Wireless Communications and Mobile Computing

The Green Internet of Things (G-IoT)

Lead Guest Editor: Fadi M. Al-Turjman
Guest Editors: Ahmed E. Kamal, Mubashir H. Rehmani, Ayman Radwan, and Al-Sakib K. Pathan

The Green Internet of Things (G-IoT)

The Green Internet of Things (G-IoT)

Lead Guest Editor: Fadi M. Al-Turjman
Guest Editors: Ahmed E. Kamal, Mubashir H. Rehmani, Ayman Radwan, and Al-Sakib K. Pathan

Editorial Board

Contents

Hindawi
Wireless Communications and Mobile Computing
Volume 2019, Article ID 6059343, 2 pages
https://doi.org/10.1155/2019/6059343

Editorial

The Green Internet of Things (G-IoT)

Fadi Al-Turjman,[1] Ahmed Kamal,[2] Mubashir Husain Rehmani,[3] Ayman Radwan,[4] and Al-Sakib Khan Pathan[5]

[1]*Antalya Bilim University, Antalya, Turkey*
[2]*Iowa State University, Ames, USA*
[3]*Cork Institute of Technology, Ireland*
[4]*Instituto de Telecomunicações Aveiro, Aveiro, Portugal*
[5]*Southeast University, Dhaka, Bangladesh*

Correspondence should be addressed to Fadi Al-Turjman; fadi@metu.edu.tr

Received 3 February 2019; Accepted 3 February 2019; Published 19 February 2019

The Internet of Things (IoT) has been envisaged to describe a number of technologies and research disciplines that enable global connectivity over the worldwide physical objects. Enabling technologies like Radio-Frequency Identification (RFID), sensor networks, biometrics, and nanotechnologies are now becoming very common, bringing the IoT into real implementations addressing varying applications, including smart grid, e-health, and intelligent transportation. They foreshadow an exciting future that closely interconnects our physical world via green networks. Green networks in IoT will contribute to reducing emissions and pollutions, exploiting environmental conservation and surveillance, and minimizing operational costs and power consumption.

The Green Internet of Things (G-IoT) is predicted to introduce significant changes in our daily life and would help realize the vision of "green ambient intelligence". Within few years, we will be surrounded by a massive amount of sensors, devices, and "things," which will be able to communicate via 5G, act "intelligently," and provide green support for users in managing their tasks. These new smart objects will also be context-aware and able to perform certain functions autonomously, calling for new forms of green communication between people and things and between things themselves, where power consumption is optimized and bandwidth utilization is maximized. This development would be relevant not only to researchers, but also to corporations and individuals alike. Considering these facts, the aim of this special issue (SI) was to focus on both theoretical and implementation aspects in green next generation networks or networks that can be utilized in providing green systems through IoT enabling technologies.

We are glad to reach this stage after processing all the submitted papers to this SI. Finally, we could include total four (4) high-quality papers that provide beneficial data points within the spectrum of this ongoing research and development efforts. These four papers were selected from 16 outstanding submissions; all accepted papers have undergone multiple reviews. We have been also very selective in choosing the reviewers. In the following, we summarize the common themes within these papers.

In the paper "An IoT Architecture for Assessing Road Safety in Smart Cities", the author introduces a novel, cost-effective Internet of Things (IoT) architecture that facilitates the realization of a robust and dynamic computational core in assessing the safety of a road network and its elements. In doing so, the author introduces a new, meaningful, and scalable metric for assessing road safety. He also showcases the use of machine learning in the design of the metric computation core through a novel application of Hidden Markov Models (HMMs). Finally, the impact of the proposed architecture is demonstrated through an application to safety-based route planning.

Meanwhile, the paper "Power Profiling of Context Aware Systems: A Contemporary Analysis and Framework for Power Conservation", addresses the need for smart service discovery, delivery, and adaptation through a context aware system, which adapts to the users' context. Given that the devices are mobile and battery operated, the main challenge in a context awareness approach is power conservation. The devices are composed of small sensors that consume power in the order of a few mW. However, their consumptions increase manifold during data processing. And thus, there is a need to conserve power while delivering the requisite functionality of the context aware system. Therefore, this feature is termed as 'power awareness.' In this paper, authors describe different power awareness techniques and compare them in terms of their conservation effectiveness. In addition, based on the investigations and comparison of the results, a power aware framework is proposed for a context aware system.

In the paper "Congestion Control and Prediction Schemes Using Fuzzy Logic System with Adaptive Membership Function in Wireless Sensor Networks", the network congestion challenge has been considered. This challenge is crucial in wireless sensor networks (WSNs) with restrictions and constraints, including limited computing power, memory, and transmission due to self-contained batteries, which limit sensor node lifetime. Determining a path to avoid congested routes can prolong the network. Thus, the authors present a path determination architecture for WSNs that takes congestion into account. The architecture is divided into 3 stages, excluding the final criteria for path determination: (1) initial path construction in a top-down hierarchical structure, (2) path derivation with energy-aware assisted routing, and (3) congestion prediction using exponential smoothing. With several factors, such as hop count, remaining energy, buffer occupancy, and forwarding rate, we apply fuzzy logic systems to determine proper weights among those factors in addition to optimizing the weight over the membership functions using a bat algorithm. The simulation results indicate the superior performance of the proposed method in terms of high throughput, low packet loss, balancing the overall energy consumption, and prolonging the network lifetime compared to state-of-the-art protocols.

Since the digital revolution led by the Internet of Things (IoT) is already reshaping several traditional business sectors. Moreover, because of its very nature, the promise of the IoT is also to reduce energy consumption and pollutant emissions in several environmental scenarios. At the same time, it is desirable to keep the development of IoT as sustainable as possible, hence truly realizing the vision of the green IoT. In the paper "An Open IoT Platform to Promote Eco-Sustainable Innovation in Western Africa: Real Urban and Rural Testbeds", authors show how a full-stack IoT framework can alleviate some real environmental problems afflicting countries in Western Africa. In this paper authors present three real IoT-based deployments currently hosted in two rural areas of Senegal and Ghana and one metropolitan area of Togo. These testbeds are connected to a Cloud-based software platform, purposely designed, and engineered to address some very specific environmental, economic, and social requirements of the region.

We hope that this collection of papers contributes to the active discussion among industry, academia, and governmental regulators. Ultimately, it will be the technical interchange among the IoT systems, and the wireless sensor networks that will support the emerging systems in the near future.

Conflicts of Interest

The editors declare that they have no conflicts of interest regarding the publication of this special issue.

Fadi Al-Turjman
Ahmed Kamal
Mubashir Husain Rehmani
Ayman Radwan
Al-Sakib Khan Pathan

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 8214989, 11 pages
https://doi.org/10.1155/2018/8214989

Research Article

An IoT Architecture for Assessing Road Safety in Smart Cities

Abd-Elhamid M. Taha

Electrical Engineering, Alfaisal University, Riyadh, Saudi Arabia

Correspondence should be addressed to Abd-Elhamid M. Taha; hamid.taha@gmail.com

Received 25 July 2018; Revised 16 October 2018; Accepted 1 November 2018; Published 19 November 2018

Academic Editor: Ahmed E. Kamal

The Safe System (SS) approach to road safety emphasizes safety-by-design through ensuring safe vehicles, road networks, and road users. With a strong motivation from the World Health Organization (WHO), this approach is increasingly adopted worldwide. Considerations in SS, however, are made for the medium-to-long term. Our interest in this work is to complement the approach with a short-to-medium term dynamic assessment of road safety. Toward this end, we introduce a novel, cost-effective Internet of Things (IoT) architecture that facilitates the realization of a robust and dynamic computational core in assessing the safety of a road network and its elements. In doing so, we introduce a new, meaningful, and scalable metric for assessing road safety. We also showcase the use of machine learning in the design of the metric computation core through a novel application of Hidden Markov Models (HMMs). Finally, the impact of the proposed architecture is demonstrated through an application to safety-based route planning.

