and a key element of the proposed system in the current work.

Lastly, most existing systems for environmental monitoring or fire prediction/detection require significant computational resources and do not incorporate self-acting operational conformation towards achieving increased precision and energy conservative automation. To the best of the authors' knowledge, this is the first complete end-to-end IoT implementation that attempts to tackle these issues based on local decision-making processes at the fog layer.

1.2. Contribution

The fundamental premise of this paper lies in the presentation of a three-layered cloud/fog computing architecture, suitable for facilitating smart agricultural applications, especially those related

to wildfire monitoring, and to propose a low-cost, WSN-based IoT system that seamlessly embeds the proposed logic. The presented architecture follows the main directions of the cloud/fog computing paradigm [10] based on the general framework found in [12]. In particular, Guardo et al. [12] revealed how the cloud/fog hybrid architecture can be effectively utilized in agriculture and they evaluated it using a prototype of ten nodes. In the current work, the basic principles are similar with an emphasis on the special case of wildfires; hence, a prototype system of 25 sensing nodes, forming six (6) distinct WSNs, each equipped with its own sink node, which is in turn connected to a fog device and then to the remote cloud server, is assumed for evaluation purposes.

The prototype has been developed in the facilities of the Ionian University and is able to autonomously adjust its operations to mirror the necessity for efficient wildfire alertness and lower power consumption. To demonstrate its potential, the prototype was put to the test in a controlled laboratory environment, with the most important evaluation metric being response time, which is considered vital for smart applications of the future. Experimental results showcase how such an architecture can indeed improve precision, by effectively reducing the average response time across all used platforms, with parallel energy efficiency, and high accuracy and throughput rates.

To further validate the criticality-adaptable behavior, the prototype was calibrated to deal with one of the most imminent threats to rural and outdoor environments, i.e., wildfires. Greece is known for suffering from heavy fire activity [23]. In fact, the annually burned area exceeds 245,000 acres of land, and so the experiment has great value for future farming in the area. As a case-study scenario, the Island of Corfu was considered for the firefighting period of 2019, to showcase the architecture's conforming character under different fire risk ignition severity degrees. Experimentation under these conditions showed great promise, highlighted the role of fog computing in dealing with such extreme phenomena, and verified the flexibility of the hardware components used.

Moreover, to better visualize and manage the systems' outputs, a user-driven graphical user interface (GUI) has also been developed to accompany the prototype system and assist the involved parties in the decision-making process, named the *Fog-assisted Environmental Monitoring System*, or F.E.MO.S. in short. That being said, F.E.MO.S. provides users with the ability to oversee the whole sensing process as it enfolds with suitable data curation and visualization. In fact, based on the two environmental sensed parameters (i.e., temperature and relative humidity), it is capable of objectively assessing the fire ignition risk, based on the popular "Chandler burning index" (CBI) [24], and generating an indication of the fire risk severity, subsequently allowing for targeted countermeasures that will mitigate the hazard. In this direction, automated notification alerts are generated to instantly mobilize the authorities to take appropriate mitigation actions.

The main contributing factors of the current paper can be summarized in the next bulletpoints.

1. A robust three-layered cloud/fog computing architecture for environmental monitoring, capable of dynamically conforming its sensing functionality to meet stringent latency requirements and the needs for energy conservation, and high accuracy and throughput.
2. A thorough presentation of its data flow and operations, starting from the initialization of the field WSNs and reaching up to the remote cloud infrastructure, in order to contextualize the steps undertaken from data acquisition to the creation of the appropriate response analysis.
3. The design, analysis, and development of a proof-of-concept prototype, mirroring the given architecture and utilizing state-of-art and low-cost hardware modules for transparent interactions.
4. Its performance evaluation primarily via the response time metric, which is crucial for time-sensitive agricultural applications of the future, especially those keeping track of wildfire activity.
5. The experimentation with real fire risk data considering the fire fighting season of 2019 for Corfu Island, which demonstrates how the considered approach can be effectively utilized to deal with such phenomena and showcases its alertness-adjustable character.
6. The implementation of an accompanying user-friendly web application to monitor the system's behavior and data curation and acquire real-time information relating to the monitored fields'

health, including CBI-based fire risk severity forecasting along with the autonomous generation of appropriate notification alerts to actuate fast mobilization and countermeasures.

The aim here is on timely environmental monitoring, especially regarding wildfire ignition prediction and early detection; nonetheless, the novelty lies with the considered cloud/fog IoT solution that can be utilized for a wide range of time-sensitive agricultural applications with simple system modifications. Since its functionality is easily adjustable to network alterations or ecosystem changes, it can be easily customized and expanded to map its activity to the farmers' occasional needs and demands regarding other smart agriculture and forestry applications.

The remainder of the present paper is organized as follows. In Section 2 necessary background concepts are summarized. The proposed hybrid cloud/fog computing architecture, networking principles, and data flow methodology are underscored in Section 3. The evaluation process is presented in Section 4, while the appliance experiment for the case study of wildfires is showcased in Section 5. Finally, Section 6 concludes the paper and outlines directions for future work.

2. Literature Background

The current section presents related work regarding IoT systems and WSNs in the environmental/agricultural monitoring and fire prediction/detection/protection sectors.

2.1. Internet of Things and Wireless Sensor Networks

For both existing and upcoming applications, given the standardization process of the emerging Fifth-Generation (5G) of mobile communications [25], one of the major aspects that requires careful consideration is the support of multiple devices with parallel real-time processing of large volumes of data [26]. This is essential for the assimilation of IoT decision-making functionalities, with low energy consumption [27] and high accuracy and throughput [28].

An IoT system consists of smart devices that collect, transmit, and act on data they acquire from the environment, without the need for human intervention [15]. Each IoT device transmits its data either directly to the Internet or through a gateway, which are then gathered at a central station for further computation and analysis. However, the centralized paradigm does not always meet the storage and process requirements for the amount of data. This becomes abundantly obvious when considering the diversity of these devices and their lightweight and resource-constrained nature [29].

An alternative road refers to the benefits brought by the integration of *cloud computing* [30]. Moreover, the Internet necessities for low latency and high mobility push the cloud functionality to the edges of the IoT network [31], making way for *fog computing* [32], offering higher cognition and agility [33,34]. That being so, the IoT appliances have made a great leap in the direction of meeting high quality of service guarantees set by the upcoming 5G era, especially when combined with heterogeneous WSN systems [35,36].

WSNs generally consist of battery-powered sensor nodes that are spatially distributed in a wide area, capable of sensing environmental conditions, using powerful processors with low energy consumption [37], a subject of the utmost importance when realizing IoT platforms [38]. The data are often transmitted in a multi-hop manner towards a sink node, which can either store them locally or transmit them to a central location [39], e.g., a collection server.

Until recently, such systems faced many challenges, mainly due to the lack of wide-area connectivity and energy resources, and sometimes harsh environmental conditions. However, modern WSNs, though their inbuilt routing and relay capabilities [40], can quickly adjust to topology changes, allowing their large-scale deployment, even in areas where the battery replenishment may not always be feasible. A popular WSN configuration nowadays embeds Arduino boards [41] and utilizes the ZigBee module [5].

2.2. Related Research in the Agricultural/Environmental Monitoring Sector

Precision agriculture and smart farming [42] are considered two of the most rapidly evolving sciences of the twentieth century and major pillars for boosting productivity and economic growth [43]. Ergo, modern research turns to the previous ICT solutions and their seamless migration towards a multidisciplinary model [11] to support these sectors [28,44]. In fact, innovative enablers and wireless technologies (like SigFox [45], LoRa [46], NB-IoT [47], GSM-IoT [48], and ZigBee [49,50]) have empowered the involved stakeholders with the ability to experiment, manage, and record the dynamics of complex systems [51].

With that in mind, many systems, besides deceasing spread [52] or pest infestations [53], now tackle other aspects for climate protection, as identified in [54]. A recent appliance that meets these guidelines was described in [55]; the paper details a control system for monitoring field data originating from camera and sensor nodes deployed in crops. It then actuated the control devices adhering to threshold constraints relating to specific climate agents.

One of the major WSN obstacles refers to the energy needed to keep the network "alive." Many works attempt to address this barrier. Suárez-Albela et al. [56] identified in this regard the opportunities that arise from smart micro-controllers, such as Orange Pis. Meanwhile in [57], a green WSN node suitable for fog computing platforms named "FROG" was proposed, which introduces proactive power management tools for smart farming. In the present project, the WSN heavy demands on energy are counterbalanced with the use of Arduino boards that have been proven power-efficient (e.g., in [27]), along with ZigBee antennas, which yield significant energy gains [58].

Likewise, the utilization of Raspberry Pis has also revolutionized the data curation process. The work in [59] details a WSN, where information is collected by a Raspberry Pi acting as the base station. In the case of [59], however, the Raspberry Pi was used as a database and web server to manage the data. Similarly, in [60], the overseeing Raspberry Pi was responsible for data acquisition and analysis, while in [61] it created appropriate visualization. Contrarily, in the current study the Raspberry Pis are assigned the role of driving the data processing procedure through fog computing methods.

Diving deeper into the cloud/fog architecture, the works in [62] proposed a scalable fog network architecture to increase coverage and throughput. Emphasis was given to cross-layer channel access and routing, combining inputs generated in multiple networks. However, this work does not involve the use of open-access hardware and software utilities offered by Arduino and ZigBee, respectively, as in the current work. On the other hand, Bin Baharudin et al. [63] showcased the benefits of using Raspberry Pis as fog gateways in a three-layered IoT architecture, similarly to the one incorporated here, using ZigBee for communication. Clearly, ZigBee has been identified as a reliable and affordable standard for smart agriculture realization (e.g., [64]), thereby becoming the central field communication protocol here.

On a different path, the authors in [65] explored agricultural WSNs consisting entirely of Raspberry Pis. In their system, WSN administration was enabled using a GUI, developed in "MATLAB" and installed on the base station's board. Likewise, Zamora-Izquierdo et al. [66] developed an IoT platform for greenhouse automation that allows human operators to configure the individual system components through an "HTML5" interface. Understandably, GUIs are essential for an enhanced end-to-end IoT monitoring solution. This is clearly demonstrated in [14] and highly acknowledged in the current work, which also provides a user-driven GUI for delivering the system outputs in a user-friendly manner.

The majority of WSNs are deployed in uncontrolled areas, making them vulnerable to various types of attacks. Souissi et al. [67] try to tackle this issue by introducing trust in three different levels, namely, the data acquisition level (when the node takes a measurement), the network level (between the nodes of the network), and the data fusion level (during the aggregation and the processing of the measurements). Fortino et al. [68] performed a comparison of the existing architectures by modeling trust in IoT environments. Meanwhile, there is always a possibility that incoming packets

may suffer from data distortions. To detect such occurrences, a field is usually used, called "checksum." Alternatively, Cao et al. [69] attempt to overcome this problem by deducing the measurements' correctness, based on predefined boundaries (e.g., for the temperature the boundary could be set in the $[-5, 40]$ °C). Another problem that WSN-based system operators have to consider is the false data detection. Casado-Vara et al. [70] offered a distributed algorithm that allows the collected temperature data to be self-corrected by the neighboring nodes' readings. In the current project, simple data validation is conducted in the considered fog computing network, where the fog devices evaluate the consistency of the received data packets.

Lastly, a special case of IoT refers to their assimilation for the detection and management of extreme events caused by climate change or other ecological agents. In general, this is a difficult ordeal due to the complex nature and conditions that lead up to their emergence; however, with WSN-based IoT infrastructures, new opportunities have come to light [71], enabling time-critical data curation, while achieving high semantic correlation and efficient risk forecasting [72]. To mention a few, consider the following for extreme weather estimation [73], air pollution detection [74], earthquake prediction [18], flood warning [75], landslide analysis [76], oceanic monitoring [77], etc. Unfortunately, one of the most sensitive and unpredictable hazardous phenomena is the wildfire, which leads to extensive catastrophes around the globe. The next subsections describe to relevant research into these events.

2.3. Related Research in the Wildfire Monitoring Sector

A plethora of systems have been developed to monitor the wildlands for fire threats. Nevertheless, the integration of WSNs for fire regime tracking has not yet been thoroughly explored, although there exists increasing research activity towards this direction (e.g., [78–80]), because compared to conventional methods, such as satellite imagery, which is affected by weather conditions (e.g., clouds), the amount of smoke, the image resolution, etc., WSNs offer faster detection [81].

According to Li et al. [82], these WSNs must possess four key aspects, namely, reactivity, reliability, robustness, and network lifespan elongation. Consequently, exertions are placed in realizing systems with these attributes. For instance, the research by [83] outlines a novel WSN monitoring methodology that adopts a maintenance process to detect temperature anomalies. On the other hand, [84] follows a contiguous approach to the one presented here, to adjust their prototype's sampling and reporting rates based on temperature fluctuations, while the work in [85] provided a fire detection WSN clustering solution with notable reduction in energy consumption. The current study adjusts the response time of the system by using the risk degree that Greece's General Secretariat for Civil Protection (GSCP) publishes every day for the fire fighting season. Additionally, the interval between measurement readings depends on the risk degree, resulting in energy conservation when the fire risk is low.

Obviously, energy conservation is important to prolonging the WSN's life. Having this in mind, the authors of [86] have developed an energy-efficient fire monitoring protocol, i.e., "EFMP," over clustered-based WSNs. Their results showed potential for overall energy consumption reduction, by forming a multi-layer cluster hierarchy depending on forest fire propagation. Despite its effectiveness, the EFMP introduces additional system complexity for dynamically computing the hierarchy. Ergo, alternative communication protocols have also been proposed. For instance, in [87,88] the authors have designed WSN systems, where the data transactions are conducted over the ZigBee protocol. Their experimentation results conduced that ZigBee is a powerful enabler for fire weather monitoring. Similarly, this study uses the ZigBee to realize the communication between the WSNs' nodes.

At the same time, many frameworks sanctioned by different emerging ICT have been also explored, e.g., [89,90], aiming to accomplish reliable field data dissemination, promptly predict wildfire ignitions, and autonomously launch avoidance responses. Similarly, Kaur and Sood [91] proposed a fog-assisted IoT framework for forecasting fire incidents. The framework, close to the current approach, comprises three layers, namely, the data accumulation, fog, and cloud layers. Experimentation exhibited high precision in assessing the susceptibility of the considered habitat

towards wildfire spreading. Note that many frameworks, like the ones presented in [92,93], employ WSNs established by Arduino sensory nodes along with Raspberry Pi gateways, since their highly customizable and flexible modes allow for the fast adaptation to the ever-changing climate conditions that favor the fire regime. That being said, the presented study also employs a similar architecture, using Arduino-powered sensory nodes and Raspberry Pis as intermediary nodes between the WSNs and the remote cloud server.

Moving on, Roque and Padilla [94] developed a prototype using Arduino Uno that communicates through Sigfox. Their prototype is able to detect fires utilizing sensors for temperature and smoke/gas concentrations. Despite their positive performance in terms of response time, their solution is only able to detect fire. In contrast, the current approach is able to also predict fire ignitions, by utilizing temperature and humidity readings. The same applies when considering the case of the LoRaWAN prototype found in [95]. On the other hand, works like [93,96–98] take a different approach through neural networks, resulting in highly accurate fire models via pattern recognition, especially when considering long-term monitoring. These works highlight the efficiency of machine/deep learning approaches that are lightweight enough to run even on Raspberry Pis. Although the current work does not include such algorithms, it is understandable that they can easily be embedded in future work to increase precision and timely warnings. Other studies employ alternative means of detection, e.g., Khan et al. [99] used cameras, while Kalatzis et al. [100] used unmanned aerial vehicles. However, these are more expensive to deploy and on many occasions do not fare well under conditions of heavy rain, fog, snow, mist, etc., and so fail to promptly acknowledge fire incidents.