1. Introduction

In its Global Status Report on Road Safety – 2015, the World Health Organization (WHO) noted that the worldwide total number of road traffic deaths has plateaued at 1.25 million per year, with tens of million either injured or disabled [1]. Different initiatives, such as the United Nations' initiative for the 2011-2020 Decade of Action for Road Safety, have led to improvements in road safety policies and enforcements. However, the WHO notes that the progress has been slow and has maintained the call for urgent action to reduce these figures [2].

Added to the losses in human lives and wellbeing, considerable monetary losses are incurred in medical expenses, infrastructure repair, and production downtime. While the worldwide figures have plateaued, the Global Status Report does indicate higher road fatalities and injuries in low-income countries. Such disparity, as noted in [3], signals a barring-limitation in low-income countries to improve road-safety by adopting solutions implemented in high-income countries.

The WHO describes different measures that can be implemented with minimal economic impacts in its "Save LIVES: Road Safety Technical Package" [4]. A cornerstone of these steps is realizing economic systems for "monitoring road safety by strengthening data systems". Meanwhile, a key theme in the package is motivating the adoption of a Safe System approach, which is a holistic approach to road safety that parts from traditional management solutions by emphasizing safety-by-design.

1.1. The Safe System Approach. The Safe System (SS) approach to transport networks originated with the "Safe Road Transport System" model developed by the Swedish Transport Agency. In its essence, the approach migrates from the view that accidents are largely and automatically the driver's fault to a view that identifies and evaluates the true causes for accidents. Through the categorization of safety into the safety of three elements (vehicle, road, and road user), SS minimizes fatalities and injuries by controlling speeds and facilitating prompt emergency response. The model has been widely adopted since its introduction and is currently motivated by the WHO as a basis for road safety planning, policy-making, and enforcement.

An illustration of the model is provided in Figure 1. A central emphasis is given to speed in the SS approach as it is the strongest and most fundamental variable in the

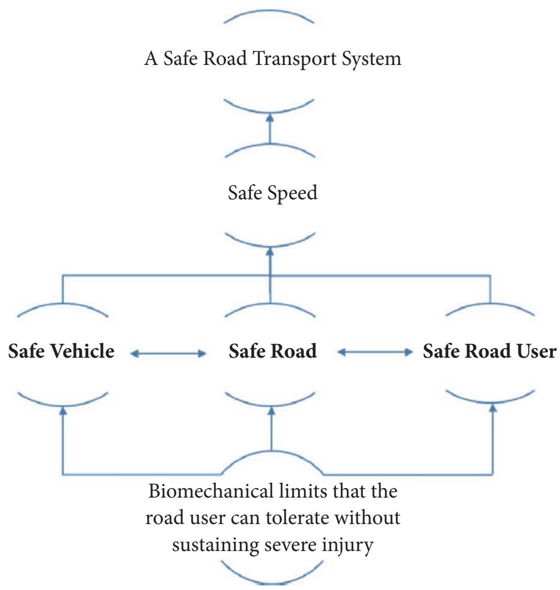


Figure 1: The Safe-System-based Safe Road Transport Systems, with its elements: safe vehicle, safe road, and safe road user [5].

outcome of fatality. The fragility of the human body makes it unlikely to survive an uncushioned impact at a speed of more than 30 km/h, with lower speeds resulting in either death or serious injury [3, 4]. The objective of the SS approach is that the three model elements should be designed and monitored to proactively prevent deadly speeds from happening and allow for a reduced emergency response time in the event of an accident.

Elements of the SS approach are as follows.

(1) Safe Vehicle. Emphasis on vehicle safety is verified through mandated regulatory testing and rating, as well as technologies such as electronic stability control. Beyond this, enforced checks (e.g., upon license renewals) combined with on the road reporting work to review the status of vehicle safety.

(2) Safe Road. The assessment of road (or road network) safety is multifaceted. Road inspection enables clear and direct observation of the state of the road and assesses the need for repairs or modifications. The structure of the road network is amenable to safety assessment through partitioning into what is called "Traffic Analysis Zones (TAZs)" [8]. In addition, considerations for crash data and other supporting data offer further insights into general safety assessment.

In 2011, the European Road Assessment Programme (EuroRAP) generated the European Road Safety Atlas for EU countries [9]. The atlas indicated the safety level of roads with a star rating based on specially equipped vehicles for multimedia-based data aggregation [10]. The EuroRAP efforts continue to implement an SS approach across the EU, along with several other national programmes within the International RAP, or iRAP, initiative [11].

(3) Safe Road User. There are several aspects to road user safety, including measures for education and awareness, travel distance, exposure, licensure, enforcement, and sober driving [5]. The need for such characterization rises substantially as the findings of crash report analysis in cities typically note a critical dependence on either driver behavior or driver awareness [12]. A great need is further established in these studies for innovative mechanisms to instill safe driving at the licensing and post-licensing stages.

1.2. Contributions. Figure 2 illustrates elements of assessing road safety. It can be seen in the figure that the scope of consideration in the SS approach is medium-to-long term, facilitating by design, systemic actions that are made to ensure the safety of the road network. While the use of "data monitoring systems" is motivated in [4] and can be utilized for shorter term scopes, the general emphasis is maintained at the medium-to-long term reaction cycles.

Our interest in this work is to extend SS to the short-to-medium term through exploiting recent advances in the context of the Internet of Things (IoT) and Intelligent Transport Systems (ITS) [13, 14]. This fits the outlook for Smart Cities where automation is emphasized to address the increasing dynamic nature of city elements [15].

Toward this, our contributions in this work are as follows:

(i) This work offers a comprehensive, IoT-based architecture with the objective of assessing the safety of the transportation road network.

(ii) In doing so, the proposed architecture is aligned with the SS approach in its entirety.

(iii) It also complements the SS approach by addressing the void in its short-to-medium scope of considerations, making the approach further fitting to the dynamic nature of smart cities (note the scopes illustrated in Figure 2).

(iv) Finally, the proposed architecture showcases the viability of an economic road safety monitoring through advances in IoT and ITS, especially those aimed at realizing smart cities.

The proposed architecture involves a novel use of machine learning as part of its road safety assessment core. This application facilitates assessments that are both dynamic and robust. We also showcase an application of the developed core aimed at safety-based route planning in smart cities.

1.3. Paper Organization. The remainder of this work is organized as follows. Section 2 reviews related work and motivates and positions the contributions made herein. Section 3 introduces the dynamic road safety assessment, with descriptions on architectural considerations, while Section 4 elaborates on the dynamic assessment core. Section 5 details an application of the developed assessment core in the context of route planning. Section 6 validates the dynamic assessment core together with a demonstration for route planning. Finally, Section 7 concludes this work.

2. Related Work and Motivation

This section reviews related works within the context of IoT and ITS, and their integrations within the more general

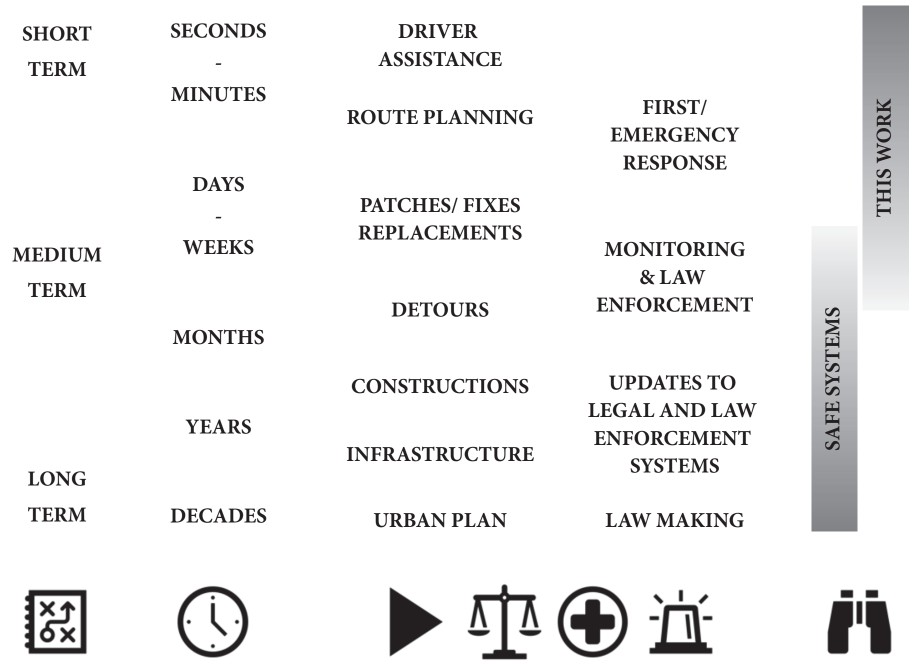


Figure 2: Safe System (SS) elements across different time considerations, i.e., short-, medium-, and long-term.

context of smart cities. The reviewed works have been categorized and presented per their relevance to elements of the adopted SS approach, i.e., in terms of facilitating safety in vehicles, roads, and drivers.

2.1. Safe Vehicle. As noted above, regulatory enforcements frame a substantial aspect of ensuring that vehicles traversing a city's road network are safe and reliable. Considerations for the short-to-medium term, however, require a more "real-time" monitoring of the vehicle status. Telematics allows for such monitoring within the IoT/ITS context and is facilitated by several options. The first includes having dedicated sensors, such as accelerometers, Carbon Monoxide (CO) level sensors, etc., mounted on the vehicle to gather and log information [15]. Such setups can be augmented with a communication module so that the collected data can be transferred to a local unit or to the cloud.

An alternative telematic approach involves accessing a vehicle's Controller Area Network (CAN), which is the network that interconnects a vehicle's computing and sensing capabilities [16]. Such access is made possible by a North American standard ratified in 1996, namely, the second-generation On-Board Diagnostics, or OBD-II. Since their introduction, OBD-II dongles have come a long way, with some models offering a mix of connectivity including Bluetooth, WIFI, and cellular (e.g., [17]). Through the OBD-II, various real-time and diagnostic information can be accessed, logged, and communicated, including RPM, speed, pedal position, coolant temperature, etc. This has allowed for applications such as TorquePro [18], which monitors a vehicle's fuel efficiency, and advises the driver more fuel-efficient driving behavior. More relevant here, another application has made it possible to identify when maintenance is required for a vehicle [19]. Meanwhile, OBD-II manufacturers, such as MUNIC [17], offers cloud-based portals for aggregating, processing, and visualizing sensed data, and that can be access for further processing by users.

The viability made by the above setups provides a key thrust to this work, especially as they enable an immediate assessment of a vehicle's safety. As will be demonstrated below, the introduction of smartphones allowed for more accessible in-vehicle monitoring that can be functional to various degrees, regardless of how the phone is mounted in the vehicle. Smartphones have also been used on their own or in a combined in-vehicle sensing unit including either dedicated sensors, OBD-II dongles, or both.

Many of these applications targeted the assessment of either road status or capturing driving behavior and are thus further elaborated on in the following two subsections.

2.2. Safe Road. As aforementioned, especially equipped vehicles within the xRAP programmes are optimal in how they facilitate exhaustive road inspections and ratings. Alternative approaches, however, have been sought using various cost-effective setups.

For example, in [20] an embedded device is realized to support various sensing techniques in road surface monitoring. Meanwhile, the system in [21] is proposed for the detection of wet-road conditions based on images captured by cameras mounted on the rear-view mirror of a vehicle. Specifically, the system employs image analysis for extracting features related to water and snow on the road. Other systems require the addition of simple hardware to such vehicles to widen the scope of the detection applications. For instance, the pothole patrol [22] depends on the deployment of 3-axis accelerometers on board of vehicles for detecting such road conditions through monitoring vibration. Another example proposed in [23] is a system that detects ice on roads by analyzing tire-to-road friction ultrasonic noise detected by a transducer installed behind the front bumper.

Several works have also employed sensing features in smartphones and tablets. For example, the work in [24] aims at recognizing road surface anomalies through the combination of accelerometers and GPS data on a tablet, allowing for ease of data aggregation and reduced cost.

A smartphone system of note is *Nericell*, which utilizes smartphone accelerometers, microphones, and GPS to detect events related to the quality of the road, e.g., potholes and bumps [25]. Vibration thresholds are used to decide on detecting such road conditions, and to detect events linked to traffic including braking and vehicles' noise. Deceleration thresholds that are sustained for a predefined time interval are used to detect braking events, while microphones are used for detecting the noise levels. Another threshold-based inference implementation is proposed in [26]. The sensor measurements, accelerometers and photometers, are compared against a set of empirical thresholds and the anomaly is then detected when all thresholds are satisfied in a given road. The values of the thresholds are chosen based on measurements that indicated the existence of true anomalies within their values, while making sure that no anomalies are outside these intervals.

In [17], Bayesian inference and Unscented Kalman Filter (UKF) are used to determine the probability that a reported sequence of wheel measurements during nominal durations of time correspond to a pothole in the associated location. Bayesian inference is also applied for cooperative applications such as weather condition monitoring in [27]. The system aims to detect the foggy and icy conditions that would increase the probability of car accidents. Particularly, given the measurements of fog (from a thermal imaging camera) or icy conditions (from the temperature or the car stability), a belief function is calculated for each of these two conditions.

It has been advised that general caution should be exercised in relying on smartphones for sensing as some of their characteristics may limit sensing accuracy [28]. These characteristics include the smartphone's orientation, as well as its relative position inside the vehicle. Smartphones may further be limited in being able to directly capture a vehicle's external context, e.g., weather or visibility. Other sources may thus be needed to validate or augment the smartphone's view.

For example, the viability of crowdsourcing for smartphone users facilitates a further advantage, especially when it comes to validation. The system in [29] resolves the low accuracy stemming from the exclusive use of smartphones and relies on applying crowd-sourcing and on using sheer numbers to combat the presence of false positives. Meanwhile, authors in [30] consider pothole detection using a large dataset collected through implicit crowdsourcing, i.e., crowdsourcing without repeated user prompt/input. Thresholds are used on the phone's z-axis acceleration, with empirical data used for differentiating potholes from speed breakers.

Careful consideration should be made in evaluating the quality of aggregated data. For example, the work in [31] notes that various errors are introduced when crash data is reported by enforcement or insurance agencies. Meanwhile, validation mechanisms need to be introduced for any data aggregation measures based on common, i.e., nondedicated, technologies such as smartphones.