Whatever the case, the main aim in all the aforementioned works remains the early fire detection in order to launch appropriate remedy countermeasures (like the ones shown in [101,102]), and to execute evacuation procedures (e.g., [103]), while conserving precious energy resources [104] and reducing the end-to-end delay [105], as outlined by [106]. With that said, a close alternative approach to the one followed here is located in [107], which provides a similar IoT platform for the semantic correlation of the generated raw data and their interpretation in terms of imposed fire risk, based on the popular "fire weather index" (FWI) [108]. That particular index is very accurate; however, it requires costly hardware installations. Contrarily, the IoT solution presented here utilizes the CBI [24], which is also precise, but most importantly it relies solely on atmospheric agents, making it a suitable candidate for low-cost implementations. To better contextualize this claim, the following subsection enlists some of the most popular fire danger indexes (FDI) and their requirements.

2.4. Overview of Fire Danger Indexes

Fire outbreaks are affected by different factors related to various physical processes and events. To quantify the fire risk situation, different FDIs have been proposed that combine different quantity environment variables to compute the ignition risk [109].

The McArthur Mark 5 FDI, also known as the "forest fire danger index" (FFDI), is one of the oldest measures, dating back to the 1960s. It is mostly utilized in Australia and is characterized by five ratings, these being low, moderate, high, very high, and extreme. Noble et al. [110] expressed the FFDI as an equitation based on wind speed, relative humidity, temperature, and drought effects.

Contrarily, the FWI is a fire risk model issued by the Canadian Forestry Service in the 1970s [108]. It is affected by four meteorological parameters regarding the noon temperature, noon relative humidity, 24 h precipitation levels, and the maximum speed of the average wind. Its mathematical output procures a number ranging from 0 to 25, which is then suitably mapped to the very low, low, moderate, high, or extreme fire severity risk indications. Although it is considered a highly accurate metric, de Groot et al. [111] showed that it requires significant calibration for the classification thresholds to suit the local weather conditions appropriately.

Compared to FFDI and FWI, a simpler solution has been proposed by Sharples et al. [112], which is correspondingly called the "simple fire danger index" (F). F takes into account the wind velocity, fuel

moisture content, temperature, and relative humidity to divide the fire danger into a five-level scale that ranges from low to extreme. Although simpler than the FFDI and FWI, it has generated mixed performance results that depend on the site of deployment [109].

The aforementioned FDIs are costly when observed from the perspective of equipment utilization. Moreover, they demand a priori the collection of large volumes of data, in order to produce accurate risk classification, especially when considering highly heterogeneous environments such as the Mediterranean Basin. The CBI, however, which was initially proposed by the Chandler et al. [24] in the 1980s, is solely based on weather conditions. Hence, it only requires the air temperature and relative humidity conditions to calculate the immediate fire risk. This makes CBI cost-efficient, since it does not postulate high equipment expenses for the collection and analysis of the field data, rendering it an ideal candidate for low-cost implementations and time-critical applications, as is the case here.

3. System Design and Configuration

Having discussed relevant literature, it is now feasible to venture forth and thoroughly present the proposed IoT solution. Although the scope here focuses on wildfire, the presented architecture is generic and can easily cope and comply with other types of timely environmental monitoring applications.

3.1. The Considered Cloud/Fog Computing Network Architecture

The particular cloud/fog hybrid architecture proposed here (i.e., Figure 1) follows a simple layering model to categorize the available services based on resource availability.

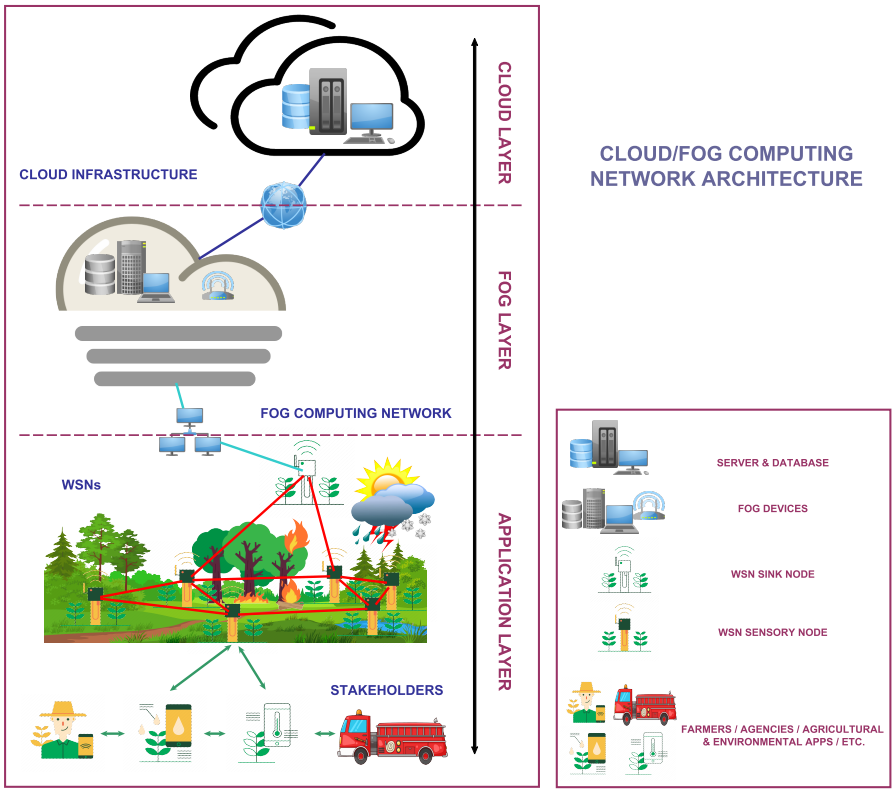


Figure 1. The considered three-layered cloud/fog computing IoT architecture for environmental monitoring, with an emphasis on fire detection applications.

Cloud computing providers commonly employ data centers considering various parameters, such as user proximity and energy consumption [113]. Thus the top layer, i.e., the *cloud layer*, usually

includes a cloud infrastructure, formed by data centers offering amenities and resources, which are dynamically allocated based on the users' demands. These services may include, among other things, storage, networking, and server (computational power, rendering tools, etc.) privileges, as illustrated in Figure 1.

As expected, cloud data centers are usually situated in remote, safe locations and require expensive installations. To procure an affordable and scalable solution, the fog computing paradigm has been proposed. Fog devices, similarly to their cloud-only counterparts, can act as mini data centers that are generally cheaper and highly accessible, extending the service provisioning to the edges of the network [114], forming an intermediary layer, i.e., the *fog layer*. In this way, the computational burden is alleviated and resources are freed, avoiding traffic bottlenecks and increasing the cloud/fog system's overall capacity.

The *application layer*, located at the bottom layer, corresponds here to agricultural/environmental monitoring applications that have already been described in Section 2, including the ones that address the subject of fire handling events, running on the deployed WSNs. The last are comprised of devices called sensory nodes, capable of monitoring environmental agents, such as temperature and humidity, and a sink node, which is tasked with collecting all the field readings and reporting back to the higher layers for further computation. Note that end users also include the various stakeholders that manage the particular WSNs—farmers, firefighting services, researchers, environmental agencies, etc., possibly through various GUIs—different displays, smart or mobile devices, diversified platforms, etc., and are responsible for initiating appropriate countermeasures in cases of emergency (e.g., fire ignition) or high risk (e.g., when temperature and humidity favor the spreading of wildfire).

Clearly, the end users on most occasions can access fog devices in their near vicinity, without the necessity for establishing connections with the remote cloud servers. As a result, communication bandwidth is saved and the proposed architecture can trigger actions with lesser delay, reducing network congestion near the cloud servers if the need arises.

3.2. Hardware and Software Specifications

In the current subsection, a proposed customizable configuration comprising popular hardware micro-controllers for realizing the WSNs and a fog computing network is provided. These rely on market-based low-cost hardware solutions and established software operating systems and formats.

The WSNs are synthesized by sensory nodes, governed by the sink node, which is responsible for data collection. All nodes consist of Arduino micro-controller boards. Arduino [115] is an open-source electronics platform, popular for its hosting capacity and simple configuration, which can easily embed various components and modules, thereby extending its functionality and fast-tracking the creation of prototypes.

In detail, each sensory node is made of an Arduino Uno. To enable wireless connectivity, the board is enhanced by an Arduino wireless Secure Digital (SD) shield, with a Digi XBee ZigBee module [116]. The shield encapsulates three different protocol stacks: IEEE 802.15.4, DigiMesh, and ZigBee. The last one specifies a spacial carrier-sense multiple access with collision avoidance (CSMA/CA) protocol for creating wireless networks from small, low-power digital radio antennas [59]. Using wireless communications, although convenient, opens the door to potential security threats. Fortunately, the XBee makes securing the network a trivial task, since it uses encryption and a secret key, named the "network key," to ensure transmission protection and packet integrity [41].

Regarding the sink node, this differs because of its special properties regarding the collection and management of all the sensed data. For this reason, an Arduino Mega is utilized, which boasts greater memory capacity. The sink node is also enhanced with a wireless SD shield and a similar ZigBee radio module, allowing wireless communication with the Arduino Uno sensors of its assigned WSN.

Dwelling deeper into the WSN connection scheme, according to [117], three basic types of node roles are identified. The *coordinator* is a key structural component during the WSN initiation, tasked with setting up the conditions for its formation, including the selection of the operating channel,

the assignment of a personal area network (PAN) identity (ID), and the establishment of a suitable routing traffic plan. There can only be a single coordinator in each WSN; hence, for the system at hand, the Arduino Mega is assigned this role during the field installation. In comparison, there can exist multiple *routers*, which are intermediate nodes with routing properties, tasked with relaying data from other nodes that cannot directly communicate with the coordinator due to long distances, i.e., the *end devices*. The routers and end devices, for the described system, are the Arduino Uno nodes.

The Arduino Mega is also utilized as a gateway for transmitting the field data to the overseeing fog device, through serial communication. With that said, the fog computing network is formed by Raspberry Pis, each responsible for receiving and analyzing readings from near WSNs and then relaying the data, if necessary, to the central cloud computing infrastructure using the local area network, in this case the Third-Generation (3G) of mobile communication connectivity. Unlike the Arduino devices, the Raspberry Pis have increased computational power that can be upgraded with cloud elastic resources, by offloading cloud demands in close proximity to the WSNs [118]. Additionally, they are able to seamlessly connect/interact with the former, thereby making them an ideal device for hosting fog-related processes.

Of course, there are many alternatives that can be employed to realize both the WSNs and the fog computing network. For instance, consider the Banana Pis or Orange Pis [119]. Although these are valid counter-proposals, the Arduino and Raspberry Pi devices were selected due to their extensive documentation, low-cost peripherals, highly configurable nature, and facile intercommunication. Similarly, the XBee modules, although they offer less reliability or range than other wireless antennas, such as the LoRa-based antennas [120], the former were chosen due to their affordable character, ease of programming, and low network and deployment cost, especially when considering large-scale, heterogeneous, and geographically distributed installation sites. In any case, the adoption of an alternative solution is strongly attainable since the elastic and extendable nature of the proposed system allows for such modifications with ease. That being said, for convenience and to better contextualize the economic aspects of the utilized hardware, Appendix A describes in more detail the specifications of the incorporated micro-controllers, whereas Appendix B compares wireless communication technologies.

3.3. Data Flow and Processing Methodology

The first phase of setting up the system, as depicted in Figure 2, involves the WSN initialization stage, where all nodes are deployed and then join and form the WSN [41]. This stage refers only to the WSN since it is assumed that the fog computing network and cloud computing infrastructure are already fired up and running, awaiting new raw data.

The first to power-on is the ZigBee coordinator, which in turn initializes the remaining WSN hardware. This is accomplished by initiating the protocol stack and performing an energy detection scan [4] to obtain a list of secure potential channels. Thereafter, it will continue with an active scan, where it chooses a free channel and enables the joining process, during which a routing plan is established. Moreover, due to WSNs being vulnerable to exogenous environmental factors, the clocks of the individual nodes experience de-synchronization. Hence, the proposed system encompasses a synchronization period during which the nodes calculate their time offset and synchronize their clocks accordingly, using as a reference point the coordinator's time.

Upon completion of the above phase, the various nodes start sensing environmental data. This occurs at regular intervals using their attached sensory modules. For the project at hand, which is targeted at fire alertness, the sensors track temperature and humidity, both variables critical to wildfire spreading. Conveniently, both come packed together in the DHT22 digital temperature/humidity sensor (also named AM2302 and depicted in Figure A1).

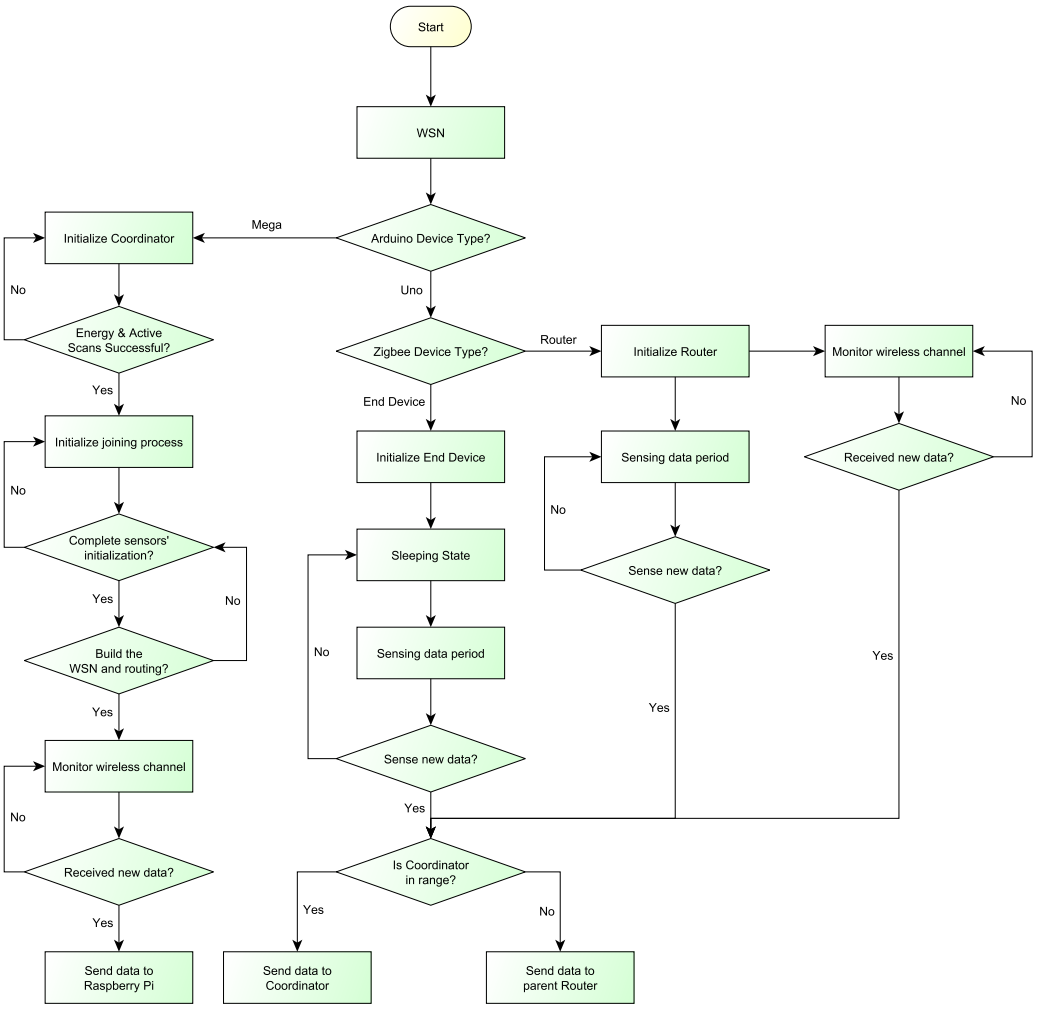


Figure 2. Flow chart of all processes taking place in the WSNs.