2.3. Safe Road User. Driver Behavior Modelling (DBM) [32, 33] is an area of road safety management that is concerned with the characterization of driver behavior. This characterization is enabled through the analysis of various inputs from either the transportation infrastructures, e.g., on-road CCTV cameras, speed-sensors; other infrastructures, e.g., smartphones, reporting to services such as Waze or Google Maps, registrations to cellular-base stations; or an in-vehicle sensing setup. Combined or separated, baselines for "safe" or "responsible" driving can be synthesized, against which counter driving behaviors are identifiable. Meanwhile, considerations for driver awareness or alertness can also be realized to extend identification to behaviors exhibited when driving under fatigue, distraction, or influence.

A smartphone-based driver activity recognition system is proposed in [34] with the objective of preventing drivers from texting while driving. The system identifies whether a smartphone holder (a) has entered a vehicle; (b) has boarded the vehicle from the left or the right; (c) sat in the front or back seat; and (d) is texting. Another system that differentiates drivers from passengers is offered in [35]. The system in [36] employs fuzzy logic and utilizes the acceleration, gravity, magnetic, and GPS sensors to estimate driving aspects such as jerk, orientation rate, speed variation, and bearing variation. A fusion module is then employed to distinguish activity such as hard/sudden acceleration or overspeeding.

The work in [37] exploits smartphone cameras to monitor the driver's alertness through recognizing head position and body orientation. It also utilizes the smartphone's back camera to process the driver's lane-change. Additionally, the system assists the user in detecting vehicles in the driver's blind-spot and alerts the driver if a lane change is undertaken while another vehicle is occupying the blind-spot.

Smartphone sensing is also utilized in [38] using both accelerometers and cameras to identify unsafe stopping behavior in busses. Meanwhile, a smartphone-based system is provided in [39] to identify aggressive driving styles.

Applications of combined smartphone and OBD-II sensing are provided in [40, 41], enabling the detection of different driving styles, and classifying whether the style is safe or unsafe. Particularly in [41] where engine load, RPM, speed, and pedal position are fused with smartphone data to assess driving behavior.

While much of the DBM has focused on human drivers, the general arguments presented herein are applicable to both human and autonomous driving agents. The model synthesized in [33], for example, considers human driving, assisted driving, and autonomous driving. The model synthesized is based on the Naturalistic Driving Study within the Second Strategic Highway Research Program (SHRP2-NDS).

The extensive data set relied on sensed data from in-vehicle setups synced with information on traversed roads, utilized vehicles, traffic conditions, and weather conditions. Through this comprehensive model, it is possible to identify behaviors such as "exceeded speed limit", "rolling stops", "drowsy", "sleepy", "asleep", and "fatigued". The model also enables characterizing distracted driving, as well as the nature of distraction, e.g., "cell phone, texting", "cell phone, holding", and "passenger in interaction seat – interaction".

2.4. Motivation. Achieving road safety is a multiterm and multifaceted objective, and the above discussion indicates strong emphasis in the SS approach on the medium-to-long term whereby road safety is achieved by design. Our interest in this work is to complement SS with a framework for a dynamic assessment of road safety. Our objectives are to first accommodate the dynamic nature of city traffic, especially in cases of major events and/or crisis. Secondly, it is to showcase the viability of an economically attractive alternative to monitor road safety using ubiquitous technologies and advances in IoT. This becomes critical to overcome any barring limitations, especially for low-to-medium income countries.

3. An IoT Architecture for Assessing the Safety of a Dynamic Road Transport System

In reviewing the related works in the previous section, we showcased how various advances are enabling the assessment of safety of vehicles, roads, and drivers. The objective of this section is to introduce a novel and adaptive IoT architecture that enables the assessment of safety in a city's road network. We elaborate on the assessment elements and how they can be used to synthesize a single, meaningful indicator for safety. We also describe the architecture components and their interrelationships, including a robust computational core for safety assessment.

3.1. Assessment Elements. The way the SS approach comprises the three elements of safe vehicle, safe road, and safe driver facilitates a hierarchical safety assessment approach whereby the safety of the individual elements can provide a collective indicator of safety for the road network, as illustrated in Figure 3. In turn, this indicator can be concatenated from the assessment of individual road segments, to routes, to the road network.

For vehicles, the assessment core would rely on inferences from the vehicle's Vehicle Identification Number (VIN) (thus establishing car make, manufacturing, and base safety rating); regulator's information (e.g., the outcome of the last regulatory check); and the updated information from an OBD-II unit.

For roads, the reliance on both historic and newly sensed data can facilitate an understanding of the road context, the road network map, and establish severity of turns, the presence of shoulders, and whether the road is on a hillside. Identifying a vehicle's location through an on-board GPS (whether dedicated or through a smartphone) would thereby

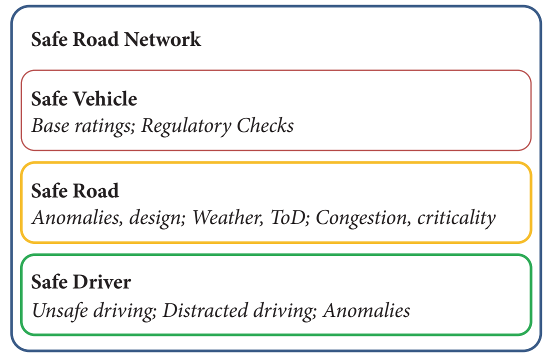


Figure 3: Elements in assessing the safety of the road network based on the safe systems approach.

provide for a first level of this understanding. Meanwhile, in-vehicle sensing can be utilized to report road anomalies (e.g., potholes). Sensing based on the OBD-II interface can also help identify instances of skidding, which in turn can be correlated with weather or reports. Weather, together with time of the day (ToD) and reports on the availability and/or state of road-lights, can be combined to ascertain visibility. Moreover, basic traffic information can indicate congestion levels, as well as road/segment criticality, i.e., repeated selection in route plans generated within a short span of time.

As for drivers, the growing maturity of DBM will facilitate identifying unsafe or distracted driving behavior. It will also enable the recognition of localized driving behavior anomalies, facilitating the identification of emerging incidents, e.g., several swift lane changes to avoid a new obstacle in the road due to a falling tree trunk or other road debris.

It is possible to consider a meaningful safety metric based on the live (or real-time) status of the road. For example, the safety level of a certain segment/road depends on the aggregate safety of vehicles currently traversing it, combined with the number of potholes and/or the wetness or how slippery is the road, in addition to safety/alertness of the drivers on the road.

In designing our architecture, we exploit three important dependencies. The first is between the SS elements, e.g., how well a car can handle a certain road, or how some drivers exhibit safer behavior in instances of higher visibility. The second dependency is in between consecutive segment/roads, especially in terms of traversing vehicles and drivers. The third dependency is like the second but is established in time. Abrupt changes in safety levels can thus be viewed as an anomaly (outlier) or inferred as indicator to a substantial change in the road context.

3.2. Architecture Considerations. An illustration of architecture consideration is provided in Figure 4. At the core of the proposed architecture is a computational core that operates in two tiers. The first is concerned with dynamically assessing the safety of each aspect in SS. The second utilizes the outcome of these three assessments to generate a single rating for the road.