After a sensing period ends, the new data are transmitted towards the sink node over the established routing paths. If the nodes are too far away to convey the data directly by themselves (i.e., the coordinator is out of range), then they forward the data to their parent router. This check is repeated on all nodes along the given route until the data reach the coordinator. For this process to be reliable, the routers and the coordinator must continuously monitor the wireless channel for incoming signals.

Upon data reception, the Arduino Mega adopts a new role, forming a gateway for connecting the WSN with the fog computing network, comprised of several Raspberry Pis. For simplicity, these are connected with their assigned Arduino Mega devices through serial connections. The Raspberry Pis, as reported in Figure 3, listen to their serial channel for incoming data so as to initiate a connection. Note that the Raspberry Pis may potentially communicate with one or more Arduino Mega devices, collecting, in the second case, data from multiple WSNs.

The moment the Raspberry Pis receive a stream of data, they decide if they will perform the necessary calculations themselves or forward them to the central server infrastructure. This depends on the adopted implementation road, since the increased capabilities of the Raspberry Pis allow for such modifications in their configuration to support customized solutions. For the case study at hand, a simple distributed approach was considered; ergo, the fog devices' processing behavior was developed as follows. For every set of data packets received, a percentage of them would be computed on the spot by the fog computing network and the rest would be forwarded to the cloud computing infrastructure. In this way, the trafficking and computational load at the cloud

servers would remain low and balanced by the fog devices, and so the system's overall response time would decrease.

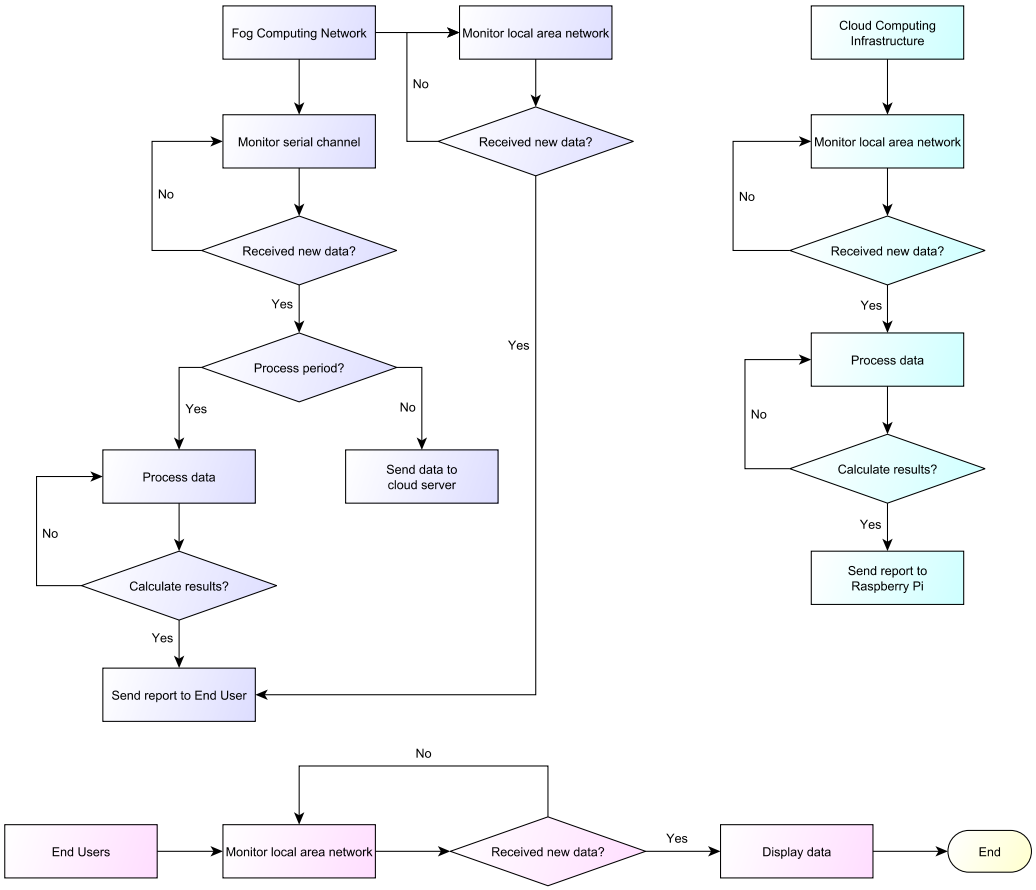


Figure 3. Flow chart of all processes occurring in the cloud/fog side of the proposed system.

The last can be further explained considering that, as the data packets are transmitted along the route, they are queued inside intermediate node buffers. The same occurs when the data reach the cloud server. However, if the data arriving rate exceeds the processing rate of the server or the data gets too many in number and too big in size, this will lead to the creation of network bottlenecks. In turn, the response time of the system might increase dramatically, especially in cases where there exist spikes and fluctuations in the sensed data, like in the cases of sudden fire spreading.

At any rate, when the sensed data are collected, computed, and analyzed, a system decision is generated and a response report is created. In the first case (where the computation takes place in the fog), the response is directly transmitted through the local area network to the end users' devices. Meanwhile, if the computation occurs in the cloud, the generated report travels backward towards the corresponding fog devices and then to the end users that administrate the corresponding WSNs.

4. Evaluation

The current section validates the effectiveness of the presented cloud/fog architecture and evaluates its adaptability in terms of reliable monitoring. To do so, an experimental prototype was designed in a closed control laboratory environment in the facilities of the Ionian University, located on Corfu Island.

4.1. Experimentation Setup

The designed experimental prototype comprised 25 Arduino Uno Rev 3 nodes, six (6) Arduino Mega 2560 nodes, and three (3) Raspberry Pi 3 Model B fog nodes (for more details about these models consult the Appendix A). In particular, six WSNs were considered, each containing a various number of Arduino Uno devices and an Arduino Mega coordinator/sink node, along with their arsenal of sensory and communication modules. Each sink node was linked with a serial cable to a Raspberry Pi. Following, the three fog devices communicated via 3G with the central cloud infrastructure, comprised of a virtual machine (VM) server with its accompanying database and storage, located in a different building of the Ionian University's facilities, as depicted in Figure 4. The exact setup of the WSNs in conjunction with the corresponding fog devices is outlined in Table 1. In fact, the number of sensory nodes varies among the WSNs in order to create diversified traffic load case scenarios (from low to high) at the respected fog devices and obtain a more objective assessment of their conforming capabilities.

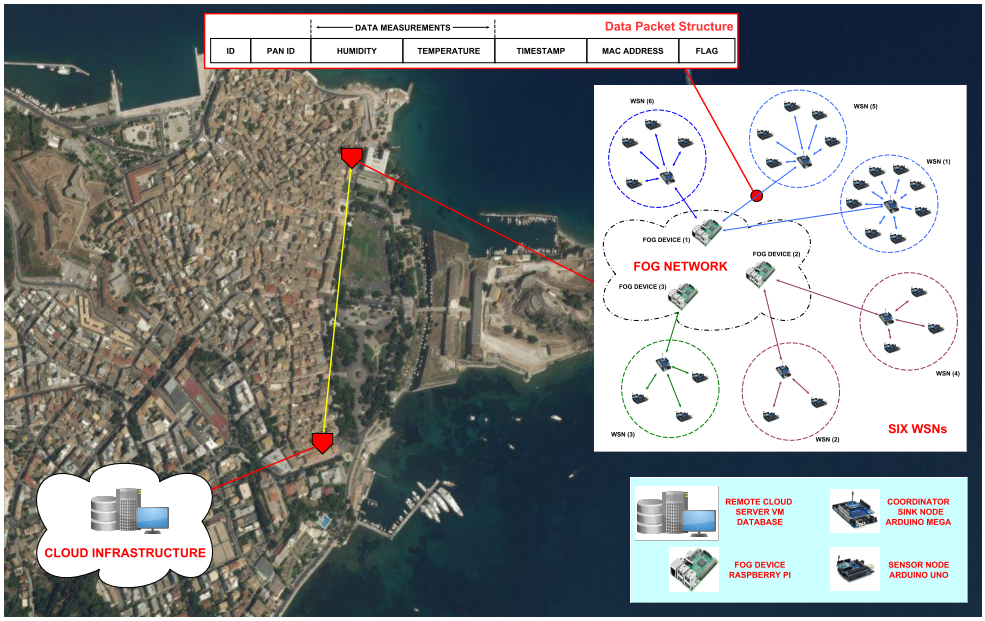


Figure 4. The experimental prototype system setup, programmed and installed in different locations of the Ionian University's facilities in Corfu Town.

Table 1. Deployment setup of the WSNs and fog network

WSN ID	Number of Sensory Nodes	Fog Device
One (1)	Eight (8)	One (1)
Two (2)	Two (2)	Two (2)
Three (3)	Three (3)	Three (3)
Four (4)	Three (3)	Two (2)
Five (5)	Five (5)	One (1)
Six (6)	Four (4)	One (1)

For simplicity, it was assumed that the WSNs are composed of their coordinator and all sensory nodes were assigned the role of routers in the constructed PANs, leading to more scalability. The sensed measurements along with the appropriate PAN ID were encapsulated inside data packets at regular intervals with a $+/-5$ s of random increment, in order to force diversity in the packet generation rate and minimize the appearance of packet collisions. In addition, the Arduino Uno timestamped each packet and encapsulated their own medium access control (MAC) address. Finally, the packets were marked with a unique ID and flagged based on the processing location (i.e., the fog or the cloud). The complete structure of a data packet is depicted in the upper part of Figure 4.

Each Arduino Mega stored the incoming information inside its SD memory card and then forwarded the data packet its assigned Raspberry Pi. The latter stochastically handled the incoming sensed data, based on a probability function, deciding whether to process the data locally or convey the data to the remote cloud server. More information regarding the server's resources can be found in Appendix A.

Both fog and cloud performed the same processing by extracting the sensed data and calculating the average values for each discrete variable. However, if the process took place directly at the fog network the data packet was flagged with a zero (0), whereas if it was carried out at the cloud, it was flagged with a one (1). Each data packet was processed once based on its unique identifier. Then the procured data (e.g., average temperature and humidity) were used to overwrite the corresponding fields in the data packet, while at the same time being stored in separate database logs and compatible formats for future reference.

At this point, it is important to state that data probity validation occurs for each data packet received to ensure that no data corruption exists and to exclude values that are miscalculated. Initially, the sensory and sink nodes' antennas perform a checksum on the data packets to verify their composition. Then, upon reception of a new data packet, the Raspberry Pis also check the structural integrity of the packet. First, based on the ID and PAN ID fields they cross-validate the origin of the data packet based on their assigned WSNs and previous observations. Then the measurement fields (i.e., the HUMIDITY and temperature) are assessed for their correctness, e.g., they must humidity be float numbers with a defined number of digits. Additionally, to verify that packets with information losses or distorted data are not taken into consideration, the fog devices inspect the format of all fields that must meet certain criteria; e.g., the ID and MAC ADDRESS fields must always process a specific sequence of characters. Data packets that infringe these conditions are automatically discarded. By embedding such mechanics, the fog devices are able to filter the data and offer basic data validation. Of course, there are other more sophisticated ways to verify the data integrity and offer reliability. For example, future work will include the cross-validation of the data based on pre-defined thresholds regarding the expected environmental and seasonal conditions in the monitored lands. For instance, during the summer pic a lower bound of 15 °C could be enforced and data packets that violate this constraint will be automatically dropped. Correspondingly, an upper bound of 50 °C could be placed and when violated the fog devices will assess the credibility of the reading based on the measurements previously obtained by the other sensory nodes in the immediate neighborhood. If multiple readings are received in sequence that record extreme heat, especially when originated in multiple sources, then the alarm will be raised appropriately.

4.2. Experimentation Results

For the presented prototype, response time is one of the most important performance metrics for timely monitoring and accurate correlation, and is computed as the *Round Trip Time* (*RTT*), i.e., the time that elapses between the moment a sensory node sends a new data packet and the moment it receives a reply from the cloud/fog system. In other words, the *RTT* is the sum of network and processing delays, which can be better observed in Figure 5, where the arrows depict the transmissions and the gray boxes illustrate the process time at each device. Note that Figure 5 represents the simplest scenario where the sensor is situated in the sink node's immediate neighborhood, i.e., one hop away.

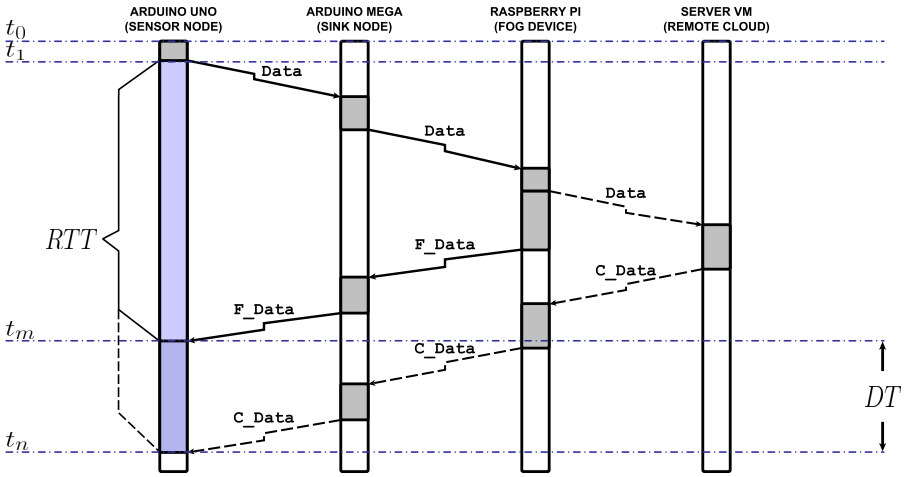


Figure 5. Calculation of the RTT, for both cloud and fog computing scenarios. The gray boxes symbolize the processing times along the route, whereas the arrows the transmissions that occur.

In detail, at time t_0 the Arduino Uno generates a data packet, hereafter called `Data`, containing the newly obtained measurements and their timestamp, which will be forwarded to the Arduino Mega at t_1, when the channel becomes available. Upon reception from the Arduino Mega, the `Data` undergoes some processing. When this is over, they are sent to the overseeing Raspberry Pi. Once there, the fog device decides with a probability P as to where the data process will eventuate, i.e, locally or remotely.

In the former case, the Raspberry Pi decodes the `Data`, processes the information, and extracts the average values, computing all relative information gathered up until that moment from the specific WSN. The average values are encapsulated into a new data packet, named `F_Data`, replacing the corresponding values of the received `Data`, but keeping intact the values of the ID, $TIMESTAMP$, and $MAC\_ADDRESS$ fields. Then the `F_Data` is flagged accordingly, by replacing the value of the $FLAG$ field, in order for the system to acknowledge the fog network as the processing location. Next, the `F_Data` begins its journey back towards the sensor node that generated the measurements. At the coordinator it stays a period equal to the time it takes for the Arduino Mega to read the $MAC\_ADDRESS$ and decide the route the `F_Data` must take to reach the respective Arduino Uno. Finally, it is forwarded to the last, which reads the $TIMESTAMP$ field and compares it to its current clock, calculating in this way the total RTT. Note again that in order for this procedure to be accurate there exists a simple clock synchronization method enforced at regular intervals.

In contrast, during the latter case, the Raspberry Pi conveys the data packet to the remote cloud server VM. The server performs similar actions to the ones described earlier. However, the information is now encoded into a data packet, named `C_Data`, and flagged with a value indicating the corresponding location, i.e., the cloud infrastructure. Similarly, the `C_Data` then travels backward the network route until it reaches the appropriate Arduino Uno, which, at that point calculates the RTT.