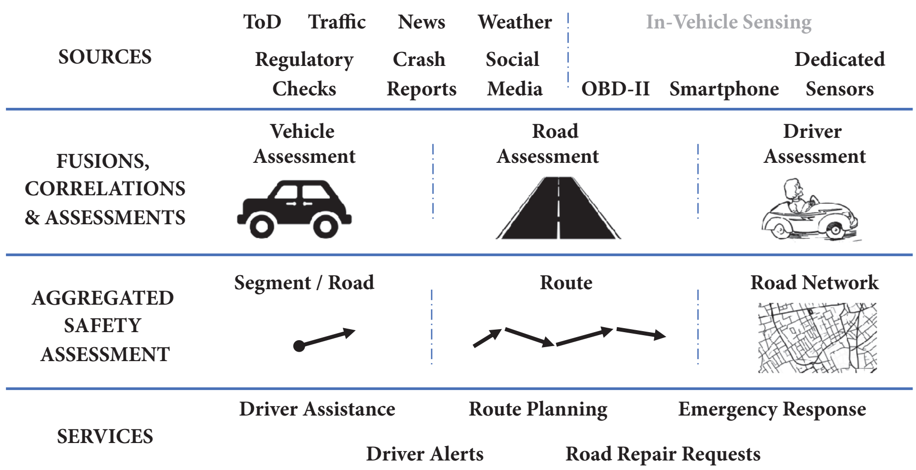


Figure 4: Architecture considerations for dynamic assessment of road network safety.

Figure 5: An example breakdown of fusion, correlation, and assessment for a Safe System component.

The individual assessments rely on a wide array of sources that generally fall into two categories. The first is sourced through in-vehicle sensing, while the second through sources that establish the general context of the road. These include traffic problems, regulatory checks, crash reports, processing of news and social media, weather updates, and time of day.

As aforementioned, inferences and assessments in each of the three elements of SS can utilize the same sources. This motivates the placement of the data fusion and correlation element in the architecture, as shown in Figure 5. In data fusion, sources that yield to the same knowledge synthesis are combined to enhance its reliability, while correlation is concerned with cotiming and colocation of sensed data, as well as cross-validation of input knowledge streams.

Based on the output of the three assessment modules, a live representation of a segment's safety to traverse can be made. We elaborate on a model for generating this representation below. Once the aggregated safety assessments have been generated, the values can be provided to several services that include driver-assistance modules and generating requests for road repair or emergency response.

In the following section we demonstrate how this assessment core can be utilized in a safety-based route planning application.

3.3. In-Vehicle Sensing. A possible setup for in-vehicle sensing comprises three elements: an OBD-II dongle, a dedicated sensor, and a smartphone. We evaluated two OBD-II interface alternatives: munic.box's Munic3 4G-cat4-WBT dongle and ScanTool's OBDLink MX Bluetooth. Both performed as desired on the tested vehicles. The dedicated sensor was Aaronia's GPS logger, which combines sensors for GPS localizations, compass, accelerometer, gyroscope, and altimeter with pressure sensor. The logger does not have wireless communication capabilities but can be connected to a PC through USB. Meanwhile, recent iOS and android-based phones were utilized for both their communication and sensing caspabilities.

While an objective of the work is to realize a cost-effective solution for dynamic assessment, the purpose of the setup described above was designed for both exploratory and validation purposed. Where the sensing elements were able to connect to the cloud repository directly, e.g., as in the case of the munic.box dongle, or the smartphone, the connection was allowed. Otherwise, a connection was established indirectly through another device, e.g., the logger through the PC laptop or ScanTool's dongle through the smartphone.

For the backend, sensed data and be transferred and stored on a PostgreSQL 9.6 built on a Red Hat Enterprise Linux Developer OS. This setup would allow for an Apache web portal access, facilitating quick prototyping and ease of simultaneous data reports from multiple sources.

3.4. The Assessment Core. Road networks comprise a complex, living structure, and thus require care in modeling any of their aspects. This is especially the case in cities and high density urban areas where traffic dynamics becomes further involved and further engaged with other planes of city activity.

In remarking on dependencies above, we noted that a reasonable model to assess a road network's safety should mind the interelement dependencies in SS, as well as the dependencies in both space and time. It should also facilitate the quantification of these interdependencies without substantial overhead.

The design of such assessment core is elaborated upon in the next section.

4. Designing a Dynamic Assessment Core

Recent thrust into machine learning and its different variants (supervised, unsupervised, and reinforcement-based) has made addressing high processing demands in solving various problems possible. The operational requirements highlighted in Section 3.4 above position the target assessment core as a typical candidate for machine learning considerations. In what follows, we showcase an example modeling to the road network that can facilitate a robust and dynamic processing.

Hidden Markov Modelling (HMM) is a powerful statistical tool for modelling time-series systems that can be characterized to represent probability distributions over a sequence of observations. The tool thus lends itself easily to the nature of data gathering found in IoT and smart cities applications. It further stands as a potential base model for several machine learning approaches, including Bayesian or Mixture Density Network inferences.

Our interest herein is in a novel application HMMs to our problem. Specifically, a first-order, time-homogeneous, and discrete HMM is employed to identify the degree to which traversing a certain road link is safe, thereby realizing a safety metric.

The HMM at hand can be formally defined by the five-tuple $\Phi = (S, M, P, \delta, \pi)$, where S is an array of links' states; M is the emission symbols that characterizes observations per each state; P is the states' transition probabilities; δ is the emission (or output) probability matrix; and π is the initial state distribution array. We assume that the road link state in terms of safety, denoted S_t, is hidden from the observer. We also assume that the current hidden state of a certain road link depends only on the preceding state of the same road, i.e., that the Markov property is satisfied.

In what follows we elaborate on the tuple elements.

States, *S*: the hidden states of the road link status, which describe the safety of the road link. For example, and without loss of generality, two-states can be utilized, whether safe or not. Further in-between states can be added.

Emission Symbols, *M*: the observations from which the hidden states can be deduced. Examples of possible observations are road link congestion rate; road condition metric; road infrastructure type, e.g., number of road link lanes, type side or highway link; or road infrastructure characteristic; e.g., road visibility, etc. A metric utilizing a combination of two or more of these and other observations can also be synthesized.

For illustration, M can be made to capture road link congestion, with $M_i(t) = 0, 1, 2, \ldots, m - 1$, representing m levels of congestion. If $m = 4$, the congestion can be digitized into 4 levels at 0, 25%, 50%, 75%, all relative to the link's maximum capacity. If we define the random variable $q_n(t)$ as the congestion of road segment, n at time, t, then, $M_n(t)$, as a function of road segment congestion, can be defined to be as follows.

$$M_n(t) = \begin{cases} 0, & 0.25 > q_n(t) \geq 0 \\ 1, & 0.5 > q_n(t) \geq 0.25 \\ 2, & 0.75 > q_n(t) \geq 0.5 \\ 3, & 1 \geq q_n(t) \geq 0.75 \end{cases} \quad (1)$$

It is essential to note, however, that the limits of these levels would need to be normalized to usefully reflect actual links congestion levels.

Transition Probabilities, *P*, is the probability of transition among the two states in the HMM.

Probability of Emission, $\boldsymbol{\delta}$ (or output probabilities), is equal to $P(M_i/S_1)$ and $P(M_i/S_2)$ given the current state is S_1 (safe) or S_2 (unsafe) road link, respectively. Extending the illustrative example above, δ_i can be calculated if we know the probability distribution of $q_n(t)$, an empirical distribution interpolated from the sensory data. The empirical distribution can be approximated into the best standard distribution, if possible, taking into consideration the trade-off between accuracy and complexity.