In Figure 5, the fog processing scenario is depicted through straight-arrow transmissions, whereas the cloud processing scenario is illustrated through dotted-arrow transmissions. Notice that the `F_Data` reaches the Arduino Uno in t_m, while the `C_Data` arrives at t_n, where $t_n = t_m + DT$. The DT is the difference between the two calculated RTTs, and is viewed as a measure estimate of the system's performance in terms of response time for multiple values of P.

Following the last assumption, Figure 6 encapsulates the results of the approach by presenting the average achieved RTT, as a function of P, for different experimental runs with varying time intervals regarding the data packet generation rate. The error bars demonstrate the upper and lower 95% confidence intervals. Clearly, for all depicted cases as P increases the RTT decreases. In fact, starting from the extreme scenario, where all processing occurs in the remote cloud server, i.e., for $P = 0$, the average $RTT \simeq 1160$ ms. Then, for every increment in P, more operations are performed in the fog

network and so the RTT steadily drops, reaching its minimum value for the opposing extreme scenario, where all processing takes place directly in the fog network, i.e., for $P = 1$ the average $RTT \simeq 1080$ ms. Any spikes in the depicted plots are attributed to cloud/fog processing pulsations and delays posed by the ZigBee CSMA/CA protocol or other network intermediates during experimentation. What is more, no clear tendency can be derived by the abatement in the time interval period between sensory readings, eliciting that the fog network successfully coped with the incoming traffic in all circumstances.

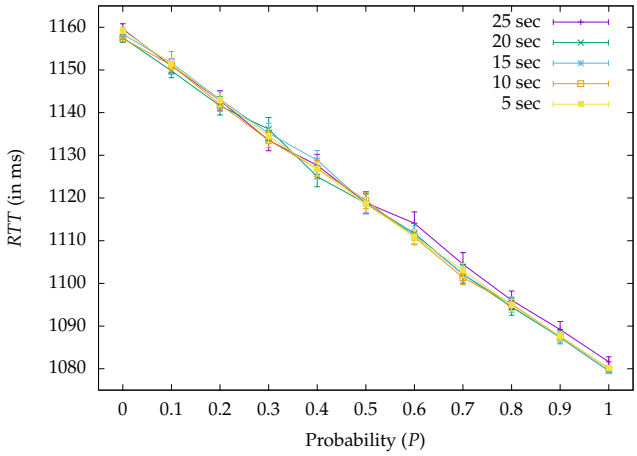
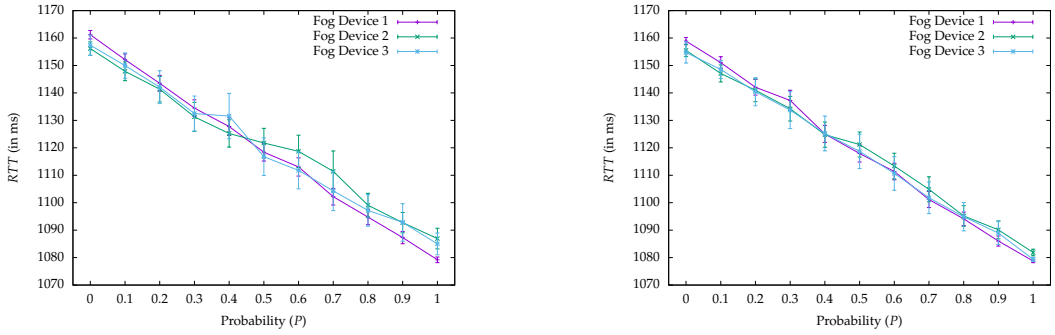


Figure 6. Experimental results of the system's total mean RTT as a function of the probability P for different data packet generation intervals, i.e., for 25 s, 20 s, 15 s, 10 s, and 5 s. The error bars represent the upper and lower 95% confidence intervals.

To further highlight the latency reduction behavior, Figure 7 decomposes each depicted case of Figure 6, by showcasing the mean RTT from the viewpoint of the involved fog devices. Even at a lower level of illustration the behavior still holds, proving once more the response time reduction across all fog devices as probability P is increased. To complete the system's RTT deconstruction Figure 8 offers a microscopic view of the system's RTT decaying behavior by encasing the same results from the scope of the deployed WSNs. In all six depicted cases the particular decreasing behavior is verified once more.

Finally, it is noteworthy that the number of WSNs and eventually of sensory nodes (according to Table 1) does not impact the responsiveness. Actually, all three Raspberry Pis successfully managed to handle the incoming traffic load, and thus it is further assumed that the RTT is affected by other processes running at the background during the experiment at the cloud or the fog devices.



(**a**) (**b**)

Figure 7. *Cont.*

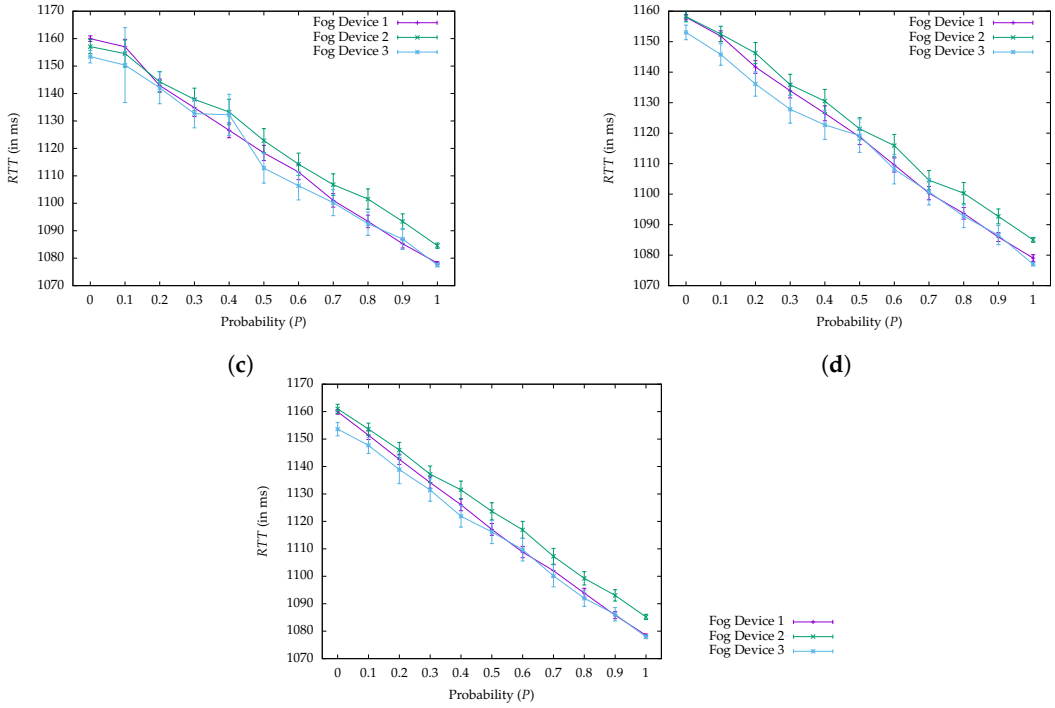


Figure 7. Experimental results of the system's total mean *RTT* value, through the viewpoint of the involved fog devices. The error bars correspond to the 95% upper and lower confidence intervals. (**a**) Interval: 25,000 (in ms). (**b**) Interval: 20,000 (in ms). (**c**) Interval: 15,000 (in ms). (**d**) Interval: 10,000 (in ms). (**e**) Interval: 5000 (in ms).

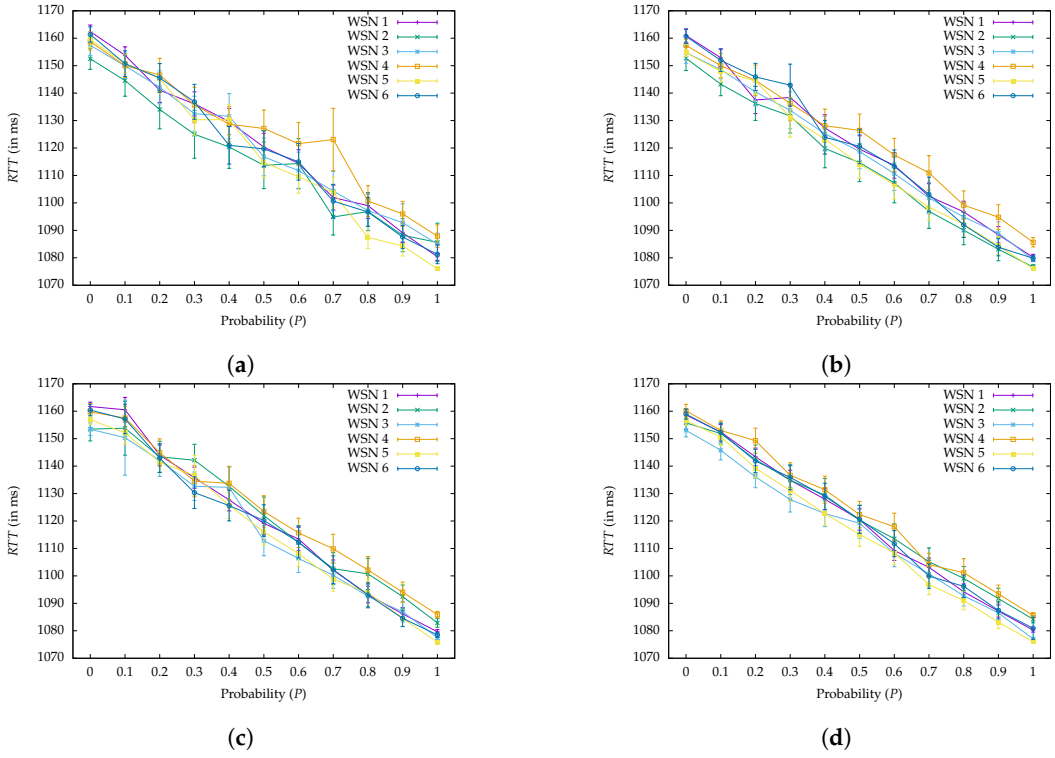
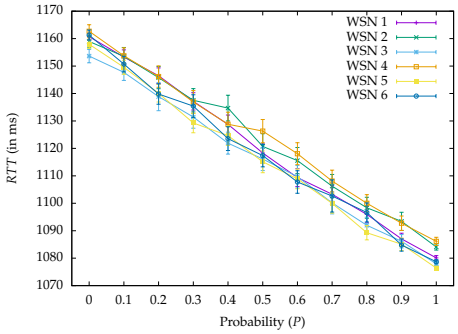


Figure 8. *Cont.*



(**e**)

Figure 8. Experimental results regarding the mean *RTT*, through the scope of the deployed WSNs. The error bars refer to the upper and lower 95% confidence intervals. (**a**) Interval: 25,000 (in ms). (**b**) Interval: 20,000 (in ms). (**c**) Interval: 15,000 (in ms). (**d**) Interval: 10,000 (in ms). (**e**) Interval: 5000 (in ms).

5. System Conformation Based on Wildfire Risk Forecasting

To demonstrate the potential of the presented solution, which does not impose the irreducible network delays associated with conventional environmental IoT systems, whilst remaining energy efficient, the prototype is put against one of Greece's most hazardous states of emergency, i.e., the wildfires, to determine how well it can react and adapt its behavior to deal with a potential crisis. Additionally, a user-friendly designed GUI for data visualization and fire risk forecasting is presented.

5.1. The Case of Greece's Wildfires

According to Greece's Fire Brigade's (GFB) official statistics, 8006 forest fires were recorded during the year 2018 alone [121], while for 2019 the number is much higher, reaching 9502 fire fronts. The year 2007, however, was reportedly the worst of the last thirty years, since the country, during the summer, was hit by three consecutive heat waves (over 46 °C each), which coupled with strong winds and low relative humidity (around 9%), resulted in forest fires breaking out all over Greece. In fact, according to the European Space Agency (ESA), Greece has witnessed more wildfire activity during the summer of 2007 than other European countries have experienced over the last decade [122]. This led to 11,996 forest fires by the end of the year, burning over 675,000 acres of land, which was a European record for that period. To put this in more perspective, just on Corfu Island, an area of barely 236 square miles, in the year 2019 alone, 171 wildfires were recorded, burning around 137.5 acres of land, a great percentage of which consisted of farming lands and forests [121].

To deal with the fire peril and increase the degree of readiness, the GSCP during the firefighting period (from June the 1st to October the 31st), issues a daily map depicting the fire risk degree regarding all regions of Greece for the following day. In this way, the GSCP warns the corresponding authorities to prepare for the possibility of environment-threatening fire events in their respective regions and the citizens to stay on alert. The map is color-coded based on a five risk rate scale, starting from green, which indicates low risk, and up to red, which raises alarm to the highest possible level.

Having this information available is important because it allows for targeted adjustments to the proposed system's configuration, in order to maximize the effectiveness in detecting such catastrophic events. Consequently, for the purposes of the current experiment the cloud/fog system was programmed to autonomously adapt its behavior based on the risk degree scale regarding Corfu Island, where the experiment takes place. Figure 9 describes this procedure in depth.

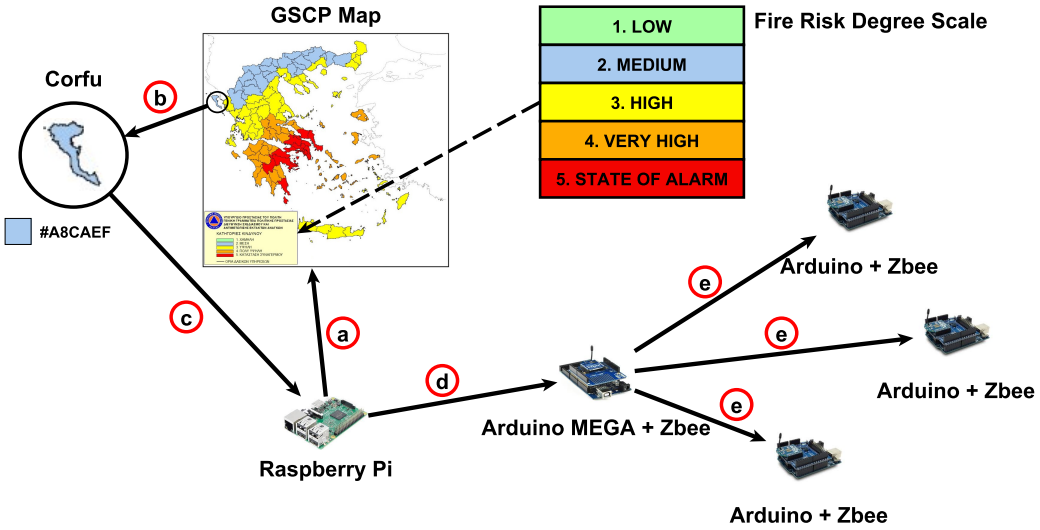


Figure 9. Steps for retrieving the risk degree and relaying the information to a WSN.

Once per day, during the early hours, when the map has been already published, the Raspberry Pis connect to the website of the GSCP (*step a* in Figure 9), which holds the information regarding the updated map [123]. Then they extract the specific risk degree regarding Corfu Island. To do so, the fog devices recognize the particular pixels that form Corfu on the map and retrieve their RGB color-code. For example, in Figure 9, where the fire risk for Corfu is set to "medium," the RGB color-code is determined as "#A8CAE" (i.e., *step b*). By reading the particular value, the Raspberry Pis immediately assess, according to the fire risk degree scale, the risk to be at level two (i.e., *step c*). After obtaining this value, they relay the information to their assigned Arduino Mega devices (i.e., *step d*), which in turn broadcast the information to their WSN, as illustrated in *step e* of Figure 9.

By the end of this procedure, all WSNs' nodes have received the fire risk degree and can modify their operation accordingly. Specifically, in order to be energy efficient, during the experiment, the sensors mapped their sampling behavior based on the risk degree that was fed into the WSNs by the fog devices. As such, when the sensory nodes received a degree of one (1), the interval period that intervened between two successive sensing periods was stretched to 25 s, with the goal of preserving battery energy and ensuring that communication bandwidth is not wasted, since there was no need for heavy monitoring at the time. For every degree that was added to the fire risk scale, however, the particular interval was autonomously reduced by 5 s. In this way, the generation rate of the packets was sped up as the risk got higher and continuous monitoring was required.