Initial State Distribution, π, specifies the initial probability distribution of the states. While typically initialized using a uniform distribution assignment, there are generally no assumptions needed regarding prior distribution.

Three traditional algorithms can be employed to efficiently compute an HMM, namely, the Viterbi algorithm, the Forward-Backward algorithm, and the Expectation-Maximization algorithm [42]. This is particularly advantageous in the computation of link safety for our purposes. Furthermore, once the HMM is determined, the state of the links (as well the state of regions) can be identified and utilized using various services, including route planning, as will be discussed in the next section.

5. Safety-Based Route Planning

Route planning has become widely used in both personal and commercial use, resulting in an increasing dependence on its reliability. Various applications employ efficient algorithms for route planning [43]. Trip time and cost, e.g., for tolls, have been the typical metrics for route planning applications, but other metrics, however, have been utilized, e.g., for fuel emission/consumption or energy requirements of electric vehicles.

Using the dynamic safety assessment proposed above, it is now possible to route vehicles across cities based on a safety. In this manner, drivers can be directed through routes that minimizes their overall risk in traversing the road network. Meanwhile, enforcement can distribute vehicles across different paths to distribute risk of the network and avoid having critically unsafe links or routes within the network. It is furthermore possible to target auxiliary mechanisms for safety-control across the network by controlling and redirecting traffic based on user driving behavior or in-response to incidental changes in the road network.

An advantage of the assessment core proposed above is that a routing algorithm can be operated directly on its generated values. In what follows we describe a direct application for routes assumed to be traversed shortly after the route have been computed, and that require a traversal time sufficiently less than transition time in the HMM. These assumptions are without loss in generality and can be relaxed with easy modifications to the route planning formulation presented below.

Consider a graph (V, E), with V comprising $n+1$ nodes (vertices), and E comprising the edges in the graph. Nodes represent starting, ending, and midway stops for the vehicle, and our interest is in routing a vehicle from a source node (s) to a destination node (d). Vertices are further identified by numbers, with vertex 0 identifying the source node and $0 < i < n$ identifying possible target destinations. Nonsource nodes make the set $V^* = V - \{0\}$.

The set of edges $E = \{(i, j) : i, j \in V; i < j\}$ represents the set of $n \cdot (n+1)/2$ links between the $n+1$ nodes. Each edge has an associated traversal cost $c_{ij} > 0$, which may be either symmetric ($c_{ij} = c_{ji}$) or asymmetric ($c_{ij} \neq c_{ij}$). Herein, this cost can be inversely proportional to a link's safety assessment.

Considering the above, a formulation can be presented as shown in Table 1. The formulation minimizes the expended cost. The first four constraints suffice for a general (uncapacitated) shortest-path problem. The first constraint mandates that a stop along the chosen route is visited only once. The second constraint specifies that, except for the source and the destination, each stop has as many ingoing as outgoing traversals. The third and fourth constraint mandate single departure from the source and a single arrival at the destination.

The above formulation is an integer-programming instance, making the essential route planning problem NP-hard. Meanwhile, when engaging the computation core, as specified by the formulation above, the core is supplied with the updated "costs" and the "capacities" of the edges. This *updating*, while facilitating more relevant solutions, results in solutions that are at best critically stable. Dealing with probabilistic coefficients or constraint thus becomes inevitable when extending considerations for more realistic modelling. Such coefficients and limits, however, only add to the problem complexity.

The application of the road-safety assessment core introduced in Section 4 above overcomes this complexity aspect by simplifying the handling of the probabilistic. Enumerations. Specifically, the formulation can be revised to be applied over the HMM abstraction of the road-network, allow an end-to-end path selection based on safety.

6. Results and Validation

We validate the work through a case study based on data available for New York County, NY, USA. Maps and shapefiles were sourced from Open Street Maps (OSM) [6], United States Census Bureau [7], and David Gleich's US Roads extracts [44]. We utilized collisions data sourced from the New York City (NYC) OpenData portal [45]. Notwithstanding, the model applicability extends to all localized data. Core processing were performed in Matlab [46], with some preprocessing made in Geomatica 17 [47].

Table 1: Formulation for the route planning problem.

$\min \sum_{k\in F} \sum_{\substack{i,j\in V\\ i\neq j}} c_{ij}x_{ij}$
Subject to: -
$x_{ij} \in \{0,1\} \quad \forall i, j \in V$
$\sum_{k\in F} \sum_{i\in V} x_{ij} = 1 \quad \forall j = V^*$
$\sum_{i\in V} x_{iu} - \sum_{j\in V} x_{uj} = 0 \quad \forall u \in V \setminus \{s, d\}$
$\sum_{i\in V} x_{iu} - \sum_{j\in V} x_{uj} = -1 \quad \text{if } u = s,\ \forall u \in V^*$
$\sum_{i\in V} x_{iu} - \sum_{j\in V} x_{uj} = +1 \quad \text{if } u = d,\ \forall u \in V^*$

Figure 6(a) shows the map of the target area of analysis, while Figure 6(b) shows the road network as extracted from OSM shapefile. We note the longitude and latitude limits in Figure 6(b) as they dictate our focus for the results to follow.

The NYC OpenData collisions dataset, as detailed in [45] specifies collision date, time, longitude, latitude, and address (street and/or intersection). The dataset also qualifies collisions based on deaths and injuries for motorists, pedestrians, and cyclists, as well as other contributing factors and vehicle types. The data is dated starting December 2, 2013, and populated until the date of download. An example of collision overlay on the road network is shown in Figure 7, where the red dots denote collision location. The extract in Figure 7 is partial for the map for the date June 23, 2018 during the hours 06:00 to 07:30.

Close analysis of the data details further characterization of collision distribution in terms of time and space. Here, we isolated the data for 2018 (starting January 1, 2018) and reviewed their space and time characteristics.

Figure 8 shows an overlay between location aggregate and the County's street network. This showcases a higher number of collisions in areas mapped to the County's main ingress/egress points (i.e., the Lincoln Tunnel, Manhattan Bridge, etc.), while the remainder of County area have on average near-equal collision numbers.

A second view is availed through reviewing collision distribution in time, which is shown in Figure 9. The histogram shows a drop in average daily collisions during the hours 00:00 (i.e., midnight) to 06:00, with a substantial but piecewise rise in collision counts during the hours 06:00 to 18:00.

Based on the above, we proceeded to feed the road network's HMM. In doing so, we mapped the network' edges to the HMM's nodes. We also considered four six-hour time windows, namely, the hours 00:00 to 06:00; 06:00 to 12:00; 12:00 to 18:00; and 18:00 to 24:00. We note, however, that a closer categorization of collision might be needed after isolating weekdays from weekends. Notwithstanding, the choice of windows does not result in loss of generality for results.