Table 2 captures the described behavior by presenting the interval period for every fire risk degree along with the probability P that drives the fog processing. In order to keep the main cloud infrastructure informed about the general ongoing situation in the fields, even in cases where the fire risk degree is five, a small percentage of the generated packages are still transmitted to the cloud server.

Table 2. System behavior under the different degrees of the Fire Risk.

Fire Risk Degree	Interval Period	Value of P
One (1)	25 s	5%
Two (2)	20 s	25%
Three (3)	15 s	50%
Four (4)	10 s	75%
Five (5)	5 s	95%

To test its performance the IoT system was instructed to retrieve each fire risk degree regarding Corfu, for the firefighting period of 2019. Ergo, 153 values corresponding to the five (5) months, were

extracted and used as an input. The upper part of Figure 10 illustrates these values organized per month. Note that only once the risk achieved a degree of four (4), while no days existed that reached the value of five (5). Having these values stored, made it practicable to use them as system parameters. Thus, each day was represented with twenty (20) min of experimentation time.

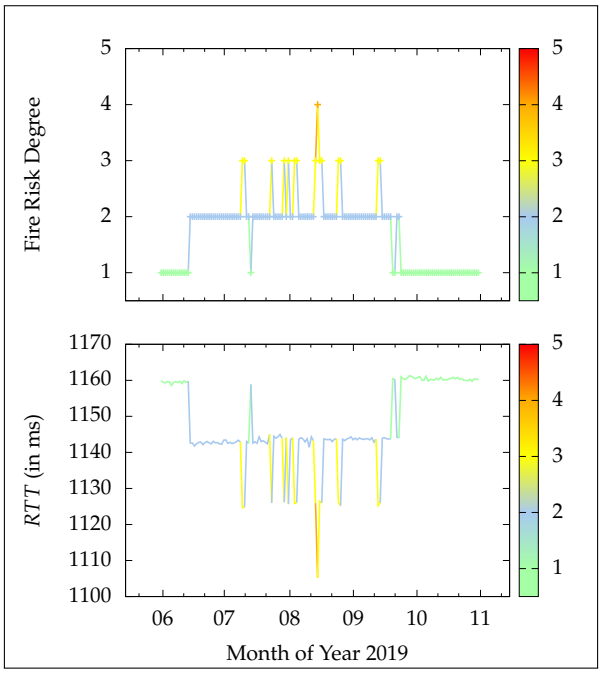


Figure 10. The fire risk degree (upper part) versus the average *RTT* in ms (bottom part), per day during the firefighting season of 2019.

The system's alertness encapsulates perfectly the change in the fire risk degrees as time goes on. Specifically, the system's obtained average *RTT* over all six deployed WSNs, is adjusted autonomously by pushing/pulling functionality towards/from the fog computing network, without human intervention. The result of this procedure is shown in the bottom part of Figure 10. In fact, the average *RTT* is reduced as the degrees climb the fire risk scale, enabling faster field-health analysis. In fact, it reaches its minimum value on the day that reported a fire risk degree of four (4), i.e., $RTT \leq 1110$ ms, during the pic of the firefighting season. In comparison, when the fire risk is one (1), most notably recorded during the beginning and ending of the firefighting season, the mean *RTT* is maximized, i.e., $RTT \simeq 1160$ ms, since most processing activity befalls the cloud server and there is no need for heavy field monitoring.

Regarding the traffic load, i.e., the data packet generation rate here, the system dynamically adapts its sampling periods, as validated in the upper part of Figure 11. Ergo, for low risk, the data packet generation rate is slowed down to save communication bandwidth and conserve precious energy, untimely elongating the WSNs' lifespan. The last is perfectly mirrored in the bottom part of Figure 12, where the energy consumption is visualized per day (i.e., for the 20 minutes experimental cycle here). This equates to the sum of the energy spent by all sensory modules, i.e., the temperature/humidity sensor to generate a new reading or not, plus the energy consumed by all antennas for their two alternate states, i.e., for transmission or idle respectively. Specifically, the DHT22 when in standby mode uses 50 uA while in reading mode 1.5 mA [124]. As for the XBee antenna, it consumes 31 mA in idle state and 120 mA in transmission state, respectively [125]. Likewise, as the degrees rise, the demand for higher precision impels the system to prioritize the need for continuous wildfire monitoring, thereby increasing the packet generation and transmission rates. In particular, for fire risk equal to one (1) the number of data packets per day is clustered around 900, translating to roughly

2.567 kJ of energy consumption. Contrarily, for the day reporting a degree of four (4), the packet number is launched to almost double and close to 1800, leading to increased energy consumption, which reports a value of around 2.572 kJ.

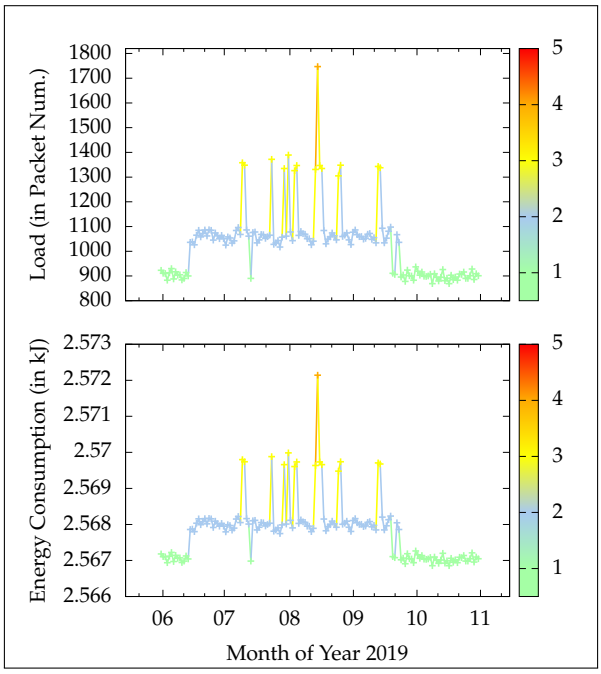


Figure 11. The traffic load, i.e., the number of transmitted packets (upper part), versus the corresponding energy consumption (bottom part) in kJ, per day of the firefighting season of 2019.

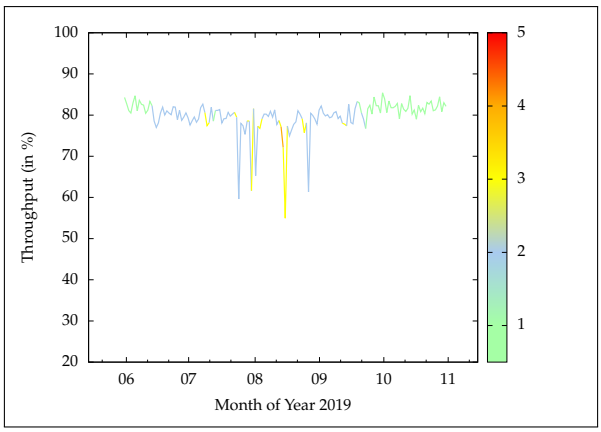


Figure 12. The achieved throughput in percent, per day during the firefighting season of 2019.

Although the aforementioned differences, especially in regards to energy consumption, at first glance might not seem crucial, it is understandable that under a real-world field deployment, where the number of sensory devices could grow to hundreds or even thousands of nodes and the time period will be expanded to real days, then the number of transmissions will greatly increase. However, the distributed fog processing methods adopted here will significantly reduce the *RTT* when compared to cloud-only approaches, especially considering the long distances that data packets will have to transverse to reach the remote cloud server from the field sites. In conjunction with multi-hop topologies and CSMA/CA re-transmissions and carrier sensing waiting times, it is obvious that the gap between the energy consumption during a low-risk day and an extreme-risk day will also

be vastly larger. Moreover, in a real-world scenario, many of the WSNs' nodes will be assigned the role of end devices according to the ZigBee standard. Ergo, they will be permitted to enter "sleep" mode, by powering-off their antennas between transmissions to save additional battery usage. The network's lifetime can then be extended until the point of the first router or sink failure, which can jeopardize the WSN's connectivity, cutting access to the sink or fog network respectively. Nevertheless, this can also be addressed with energy-replenishment and harvesting tools, like solar-panel power-banks. Besides, ZigBee embeds re-routing methods that can promptly establish a new routing plan among the routers. With that said, it is possible to estimate the lifetime, by computing the average energy consumption per router/sink for the different fire risk degrees (using past observations), so as to predict the possible time of failure based on their residual battery energy from the previous day.

To verify that the rise in data packet traffic load does not sever the system's credibility, Figure 12 depicts the total throughput for each day of the experiment, based on the number of successfully retrieved data packet *ID*s. In most circumstances, even during heavy network traffic, the achievable throughput fluctuates around 80%. A few exceptions (i.e., throughput drops) are clearly attributed to the underlying CSMA/CA protocol. This finding is crucial because it enables the almost real-time forecasting/detection of a fire incident, with minimum loss of data, allowing for accurate monitoring and early notification of the authorities in case of fire ignition, in order to launch appropriate countermeasures that will mitigate the danger. Additionally, it is clear that the system can effectively cope with the computational burden, achieving high network performance, and retaining its accountability intact.

5.2. F.E.MO.S.: The Fog-Assisted Environmental Monitoring System

A key factor for truly providing an end-to-end IoT solution is the ability to visualize the data in a user-friendly manner through a proper GUI. Besides, as already mentioned the whole purpose of the considered monitoring cloud/fog system is to offer end users in the application layer the opportunity to quickly access and make sense of the field data, informing them about the ongoing status and alerting them regarding potential environmental hazards, like probable wildfire ignitions. The necessity becomes even more prevalent when the monitored lands are geographically scattered, with potentially diversified exposure to elements, altitudes, weather conditions, etc., and so the sensed data may vary substantially among the WSNs, resulting in complexities that jeopardize the decision making process.

To this end, a simple, yet accurate and extendable, web application has been developed for visualizing the streams of data arriving from either the cloud or the fog, suitably named the "**F**og-assisted **E**nvironmental **Mo**nitoring **S**ystem," or *F.E.MO.S.* in short. Both the server and the Raspberry Pis feed into F.E.MO.S. In real-time the captured environmental readings from the WSNs, and then F.E.MO.S. generates proper data-gram charts of their behavior in relation to time. Figure 13 depicts the web GUI dashboard of the F.E.MO.S. The dashboard is separated into panels, which are as follows.

Starting from the top central panel, namely, "Data Visualization," this is where the generated field data are plotted and custom-made figures are created regarding the chosen environmental parameters in relation to time. For the demo prototype system, the possible parameters to choose from are temperature (in °C) and relative humidity (in %). By default, F.E.MO.S. will plot the mean values of both, computed by the accumulated data from all installed WSN nodes, to procure an overall overview of the prevailing weather conditions in the deployment sites during the latest week, as is the case in Figure 13. Clearly, the generated plots are of high resolution and showcase the precision of the sensory nodes. Next to the bottom left corner of the panel, an indication of the current energy consumption (in kJ) is also provided to inform the stakeholders about the nodes' power utilization and help them in cases where energy replenishment is required. This value is updated on an hourly basis.

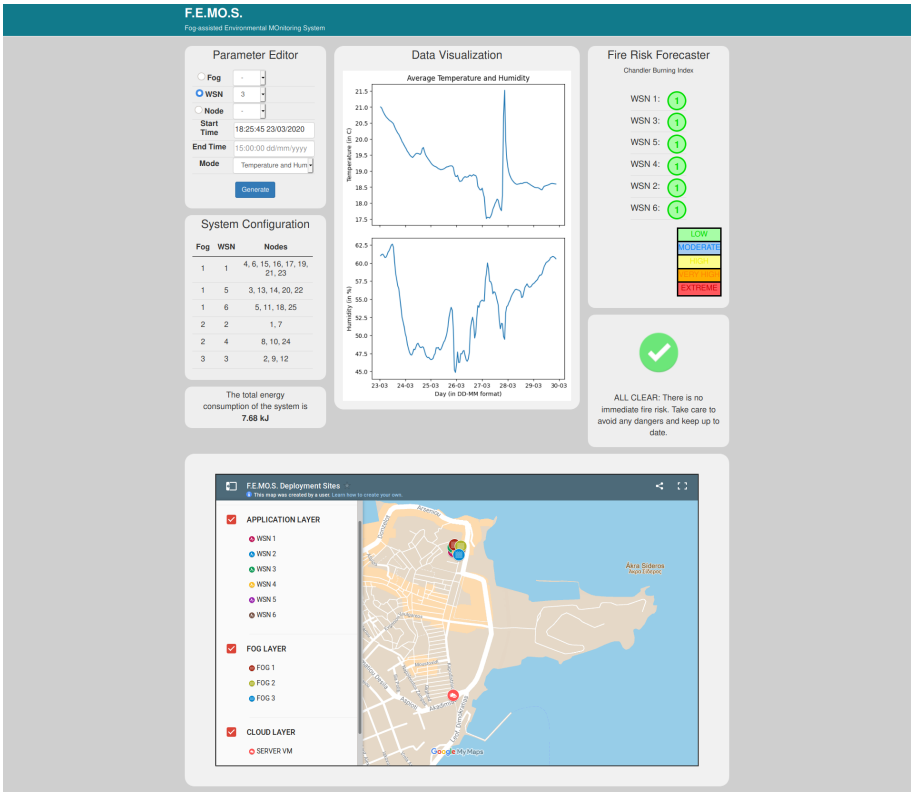


Figure 13. The web GUI dashboard of F.E.MO.S.

To customize the plots the left top panel, namely, "Parameter Editor," enables the users to configure the data visualization. In this respect, they may choose from a plethora of available options, to create targeted filtered queries and generate appropriate plots. The first option determines the type of system entity. There are mainly three types, these being the fog computing network, the deployed WSNs, or the sensor nodes themselves. However, in all three, the user is free to select either an averaged view of all sensory readings or further filter the data source. Ergo, he/she can designate which specific Raspberry Pi, WSN, or sensory node to explore respectively. A complete map of the system setup is provided, in the form of a Table, named "System Configuration," below the parameter editor, so as to assist the user in searching the appropriate device ID. Moreover, to access past measurement logs, a precise time period for data retrieval can be specified, in an hour/date format. Finally, a separate drop-down field is attributed to choosing the desired environmental parameter to populate the data visualization panel. In Figure 13, a simple user input example is shown, just before data generation.