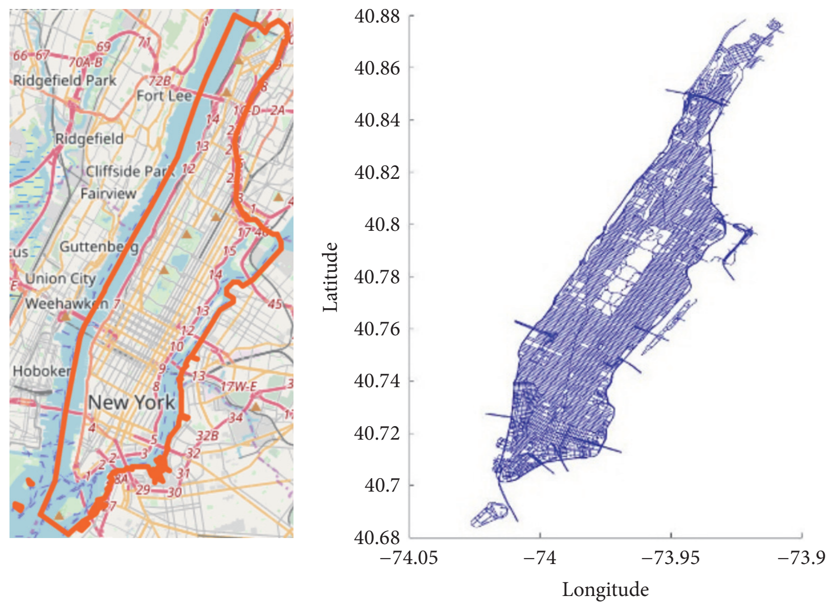


FIGURE 6: Extracts for New York County from Open Street Maps (OSM): (a) overall map extract [6]; and (b) extract of the New York County road network [7].

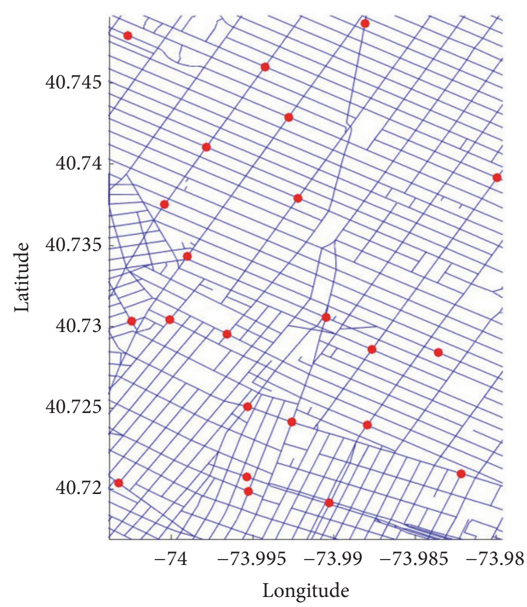


FIGURE 7: Collisions during the hours 06:00 to 07:30 on June 23, 2018, overlaid on a partial map of the New York County road network, with collision locations shown in red dots.

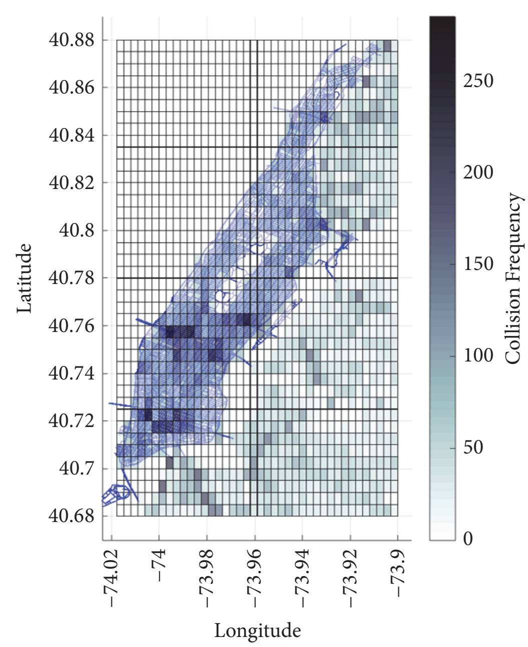


FIGURE 8: Overlay of New York County street network and collisions for 2018 (January 1, 2018 onward). Darker regions indicate higher collision frequencies.

Safety level was graded into four levels. Emission was normalized across County area in a manner like the description in Section 4. The HMM was initialized with uniform distribution.

We implemented the HMM using Matlab as per the description offered in Section 4. In doing so, we mapped the state of safety to the edge weights in the County road network. Figure 10 shows the route plan between two points on the road network, with the red highlighting the shortest path based on distance, and the green based on safety. The shortest path algorithm followed the description provided in Section 5.

7. Conclusions

This work illustrates the viability of an economic road safety monitoring and assessment solution through exploiting advances in the Internet of Things (IoT) within the context

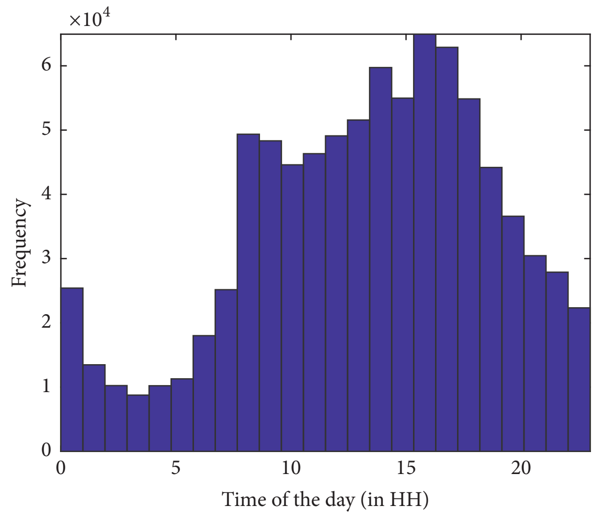


Figure 9: Collision distribution in time for the New York County street network and collisions for 2018 (January 1, 2018 onward).

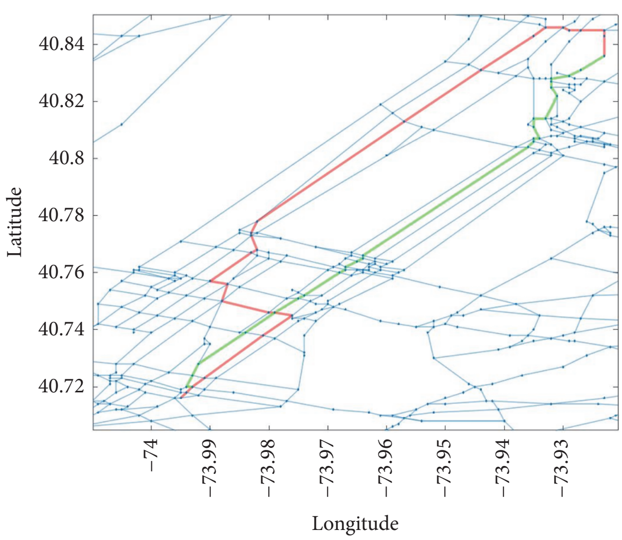


Figure 10: Route planning between two points on the New York County road network, with the red based on the shortest path, and the green based on safety.

of smart cities. The introduced architecture facilitates robust and dynamic road safety assessment that complements the Safe System approach motivated by the World Health Organization (WHO), which has been increasingly adopted worldwide. An application of the dynamic assessment framework for route planning is also demonstrated.

Future work involves exploring further applications, especially in the context of raising driver awareness of the road safety conditions during their trips.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper contains the results and findings of a research project that is funded by King Abdulaziz City for Science and Technology (KACST) Grant No. GDRG – 37-278.

References

[1] World Health Organization, "Global status report on road safety 2015," https://www.who.int/violence_injury_prevention/road_safety_status/2015/en/.

[2] World Health Organization, "Decade of Action for Road Safety 2011-2020 seeks to save millions of lives," http://www.who.int/roadsafety/decade_of_action/en/.

[3] F. Wegman, "The future of road safety: A worldwide perspective," *IATSS Research*, vol. 40, no. 2, pp. 66–71, 2017.

[4] World Health Organization, "Save LIVES - A road safety technical package," 2017.