While the aforementioned panels focused only on the presentation of the raw data, it was also considered imperative to augment F.E.MO.S. with cognition behavior, by offering semantic correlation of the readings in terms of wildfire forecasting, and thus highlight its potential in sensitive and timely decision-making procedures. To this end, the right top panel, suitably named the "Fire Risk Forecaster," presents the fire risk in each separate deployed WSN, using a color-coded, five-degree scale. To evaluate the risk, the CBI is utilized here [24], which is based solely on weather conditions. As such, CBI uses the air temperature and relative humidity to calculate in real-time a numerical index of the fire danger at the corresponding WSN sites, according to the following formula:

$$CBI = \frac{((110 - 1.373 \times RH) - 0.54 \times (10.20 - T)) \times (124 \times 10^{-0.0142 \times RH})}{60}, \quad (1)$$

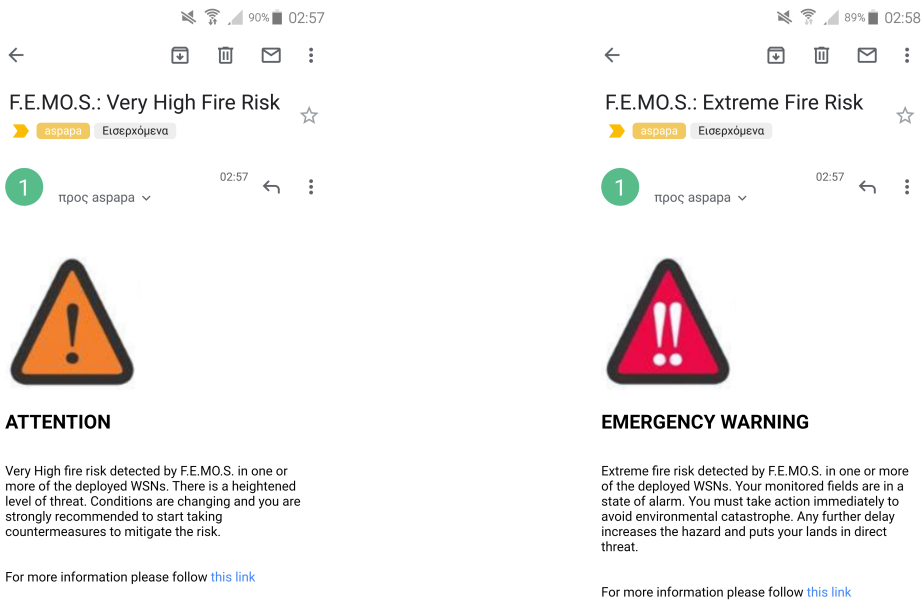
where T is the current atmospheric temperature (in °C) and RH is the current relative humidity (in %). That number is then equated to the fire risk severity of either low, moderate, high, very high, or extreme, and mapped to the F.E.MO.S. color-coded scale mentioned earlier based on Table 3.

Table 3. F.E.MO.S. fire risk forecaster scale

Chandler Burning Index	Label & Color Code	Fire Risk Forecasting Rating
CBI $<$ 50	LOW (Green)	1
50 $\leq$ CBI $<$ 75	MODERATE (Blue)	2
75 $\leq$ CBI $<$ 90	HIGH (Yellow)	3
90 $\leq$ CBI $\leq$ 97.5	VERY HIGH (Orange)	4
CBI $>$ 97.5	EXTREME (Red)	5

Having this information available in real-time is exceptionally important, since it facilitates the dynamic conformation of the cloud/fog system to meet the requirements for greater or lesser environmental monitoring, as explained in Subsection 5.1. Moreover, it enables the early notification of the stakeholders and respective authorities for a possible fire threatening event at the monitored lands, allowing for timely interventions and targeted countermeasures (e.g., the launch of water-spraying mechanisms). Actually, to further increase mobilization, when the fire risk forecaster reports a fire risk rating equal or greater than three (3) at any given WSN (i.e., for CBI $\geq$ 75), F.E.MO.S. automatically generates and sends alerts to the registered email addresses of the responsible parties. An example of two such notifications is shown in Figure 14 for the case where the fire risk forecasting hits the severity score of four (4) and five (5), i.e., for "very high" and "extreme" danger respectively.

Similarly, F.E.MO.S. also provides in a separate panel the corresponding alert indication. In Figure 13, it is observable that all WSNs have a fire risk score of one (1), suitably highlighted with the color green. As a result, the indication is also a green check marker symbol, declaring that the ongoing situation in the WSN deployment sites is safe, and so there is no need for alarm. Contrarily, in Figure 15, an example of how a caution warning alert indication will look like in the case where the fire risk in the WSNs reaches the yellow indication, i.e., "high," is given, by zooming in on the fire risk forecaster and the corresponding notification panel of F.E.MO.S.



(**a**) Very high fire risk (**b**) Extreme fire risk

Figure 14. F.E.MO.S. automated notification alert messages for the cases of (**a**) very high and (**b**) extreme fire risk forecasting.

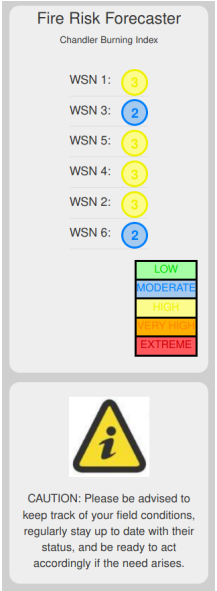


Figure 15. F.E.MO.S. indication regarding the case of a high fire risk scenario in one or more WSNs.

Lastly, for the users' convenience, F.E.MO.S. also provides a separate panel containing a map with all information regarding the marked deployment sites, organized in different layers that follow the considered three-layered system network architecture. The map is scrollable and flexible to allow users to explore the installation sites even when the system entities are situated in geographically different or remote locations, with long distances among them. An instance of the map panel, regarding the deployment sites of the prototype system, can be seen in the bottom part of Figure 13.

6. Conclusions and Future Directions

A hybrid, three-layered system architecture for smart and timely environmental monitoring, embedding affordable IoT and WSN appliances, while in its core following a cloud/fog computing approach, was presented here. The architecture was then extensively studied and its data flow functions were analyzed in depth, starting from the field nodes and up to the moment the information was delivered to the appropriate parties. Furthermore, the current work reported on the design and implementation of a demo prototype that can easily conform its functionality to address critical environmental challenges, based on the described architecture. To this end, a hardware/software solution was also proposed that is affordable and suitable for fast prototyping, while utilizing highly customizable components and controllers capable of networking, communication, measurement sensing, and processing.

Moreover, extensive experimentation was conducted, in controlled laboratory conditions in the facilities of the Ionian University, to evaluate the prototype's performance and highlight its alertness-conforming characteristics. A popular metric for precision environmental monitoring was considered, i.e, the response time. Initial experimentation to investigate the system's elements interactions and seamless adaptability captured the expected behavior and validated its effectiveness in dealing with time-sensitive agricultural and environmental applications. In all cases presented, the average response time was reduced as more operations took place in the intermediate fog computing network. That being said, for the two extreme cases, i.e., in the cloud-only processing and fog-only processing, the average *RTT* values were 1160 ms and 1080 ms respectively, indicating an overall 80 ms improvement across all system elements without exception.

To further highlight its performance, the developed prototype was put to the test under a real case scenario involving one of Greece's most notable environmental hazards, the wildfires. Again the results support the research claims, as it is shown that the system is able to adapt its operation and alertness, sufficiently addressing the problem, based on the five-degree fire risk scale retrieved from the GSCP.

Hence, it was able to reserve communication bandwidth (on average $\simeq$ 900 data packets) and decrease energy consumption, when the fire risk was low, all the while increasing its monitoring precision, effectively raising its sensory readings to almost the double (i.e., $\simeq$ 1800 data packets), when there existed high probability of fire ignition, achieving in most cases an average throughput over 80%.

To contextualize the system outputs and augment the data visualization process, a user-driven web-based GUI was also developed to accompany the prototype and allow users to proactively partake in the monitoring process. F.E.MO.S., as is its name, provides friendly panels to enable the filtering of the raw data and the flexible presentation of the monitored environmental conditions and system entities. Actually, it even goes a step further, by enhancing the decision-making process and offering real-time CBI-based fire risk forecasting, considering the prevailing weather conditions at each separate WSN deployment site. To increase alertness and actuate higher mobilization, targeted alert notifications are generated automatically in cases of emergency, while systematic logs are maintained to successfully correlate the data and allow the stakeholders to efficiently keep track of the status and health of the monitored lands and energy consumption levels of the system modules.

The paper also focused on the prototype's limitations. That being so, the biggest drawback of the approach is the lack of actual large-scale deployment in a real-world environment, which due to the poor network connectivity, power limitations, hostile environmental conditions, multi-hop routing, etc., will certainly affect the credibility and reliability of the sensor readings. However, the system in its current form can act as a proof-of-concept testbed, to fast test various other parameters or technologies prior to field deployment and investigate more deeply their implications, without endangering hardware/software elements. Moreover, it can significantly boost system debugging and offer insights regarding cases where device/module failures or data corruptions occur. The last is considered critical for the successful adoption of the approach; therefore, appropriate actions must also be undertaken to validate the accountability and credibility of the measured readings with the adoption of appropriate filtering and threshold methods. Other constraints, relating to automation, diversified network topologies, sensor clock synchronization, power utilization, and so on, are also taken into consideration during the prototype's design process, through proper software solutions and engineering decisions. The latter, although custom-made, remain generic and flexible, following established practices, and so can be easily modified to fit and include other appliances. Nevertheless, more research must be conducted towards the direction of securing these functionalities and augmenting their operations, even under extreme scenarios, e.g., during an actual fire outbreak. Another subject that raises attention relates to the optimal placement of the sensor nodes. For the case at hand, this was not necessary; however, during field deployment, this aspect will definitely go a long way towards alleviating possible WSN bottlenecks, routing issues, transmission collisions, and ultimately energy consumption. The same applies in the case where a proper subset of sensor nodes (backbone network) is discovered for each WSN, e.g., using a dominating set methodology, to boost information collection and dissemination. Obviously, tackling these challenges and limitations will greatly improve the system's overall performance, especially when considering that field deployment will involve a large number of sensory nodes.

With that being said, future guidelines will explore the system's standardization and its large-scale deployment in outdoor areas, where its activity will be extensively recorded and documented, as well as assessed in comparison with alternative techniques and systems. In fact, through mathematical formulation, it is feasible to identify additional crucial climate variables that lead to the formation of wildfire devastating events, in order to implement targeted field interventions. To this end, substitute fire burning indexes, such as the FWI, will be also researched, and their accuracies will be evaluated. Moreover, with a few tweaks in the code, the considered cloud/fog prototype system and F.E.MO.S. could be easily configured to support additional sensor modalities, and by extension alternative agricultural and environmental applications. In this respect, they will be enriched with extra monitoring tools (e.g., for determining smoke/gas emissions, soil moisture, wind velocity, pH levels, and rain intensity), which among other things, can also target or detect other environmentally

hazardous events, for instance, earthquakes, rain floods, volcanic eruptions, tree deaths, pest infestations, etc. To manage the huge amount of raw data acquired by these additions, machine learning algorithms will be investigated, which coupled with the F.E.MO.S. forecasting metrics will lead to new prediction models, fostering high accuracy and enhanced decision-support mechanics. Furthermore, alternative wireless technologies (e.g., LoRa, NB-IoT, etc.) and hardware controllers (e.g., Orange Pis) will be tested to discover the optimum configuration. Great effort will be put toward providing a complete, low-cost IoT solution, reflecting the daily needs and accomplishing alignment with the expectations of future smart agriculture and environmental protection and preservation.

Author Contributions: Conceptualization, A.T. and A.P.; methodology, A.T., A.P., and I.A.; software, A.P. and G.K.; validation, K.O. and I.A.; formal analysis, A.T. and G.K.; investigation, G.K. and G.T.; resources, I.A. and G.T.; data curation, A.P.; writing—original draft preparation, A.T. and I.A.; writing—review and editing, A.P. and K.O.; visualization, A.T. and G.T.; supervision, A.T.; project administration, K.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments

This work was supported in part by project "A Pilot Wireless Sensor Networks System for Synchronized Monitoring of Climate and Soil Parameters in Olive Groves," (MIS 5007309) which is partially funded by European and National Greek Funds (ESPA) under the Regional Operational Programme "Ionian Islands 2014-2020."

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3G	Third Generation of Wireless Mobile Telecommunications
5G	Fifth Generation of Wireless Mobile Telecommunications
CBI	Chandler Burning Index
CPU	Central Processing Unit
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
ESA	European Space Agency
F	Simple Fire Danger Index
FDI	Fire Danger Index
FFDI	Forest Fire Danger Index
F.E.MO.S.	Fog-Assisted Environmental Monitoring System
FWI	Fire Weather Index
GFB	Greece's Fire Brigade
GSCP	General Secretariat for Civil Protection
GUI	Graphical User Interface
ID	Identity
ICT	Information Communication Technologies
IoT	Internet of Things
MAC	Media Access Control
PAN	Personal Area Network
RAM	Random Access Memory
RTT	Round Trip Time
SD	Secure Digital
VM	Virtual Machine
WSN	Wireless Sensor Network

Appendix A. The Prototype's Utilized Hardware and Software Specifications

The utilized hardware micro-controllers for the design of the experimental cloud/fog IoT prototype include Arduino Uno, Arduino Mega, and Raspberry Pi devices. The current appendix encloses detailed information regarding technical specifications of these technologies, their selected models, and their accompanying modules, in addition to the resources utilized by the remote cloud server VM.

- *Arduino Uno*: In the current project implementation, the considered WSN sensors consist of an Arduino Uno Rev. 3, which is built on top of the Atmel ATmega328P micro-controller. This is in turn enhanced with a Digi XBee-PRO S2C ZigBee module [116] for wireless communication. The sensors were equipped with a DHT22 sensory module which is able to calculate the temperature in the scale of $-40^{\circ}C$ to $80^{\circ}C$, with a $\pm 5^{\circ}C$ inaccuracy, and assess the humidity atmospheric levels in a scale of 0% to 100%, with an accuracy deviation between 2% and 5%.
- *Arduino Mega*: For the programming of the WSNs' sink nodes, an Arduino Mega 2560 micro-controller board was chosen, which is based on the ATmega2560. The sink nodes were augmented with communication capabilities using a wireless SD shield and a Digi XBee-PRO S2C module. They were also equipped with an SD memory card to save logs regarding the incoming readings. Moreover, they serially forwarded the data packets to their overseeing Raspberry Pi at a data rate of 115,200 bps.
- *Raspberry Pi Model B*: The fog devices composing the second hierarchy layer of the system's architecture, correspond to Raspberry Pis 3 Model B. This model was chosen due to its low-cost and low-power consumption attributes and its ability for wireless and serial connectivity. Essentially it is a small computer board that supports a number of different operating systems. For the purposes of current work, the Debian-based Linux operating system, named "Raspbian," was used.
- *Cloud Server VM*: The cloud server runs on a Unix-based VM, with a four-core central processing unit (CPU) and 4 GB of random access memory (RAM), which is part of the Ionian University's central cloud data center infrastructure, capable of high-speed computation and data transmission.

To put the aforementioned technologies in more perspective, Table A1 enlists technical specifications of the three micro-controller boards used, whereas Figure A1 depicts them after their assembly.

Table A1. Technical specifications of the devices used in this paper for the realization of the WSNs and fog computing network.

Specification	Arduino Uno Rev 3 [126]	Arduino Mega 2560 [127]	Raspberry Pi 3 Model B [128]
Microcontroller	ATmega328P	ATmega2560	Broadcom BCM2837 64 bit
Connectivity	-	-	Bluetooth 4.1 Classic/Low Energy, CSI, 10/100 Ethernet, 2.4 GHz 802.11b/g/n wireless
RAM	2 KB SRAM, 32 KB Flash Memory	8 KB SRAM, 256 KB Flash Memory	1GB LPDDR2 (900 MHz)
Pins	14 (of which 6 provide PWM output)	54 (of which 14 provide PWM output)	40-pin GPIO header
CPU	Intel Quark (x86) 16 MHz	Intel Quark (x86) 16 MHz	4 × ARM Cortex-A53, 1.2 GHz
GPU	-	-	Broadcom VideoCore IV @ 250 MHz
MSRP	≃20 €	≃35 €	≃40 €

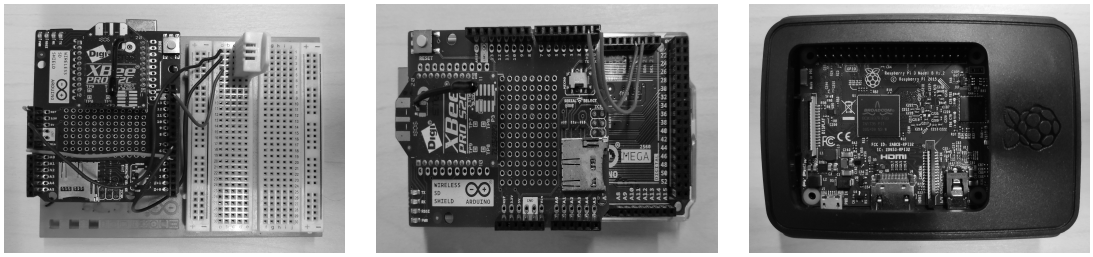


Figure A1. From left to right, the utilized Arduino Uno with the attached Digi XBee-PRO S2C module and DHT22 Temperature and Humidity Sensor, the Arduino Mega with its Digi XBee-PRO S2C module and microSD card slot, and the Raspberry Pi 3 Model B enclosed in a protective case.

Appendix B. Comparison of Existing Wireless Technologies

The current appendix contains the comparisons of alternative wireless technologies in order to showcase the affordable character of the adopted approach in the current work and propose alternative solutions that can easily be incorporated in future system alterations. As such, Table A2 compares various communication technologies and their characteristics, including the utilized ZigBee.

Table A2. Wireless technologies comparison table [25,58,129–133]

Wireless Technology	Range	Security	Deployment Cost	Power Usage	Maximum Data Rate
Zigbee	≤100 m	LOW	LOW	LOW	250 Kbps
LoRa	≤20 Km	HIGH	LOW	LOW	50 Kbps
NB-IoT	≤10 Km	HIGH	HIGH	HIGH	200 Kbps
Sigfox	≤50 Km	HIGH	MEDIUM	MEDIUM	100 Bps
BLuetooth	≤50 m	HIGH	LOW	HIGH	2 Mbps
LTE	≤30 Km	HIGH	MEDIUM	MEDIUM	1 Mbps
Z-Wave	≤100 m	LOW	MEDIUM	LOW	100 Kbps
Weigtless	≤5 km	HIGH	LOW	MEDIUM	100 Kbps

References

1. Yost, M.; Sudduth, K.; Walthall, C.; Kitchen, N. Public-private collaboration toward research, education and innovation opportunities in precision agriculture. *Precis. Agric.* **2019**, *20*, 4–18.
2. Mekala, M.S.; Viswanathan, P. A Survey: Smart agriculture IoT with cloud computing. In Proceedings of the IEEE International Conference on Microelectronic Devices, Circuits and Systems (ICMDCS), Vellore, India, 10–12 August 2017; pp. 1–7.
3. Ojha, T.; Misra, S.; Raghuwanshi, N.S. Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges. *Comput. Electron. Agric.* **2015**, *118*, 66–84.
4. Baronti, P.; Pillai, P.; Chook, V.W.; Chessa, S.; Gotta, A.; Hu, Y.F. Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and ZigBee standards. *Comput. Commun.* **2007**, *30*, 1655–1695.
5. Kalaivani, T.; Allirani, A.; Priya, P. A survey on Zigbee based wireless sensor networks in agriculture. In Proceedings of the IEEE 3rd International Conference on Trendz in Information Sciences & Computing (TISC2011), Chennai, India, 8–9 December 2011; pp. 85–89.
6. Gupta, M.; Abdelsalam, M.; Khorsandroo, S.; Mittal, S. Security and privacy in smart farming: Challenges and opportunities. *IEEE Access* **2020**, *8*, 34564–34584.
7. Lee, I.; Lee, K. The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Bus. Horizons* **2015**, *58*, 431–440.
8. Chiang, M.; Zhang, T. Fog and IoT: An overview of research opportunities. *IEEE Internet Things J.* **2016**, *3*, 854–864.
9. Popović, T.; Latinović, N.; Pešić, A.; Zečević, Ž.; Krstajić, B.; Djukanović, S. Architecting an IoT-enabled platform for precision agriculture and ecological monitoring: A case study. *Comput. Electron. Agric.* **2017**, *140*, 255–265.

10. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog computing and its role in the Internet of Things. In Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, Helsinki, Finland, 17 August 2012; ACM: New York, NY, USA, 2012; pp. 13–16.
11. Channe, H.; Kothari, S.; Kadam, D. Multidisciplinary model for smart agriculture using internet-of-things (IoT), sensors, cloud-computing, mobile-computing & big-data analysis. *Int. J. Comput. Technol. Appl.* **2015**, *6*, 374–382.
12. Guardo, E.; Di Stefano, A.; La Corte, A.; Sapienza, M.; Scatà, M. A Fog Computing-based IoT Framework for Precision Agriculture. *J. Internet Technol.* **2018**, *19*, 1401–1411.
13. Dastjerdi, A.V.; Gupta, H.; Calheiros, R.N.; Ghosh, S.K.; Buyya, R. Fog computing: Principles, architectures, and applications. In *Internet of Things*; Morgan Kaufmann, Elsevier: Amsterdam, The Netherlands, 2016; pp. 61–75.
14. Nundloll, V.; Porter, B.; Blair, G.S.; Emmett, B.; Cosby, J.; Jones, D.L.; Chadwick, D.; Winterbourn, B.; Beattie, P.; Dean, G.; others. The design and deployment of an end-to-end IoT infrastructure for the natural environment. *Future Internet* **2019**, *11*, 129.
15. Sethi, P.; Sarangi, S.R. Internet of things: Architectures, protocols, and applications. *J. Electr. Comput. Eng.* **2017**, *2017*, doi:10.1155/2017/9324035.
16. Ray, P.P.; Mukherjee, M.; Shu, L. Internet of things for disaster management: State-of-the-art and prospects. *IEEE Access* **2017**, *5*, 18818–18835.
17. Visconti, P.; Primiceri, P.; Orlando, C. Solar powered wireless monitoring system of environmental conditions for early flood prediction or optimized irrigation in agriculture. *J. Eng. Appl. Sci.* **2016**, *11*, 4623–4632.
18. Alphonsa, A.; Ravi, G. Earthquake early warning system by IOT using Wireless sensor networks. In Proceedings of the IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 23–25 March 2016; pp. 1201–1205.
19. Awadallah, S.; Moure, D.; Torres-González, P. An Internet of Things (IoT) Application on Volcano Monitoring. *Sensors* **2019**, *19*, 4651.
20. Tsipis, A.; Papamichail, A.; Koufoudakis, G.; Tsoumanis, G.; Polykalas, S.E.; Oikonomou, K. Latency-Adjustable Cloud/Fog Computing Architecture for Time-Sensitive Environmental Monitoring in Olive Groves. *AgriEngineering* **2020**, *2*, 175–205.
21. Meyn, A.; White, P.S.; Buhk, C.; Jentsch, A. Environmental drivers of large, infrequent wildfires: the emerging conceptual model. *Prog. Phys. Geogr.* **2007**, *31*, 287–312.
22. Pausas, J.G.; Llovet, J.; Rodrigo, A.; Vallejo, R. Are wildfires a disaster in the Mediterranean basin?–A review. *Int. J. Wildland Fire* **2009**, *17*, 713–723.
23. Papadopoulos, A.; Paschalidou, A.; Kassomenos, P.; McGregor, G. Investigating the relationship of meteorological/climatological conditions and wildfires in Greece. *Theor. Appl. Climatol.* **2013**, *112*, 113–126.
24. Chandler, C.; Cheney, P.; Thomas, P.; Trabaud, L.; Williams, D.; others. *Fire in forestry. Volume 2. Forest fire management and organization.*; John Wiley & Sons, NY, USA, 1983.
25. Akpakwu, G.A.; Silva, B.J.; Hancke, G.P.; Abu-Mahfouz, A.M. A survey on 5G networks for the Internet of Things: Communication technologies and challenges. *IEEE Access* **2017**, *6*, 3619–3647.
26. Li, S.; Da Xu, L.; Zhao, S. 5G Internet of Things: a survey. *J. Ind. Inf. Integr.* **2018**, *10*, 1–9.
27. Jawad, H.M.; Nordin, R.; Gharghan, S.K.; Jawad, A.M.; Ismail, M. Energy-Efficient Wireless Sensor Networks for Precision Agriculture: A Review. *Sensors* **2017**, *17*, doi:10.3390/s17081781.
28. Tzounis, A.; Katsoulas, N.; Bartzanas, T.; Kittas, C. Internet of Things in agriculture, recent advances and future challenges. *Biosyst. Eng.* **2017**, *164*, 31–48.
29. Yaqoob, I.; Ahmed, E.; Hashem, I.A.T.; Ahmed, A.I.A.; Gani, A.; Imran, M.; Guizani, M. Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges. *IEEE Wirel. Commun.* **2017**, *24*, 10–16.
30. Botta, A.; De Donato, W.; Persico, V.; Pescapé, A. Integration of cloud computing and internet of things: A survey. *Future Gener. Comput. Syst.* **2016**, *56*, 684–700.
31. Yu, W.; Liang, F.; He, X.; Hatcher, W.G.; Lu, C.; Lin, J.; Yang, X. A survey on the edge computing for the Internet of Things. *IEEE Access* **2017**, *6*, 6900–6919.
32. Puliafito, C.; Mingozzi, E.; Longo, F.; Puliafito, A.; Rana, O. Fog computing for the internet of things: a Survey. *ACM Trans. Internet Technol.* **2019**, *19*, 1–41.

33. Cao, H.; Wachowicz, M.; Renso, C.; Carlini, E. Analytics everywhere: Generating insights from the internet of things. *IEEE Access* **2019**, *7*, 71749–71769.
34. Bellavista, P.; Berrocal, J.; Corradi, A.; Das, S.K.; Foschini, L.; Zanni, A. A survey on fog computing for the Internet of Things. *Pervasive Mob. Comput.* **2019**, *52*, 71–99.
35. Xu, L.; Collier, R.; O'Hare, G.M. A survey of clustering techniques in WSNs and consideration of the challenges of applying such to 5G IoT scenarios. *IEEE Internet Things J.* **2017**, *4*, 1229–1249.
36. Qiu, T.; Chen, N.; Li, K.; Atiquzzaman, M.; Zhao, W. How can heterogeneous Internet of Things build our future: a survey. *IEEE Commun. Surv. Tutorials* **2018**, *20*, 2011–2027.
37. Akyildiz, I.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Networks* **2002**, *38*, 393–422.
38. Abbas, Z.; Yoon, W. A survey on energy conserving mechanisms for the internet of things: Wireless networking aspects. *Sensors* **2015**, *15*, 24818–24847.
39. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. A survey on sensor networks. *IEEE Commun. Mag.* **2002**, *40*, 102–114.
40. Jindal, V. History and Architecture of Wireless Sensor Networks for Ubiquitous Computing. *History* **2018**, *7*, 214–217.
41. Kooijman, M. *Building Wireless Sensor Networks Using Arduino*; Packt Publishing Ltd: Birmingham, UK, 2015.
42. Bacco, M.; Berton, A.; Ferro, E.; Gennaro, C.; Gotta, A.; Matteoli, S.; Paonessa, F.; Ruggeri, M.; Virone, G.; Zanella, A. Smart farming: Opportunities, challenges and technology enablers. In Proceedings of the IEEE IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany), Tuscany, Italy, 8–9 May 2018; pp. 1–6.
43. McConnell, M.D. Bridging the gap between conservation delivery and economics with precision agriculture. *Wildl. Soc. Bull.* **2019**, *43*, 391–397.
44. Farooq, M.S.; Riaz, S.; Abid, A.; Abid, K.; Naeem, M.A. A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming. *IEEE Access* **2019**, *7*, 156237–156271.
45. Joris, L.; Dupont, F.; Laurent, P.; Bellier, P.; Stoukatch, S.; Redouté, J.M. An Autonomous Sigfox Wireless Sensor Node for Environmental Monitoring. *IEEE Sensors Lett.* **2019**, *3*, 01–04.
46. Botero-Valencia, J.; Castano-Londono, L.; Marquez-Viloria, D.; Rico-Garcia, M. Data reduction in a low-cost environmental monitoring system based on LoRa for WSN. *IEEE Internet Things J.* **2018**, *6*, 3024–3030.
47. Yao, Z.; Bian, C. Smart Agriculture Information System Based on Cloud Computing and NB-IoT. *DEStech Transactions on Computer Science and Engineering*; 2019.
48. Biswas, S. A remotely operated Soil Monitoring System: An Internet of Things (IoT) Application. *Int. J. Internet Things Web Serv.* **2018**, *3*, 32–38.
49. Jawad, H.M.; Jawad, A.M.; Nordin, R.; Gharghan, S.K.; Abdullah, N.F.; Ismail, M.; Abu-Al Shaeer, M.J. Accurate Empirical Path-loss Model Based on Particle Swarm Optimization for Wireless Sensor Networks in Smart Agriculture. *IEEE Sensors J.* **2019**, doi:10.1109/JSEN.2019.2940186.
50. Li, N.; Xiao, Y.; Shen, L.; Xu, Z.; Li, B.; Yin, C.; others. Smart Agriculture with an Automated IoT-Based Greenhouse System for Local Communities. *Adv. Internet Things* **2019**, *9*, 15.
51. Kumar, S.A.; Ilango, P. The impact of wireless sensor network in the field of precision agriculture: A review. *Wirel. Pers. Commun.* **2018**, *98*, 685–698.
52. Azfar, S.; Nadeem, A.; Alkhodre, A.; Ahsan, K.; Mehmood, N.; Alghmdi, T.; Alsaawy, Y. Monitoring, Detection and Control Techniques of Agriculture Pests and Diseases using Wireless Sensor Network: a Review. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 424–433.
53. Azfar, S.; Nadeem, A.; Basit, A. Pest detection and control techniques using wireless sensor network: A review. *J. Entomol. Zool. Stud.* **2015**, *3*, 92–99.
54. Grift, T. The first word: the farm of the future. *Resour. Mag.* **2011**, *18*, 1.
55. Chunduri, K.; Menaka, R. Agricultural Monitoring and Controlling System Using Wireless Sensor Network. In *Soft Computing and Signal Processing*; Springer: Berlin, Germany, 2019; pp. 47–56.
56. Suárez-Albela, M.; Fernández-Caramés, T.M.; Fraga-Lamas, P.; Castedo, L. A practical evaluation of a high-security energy-efficient gateway for IoT fog computing applications. *Sensors* **2017**, *17*, 1978.
57. Castillo-Cara, M.; Huaranga-Junco, E.; Quispe-Montesinos, M.; Orozco-Barbosa, L.; Antúnez, E.A. FROG: a robust and green wireless sensor node for fog computing platforms. *J. Sensors* **2018**, *2018*.
58. Hossein Motlagh, N.; Mohammadrezaei, M.; Hunt, J.; Zakeri, B. Internet of Things (IoT) and the energy sector. *Energies* **2020**, *13*, 494.

59. Nikhade, S.G. Wireless sensor network system using Raspberry Pi and zigbee for environmental monitoring applications. In Proceedings of the IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), Avadi, Chennai, India, 6–8 May 2015, pp. 376–381.
60. Flores, K.O.; Butaslac, I.M.; Gonzales, J.E.M.; Dumlao, S.M.G.; Reyes, R.S. Precision agriculture monitoring system using wireless sensor network and Raspberry Pi local server. In Proceedings of the IEEE Region 10 Conference (TENCON), Singapore, 22–25 November 2016; pp. 3018–3021.
61. Deshmukh, A.D.; Shinde, U.B. A low cost environment monitoring system using raspberry Pi and arduino with Zigbee. In Proceedings of the International Conference on Inventive Computation Technologies (ICICT), Tamilnandu, India, 26–27 August 2016; Volume 3, pp. 1–6.
62. Ahmed, N.; De, D.; Hussain, I. Internet of Things (IoT) for Smart Precision Agriculture and Farming in Rural Areas. *IEEE Internet Things J.* **2018**, *5*, 4890–4899.
63. Bin Baharudin, A.M.; Saari, M.; Sillberg, P.; Rantanen, P.; Soini, J.; Jaakkola, H.; Yan, W. Portable fog gateways for resilient sensors data aggregation in internet-less environment. *Eng. J.* **2018**, *22*, 221–232.
64. Keshtgari, M.; Deljoo, A. A wireless sensor network solution for precision agriculture based on zigbee technology. *Wirel. Sens. Netw.* **2012**, doi:10.4236/wsn.2012.41004.
65. Cabaccan, C.N.; Cruz, F.R.G.; Agulto, I.C. Wireless sensor network for agricultural environment using raspberry pi based sensor nodes. In Proceedings of the IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Manila, Philippines, 1–3 December 2017; pp. 1–5.
66. Zamora-Izquierdo, M.A.; Santa, J.; Martínez, J.A.; Martínez, V.; Skarmeta, A.F. Smart farming IoT platform based on edge and cloud computing. *Biosyst. Eng.* **2019**, *177*, 4–17.
67. Souissi, I.; Azzouna, N.B.; Said, L.B. A multi-level study of information trust models in WSN-assisted IoT. *Comput. Networks* **2019**, *151*, 12–30.
68. Fortino, G.; Fotia, L.; Messina, F.; Rosaci, D.; Sarné, G.M. Trust and Reputation in the Internet of Things: State-of-the-Art and Research Challenges. *IEEE Access* **2020**, *8*, 60117–60125.
69. Cao, X.; Chen, J.; Zhang, Y.; Sun, Y. Development of an integrated wireless sensor network micro-environmental monitoring system. *ISA Trans.* **2008**, *47*, 247–255, doi:10.1016/j.isatra.2008.02.001.
70. Casado-Vara, R.; Prieto-Castrillo, F.; Corchado, J.M. A game theory approach for cooperative control to improve data quality and false data detection in WSN. *Int. J. Robust Nonlinear Control* **2018**, *28*, 5087–5102.
71. Adeel, A.; Gogate, M.; Farooq, S.; Ieracitano, C.; Dashtipour, K.; Larijani, H.; Hussain, A. A survey on the role of wireless sensor networks and IoT in disaster management. In *Geological Disaster Monitoring Based on Sensor Networks*; Springer: Berlin, Germany, 2019; pp. 57–66.
72. Poslad, S.; Middleton, S.E.; Chaves, F.; Tao, R.; Necmioglu, O.; Bügel, U. A semantic IoT early warning system for natural environment crisis management. *IEEE Trans. Emerg. Top. Comput.* **2015**, *3*, 246–257.
73. Kodali, R.K.; Sahu, A. An IoT based weather information prototype using WeMos. In Proceedings of the IEEE 2nd International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India , 14–17 December 2016; pp. 612–616.
74. Ayele, T.W.; Mehta, R. Air pollution monitoring and prediction using IoT. In Proceedings of the IEEE Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 20–21 April 2018; pp. 1741–1745.
75. Ghapar, A.A.; Yussof, S.; Bakar, A.A. Internet of Things (IoT) architecture for flood data management. *Int. J. Future Gener. Commun. Netw.* **2018**, *11*, 55–62.
76. Abraham, M.T.; Satyam, N.; Pradhan, B.; Alamri, A.M. IoT-based geotechnical monitoring of unstable slopes for landslide early warning in the Darjeeling Himalayas. *Sensors* **2020**, *20*, 2611.
77. Shaikh, S.F.; Hussain, M.M. Marine IoT: Non-invasive wearable multisensory platform for oceanic environment monitoring. In Proceedings of the IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019; pp. 309–312.
78. García, E.M.; Serna, M.Á.; Bermúdez, A.; Casado, R. Simulating a WSN-based wildfire fighting support system. In Proceedings of the IEEE International Symposium on Parallel and Distributed Processing with Applications, Sydney, NSW, Australia, 10–12 December 2008; pp. 896–902.

79. Kovács, Z.G.; Marosy, G.E.; Horváth, G. Case study of a simple, low power WSN implementation for forest monitoring. In Proceedings of the IEEE 12th Biennial Baltic Electronics Conference, Tallinn, Estonia, 4–6 October 2010; pp. 161–164.
80. Cantuña, J.G.; Bastidas, D.; Solórzano, S.; Clairand, J.M. Design and implementation of a Wireless Sensor Network to detect forest fires. In Proceedings of the IEEE Fourth international conference on eDemocracy & eGovernment (ICEDEG), Quito, Ecuador, 19–21 April 2017; pp. 15–21.
81. Yu, L.; Wang, N.; Meng, X. Real-time forest fire detection with wireless sensor networks. In Proceedings of the IEEE International Conference on Wireless Communications, Networking and Mobile Computing, Wuhan, China, 26 September 2005; Volume 2, pp. 1214–1217.
82. Li, Y.; Wang, Z.; Song, Y. Wireless sensor network design for wildfire monitoring. In Proceedings of the IEEE 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; Volume 1, pp. 109–113.
83. Díaz, S.E.; Pérez, J.C.; Mateos, A.C.; Marinescu, M.C.; Guerra, B.B. A novel methodology for the monitoring of the agricultural production process based on wireless sensor networks. *Comput. Electron. Agric.* **2011**, *76*, 252–265.
84. Manolakos, E.S.; Logaras, E.; Paschos, F. Wireless sensor network application for fire hazard detection and monitoring. In Proceedings of the International Conference on Sensor Applications, Experimentation and Logistic, Athens, Greece, 25 September 2009; Springer: Berlin, Germany, 2009; pp. 1–15.
85. Liu, Y.; Liu, Y.; Xu, H.; Teo, K.L. Forest fire monitoring, detection and decision making systems by wireless sensor network. In Proceedings of the IEEE Chinese Control and Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 5482–5486.
86. Ha, Y.g.; Kim, H.; Byun, Y.c. Energy-efficient fire monitoring over cluster-based wireless sensor networks. *Int. J. Distrib. Sens. Networks* **2012**, *8*, 460754.
87. Zhang, J.; Li, W.; Han, N.; Kan, J. Forest fire detection system based on a ZigBee wireless sensor network. *Front. For. China* **2008**, *3*, 369–374.
88. Jadhav, P.; Deshmukh, V.; others. Forest fire monitoring system based on ZIG-BEE wireless sensor network. *Int. J. Emerg. Technol. Adv. Eng.* **2012**, *2*, 187–191.
89. Trivedi, K.; Srivastava, A.K. An energy efficient framework for detection and monitoring of forest fire using mobile agent in wireless sensor networks. In Proceedings of the IEEE International Conference on Computational Intelligence and Computing Research, Coimbatore, India, 18–20 December 2014; pp. 1–4.
90. Muhammad, K.; Ahmad, J.; Baik, S.W. Early fire detection using convolutional neural networks during surveillance for effective disaster management. *Neurocomputing* **2018**, *288*, 30–42.
91. Kaur, H.; Sood, S.K. Fog-assisted IoT-enabled scalable network infrastructure for wildfire surveillance. *J. Netw. Comput. Appl.* **2019**, *144*, 171–183, doi10.1016/j.jnca.2019.07.005.
92. Khalaf, O.I.; Abdulsahib, G.M.; Zghair, N.A.K. IOT fire detection system using sensor with Arduino. *AUS* **2019**, *26*, 74–78.
93. Jadon, A.; Omama, M.; Varshney, A.; Ansari, M.S.; Sharma, R. Firenet: A specialized lightweight fire & smoke detection model for real-time iot applications. *arXiv* **2019**, arXiv:1905.11922.
94. Roque, G.; Padilla, V.S. LPWAN Based IoT Surveillance System for Outdoor Fire Detection. *IEEE Access* **2020**, *8*, 114900–114909.
95. Brito, T.; Pereira, A.I.; Lima, J.; Valente, A. Wireless Sensor Network for Ignitions Detection: an IoT approach. *Electronics* **2020**, *9*, 893.
96. Khan, S.; Muhammad, K.; Mumtaz, S.; Baik, S.W.; de Albuquerque, V.H.C. Energy-efficient deep CNN for smoke detection in foggy IoT environment. *IEEE Internet Things J.* **2019**, *6*, 9237–9245.
97. Muhammad, K.; Khan, S.; Elhoseny, M.; Ahmed, S.H.; Baik, S.W. Efficient fire detection for uncertain surveillance environment. *IEEE Trans. Ind. Informatics* **2019**, *15*, 3113–3122.
98. Cui, F. Deployment and integration of smart sensors with IoT devices detecting fire disasters in huge forest environment. *Comput. Commun.* **2020**, *150*, 818–827.
99. Khan, R.H.; Bhuiyan, Z.A.; Rahman, S.S.; Khondaker, S. A smart and cost-effective fire detection system for developing country: an IoT based approach. *Int. J. Inf. Eng. Electron. Bus.* **2019**, *11*, 16.
100. Kalatzis, N.; Avgeris, M.; Dechouniotis, D.; Papadakis-Vlachopapadopoulos, K.; Roussaki, I.; Papavassiliou, S. Edge computing in IoT ecosystems for UAV-enabled early fire detection. In Proceedings of the IEEE International Conference on Smart Computing (SMARTCOMP), Taormina, Italy, 18–20 June 2018; pp. 106–114.

101. Vimal, V.; Nigam, M.J. Forest Fire Prevention Using WSN Assisted IOT. *Int. J. of Eng. & Tech.* **2018**, *7*, 1317–1321.
102. Antunes, M.; Ferreira, L.M.; Viegas, C.; Coimbra, A.P.; de Almeida, A.T. Low-Cost System for Early Detection and Deployment of Countermeasures Against Wild Fires. In Proceedings of the IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 15–18 April 2019, pp. 418–423.
103. Jiang, H. Mobile Fire Evacuation System for Large Public Buildings Based on Artificial Intelligence and IoT. *IEEE Access* **2019**, *7*, 64101–64109.
104. Xu, Y.H.; Sun, Q.Y.; Xiao, Y.T. An Environmentally Aware Scheme of Wireless Sensor Networks for Forest Fire Monitoring and Detection. *Future Internet* **2018**, *10*, 102.
105. Lule, E.; Bulega, T.E. A scalable wireless sensor network (WSN) based architecture for fire disaster monitoring in the developing world. *Int. J. Comput. Netw. Inf. Secur.* **2015**, *7*, 40.
106. Yang, Y.; Prasanna, R.; Yang, L.; May, A. Opportunities for WSN for facilitating fire emergency response. In Proceedings of the IEEE Fifth International Conference on Information and Automation for Sustainability, Colombo, Sri Lanka, 17–19 December 2010.
107. Kalatzis, N.; Routis, G.; Marinellis, Y.; Avgeris, M.; Roussaki, I.; Papavassiliou, S.; Anagnostou, M. Semantic interoperability for iot platforms in support of decision making: an experiment on early wildfire detection. *Sensors* **2019**, *19*, 528.
108. Van Wagner, C.E. *Structure of the Canadian Forest fire Weather Index*; Environment Canada, Forestry Service: Ottawa, ON, USA, 1974; Volume 1333.
109. Hamadeh, N.; Karouni, A.; Daya, B.; Chauvet, P. Using correlative data analysis to develop weather index that estimates the risk of forest fires in Lebanon & Mediterranean: Assessment versus prevalent meteorological indices. *Case Stud. Fire Saf.* **2017**, *7*, 8–22.
110. Noble, I.; Gill, A.; Bary, G. McArthur's fire-danger meters expressed as equations. *Aust. J. Ecol.* **1980**, *5*, 201–203.
111. de Groot, W.J.; Wang, Y. Calibrating the fine fuel moisture code for grass ignition potential in Sumatra, Indonesia. *Int. J. Wildland Fire* **2005**, *14*, 161–168.
112. Sharples, J.; McRae, R.; Weber, R.; Gill, A.M. A simple index for assessing fire danger rating. *Environ. Model. Softw.* **2009**, *24*, 764–774.
113. Agusti-Torra, A.; Raspall, F.; Remondo, D.; Rincón, D.; Giuliani, G. On the feasibility of collaborative green data center ecosystems. *Ad Hoc Networks* **2015**, *25*, 565–580.
114. Hong, K.; Lillethun, D.; Ramachandran, U.; Ottenwälder, B.; Koldehofe, B. Mobile fog: A programming model for large-scale applications on the internet of things. In Proceedings of the Second ACM SIGCOMM Workshop on Mobile Cloud Computing, Hong Kong, China, 12 August 2013, pp. 15–20.
115. Banzi, M.; Shiloh, M. *Getting Started with Arduino: The Open Source Electronics Prototyping Platform*; Maker Media, Inc.: Sebastopol, CA, USA, 2014.
116. Faludi, R. *Building Wireless Sensor Networks: With ZigBee, XBee, Arduino, and Processing*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2010.
117. Farahani, S. *ZigBee Wireless Networks and Transceivers*; Newnes, Elsevier: Oxford, UK, 2011.
118. Osanaiye, O.; Chen, S.; Yan, Z.; Lu, R.; Choo, K.K.R.; Dlodlo, M. From cloud to fog computing: A review and a conceptual live VM migration framework. *IEEE Access* **2017**, *5*, 8284–8300.
119. Ojo, M.O.; Giordano, S.; Procissi, G.; Seitanidis, I.N. A Review of Low-End, Middle-End, and High-End Iot Devices. *IEEE Access* **2018**, *6*, 70528–70554.
120. Khutsoane, O.; Isong, B.; Abu-Mahfouz, A.M. IoT devices and applications based on LoRa/LoRaWAN. In Proceedings of the IEEE IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, China, 29 October–1 November 2017; pp. 6107–6112.
121. GFB. Data Sets. 2020. Available online: https://www.fireservice.gr/en_US/synola-dedomenon (accessed on 27 May 2020).
122. ESA. Greece suffers more fires in 2007 than in last decade, satellites reveal. 2007. Available online: http://www.esa.int/esaCP/SEMMGZLPQ5F_index_0.html (accessed on 2 July 2020).
123. GSCP. Daily Fire Risk Map. 2019. Available online: https://www.civilprotection.gr/en/daily-fire-prediction-map (accessed on 7 December 2019).
124. Adafruit. Digital relative humidity and temperature sensor AM2302/DHT22. Available online: https://cdn-shop.adafruit.com/datasheets/Digital+humidity+and+temperature+sensor+AM2302.pdf (accessed on 3 July 2020].

125. Digi. XBee®/XBee-PRO S2C Zigbee® RF Module User Guide. Available online: https://tinyurl.com/y5posdyh (accessed on 3 July 2020).
126. Aduino. ARDUINO UNO REV3. 2020. Available online: https://store.arduino.cc/arduino-uno-rev3 (accessed on 4 July 2020).
127. Aduino. ARDUINO MEGA 2560 REV3. 2020. Available online: https://store.arduino.cc/arduino-mega-2560-rev3 (accessed on 4 July 2020).
128. Foundation, R.P. Raspberry Pi 3 Model B. 2020. Available online: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/ (accessed on 4 July 2020).
129. Al-Sarawi, S.; Anbar, M.; Alieyan, K.; Alzubaidi, M. Internet of Things (IoT) communication protocols. In Proceedings of the IEEE 8th International Conference on Information Technology (ICIT), Jordan, 27–29 December 2017; pp. 685–690.
130. Glória, A.; Cercas, F.; Souto, N. Comparison of communication protocols for low cost Internet of Things devices. In Proceedings of the IEEE South Eastern European Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM), Kastoria,Greece, 23–29 September 2017; pp. 1–6.
131. Yaqoob, I.; Hashem, I.A.T.; Mehmood, Y.; Gani, A.; Mokhtar, S.; Guizani, S. Enabling communication technologies for smart cities. *IEEE Commun. Mag.* **2017**, *55*, 112–120.
132. Ali, A.I.; Partal, S.Z.; Kepke, S.; Partal, H.P. ZigBee and LoRa based Wireless Sensors for Smart Environment and IoT Applications. In Proceedings of the IEEE 1st Global Power, Energy and Communication Conference (GPECOM), Cappadocia, Turkey, 12–15 June 2019; pp. 19–23.
133. Mekki, K.; Bajic, E.; Chaxel, F.; Meyer, F. A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express* **2019**, *5*, 1–7.