[5] W. E. Marshall, "Understanding international road safety disparities: Why is Australia so much safer than the United States?" *Accident Analysis & Prevention*, vol. 111, pp. 251–265, 2018.

[6] "Open Street Maps, with New York County highlighte," https://www.openstreetmap.org/relation/2552485.

[7] United States Census Bureau, "TIGER/Line® Shapefiles:Roads," https://www.census.gov/cgi-bin/geo/shapefiles/index.php?year=2018&layergroup=Roads.

[8] X. Wang, X. Wu, M. Abdel-Aty, and P. J. Tremont, "Investigation of road network features and safety performance," *Accident Analysis & Prevention*, vol. 56, pp. 22–31, 2013.

[9] European Road Assessment Program (EuroRAP), "European Road Safety Atlas," http://atlas.eurorap.org/.

[10] European Road Assessment Programme (EuroRAP), "Star Ratings," http://www.eurorap.org/protocols/star-ratings/.

[11] International Road Assessment Programme (iRAP), "iRAP," https://www.irap.org/.

[12] H. M. Hassan and H. Al-Faleh, "Exploring the risk factors associated with the size and severity of roadway crashes in Riyadh," *Journal of Safety Research*, vol. 47, pp. 67–74, 2013.

[13] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, "Internet-of-things-based smart environments: State of the art, taxonomy, and open research challenges," *IEEE Wireless Communications Magazine*, vol. 23, no. 5, pp. 10–16, 2016.

[14] Y. Mehmood, F. Ahmad, I. Yaqoob, A. Adnane, M. Imran, and S. Guizani, "Internet-of-Things-Based Smart Cities: Recent Advances and Challenges," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 16–24, 2017.

[15] AARONIA, "GPS Logger including Gyro / Tilt / Compass & Accelerometer," https://www.aaronia.com/products/spectrum-analyzers/gps-logger/.

[16] M. Farsi, K. Ratcliff, and M. Barbosa, "Overview of controller area network," *Computing and Control Engineering Journal*, vol. 10, no. 3, pp. 113–120, 1999.

[17] MUNIC (company website), https://www.munic.io/.

[18] Torque Pro (OBD 2 & Car), "Torque Pro (OBD2 & Car) Google Play Page," http://goo.gl/CWD5VT.

[19] J.-S. Jhou, S.-H. Chen, W.-D. Tsay, and M.-C. Lai, “The implementation of OBD-II vehicle diagnosis system integrated with cloud computation technology,” in *Proceedings of the 2013 2nd International Conference on Robot, Vision and Signal Processing, RVSP 2013*, pp. 9–12, Japan, December 2013.

[20] A. Mednis, A. Elsts, and L. Selavo, “Embedded solution for road condition monitoring using vehicular sensor networks,” in *Proceedings of the 2012 6th International Conference on Application of Information and Communication Technologies, AICT 2012*, Georgia, October 2012.

[21] M. Yamada, K. Ueda, I. Horiba, and N. Sugie, “Discrimination of the Road Condition Toward Understanding of Vehicle Driving Environments,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 2, no. 1, pp. 26–31, 2001.

[22] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, “The pothole patrol: using a mobile sensor network for road surface monitoring,” in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services (MobiSys ’08)*, pp. 29–39, Breckenridge, Colo, USA, June 2008.

[23] M. Valentino, C. Quiligotti, and L. Carone, “Branchial cleft cyst,” *Journal of Ultrasound*, vol. 16, no. 1, pp. 17–20, 2013.

[24] A. S. El-Wakeel, J. Li, M. T. Rahman, A. Noureldin, and H. S. Hassanein, “Monitoring road surface anomalies towards dynamic road mapping for future smart cities,” in *Proceedings of the 2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 828–832, Montreal, QC, Canada, November 2017.

[25] P. Mohan, V. N. Padmanabhan, and R. Ramjee, “Nericell: Rich monitoring of road and traffic conditions using mobile smartphones,” in *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems (SenSys ’08)*, pp. 323–336, November 2008.

[26] D. Al-Yaseen, “CrowdNav: Information dissemination system for traffic enhancement,” Tech. Rep., School of Computing, Queens University, Kingston, ON, Canada, 2012.

[27] S. El-Tawab, M. Abuelela, and G. Yan, “Real-time weather notification system using intelligent vehicles and smart sensors,” in *Proceedings of the 2009 IEEE 6th International Conference on Mobile Adhoc and Sensor Systems, MASS ’09*, pp. 627–632, China, October 2009.

[28] J. Paefgen, F. Kehr, Y. Zhai, and F. Michahelles, “Driving behavior analysis with smartphones: insights from a Controlled Field Study,” in *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia (MUM ’12)*, 36 pages, ACM, December 2012.

[29] A. Ghose, P. Biswas, C. Bhaumik, M. Sharma, A. Pal, and A. Jha, “Road condition monitoring and alert application: Using in-vehicle smartphone as internet-connected sensor,” in *Proceedings of the 2012 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2012*, pp. 489–491, Switzerland, March 2012.

[30] S. Raj, A. Jain, and R. Misra, “Poster: Smartphone sensing for large data set collection of potholes,” in *Proceedings of the 11th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys 2013*, pp. 517-518, Taiwan, June 2013.

[31] M. Imprialou and M. Quddus, “Crash data quality for road safety research: Current state and future directions,” *Accident Analysis & Prevention*, 2016.

[32] N. Abuali and H. Abou-Zeid, “Driver behavior modeling: Developments and future directions,” *International Journal of Vehicular Technology*, vol. 2016, Article ID 6952791, 2016.

[33] N. Arbabzadeh and M. Jafari, “A Data-Driven Approach for Driving Safety Risk Prediction Using Driver Behavior and Roadway Information Data,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 446–460, 2018.

[34] C. Bo, X. Jian, X.-Y. Li, X. Mao, Y. Wang, and F. Li, “You’re driving and texting: Detecting drivers using personal smart phones by leveraging inertial sensors,” in *Proceedings of the 19th Annual International Conference on Mobile Computing and Networking, MobiCom 2013*, pp. 199–201, USA, October 2013.

[35] H. L. Chu, V. Raman, J. Shen, R. Roy Choudhury, A. Kansal, and V. Bahl, “Poster: You driving? Talk to you later,” in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services, MobiSys’11 and Co-located Workshops*, p. 397, USA, July 2011.

[36] G. Castignani, T. Derrmann, R. Frank, and T. Engel, “Driver behavior profiling using smartphones: A low-cost platform for driver monitoring,” *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 91–102, 2015.

[37] S. Singh, S. Nelakuditi, R. Roy Choudhury, and Y. Tong, “Your smartphone can watch the road and you: Mobile assistant for inattentive drivers,” in *Proceedings of the 13th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc’12*, pp. 261-262, USA, June 2012.

[38] R. Bhandari, R. Bhaskaran, and N. Venkata, “FullStop: Tracking unsafe stopping behaviour of buses,” in *Proceedings of the In Conference on Communication Systems Networks*, pp. 65–72, 2018.

[39] D. A. Johnson and M. M. Trivedi, “Driving style recognition using a smartphone as a sensor platform,” in *Proceedings of the 14th IEEE International Intelligent Transportation Systems Conference (ITSC ’11)*, pp. 1609–1615, Washington, DC, USA, October 2011.

[40] J. E. Meseguer, C. T. Calafate, J. C. Cano, and P. Manzoni, “DrivingStyles: A smartphone application to assess driver behavior,” in *Proceedings of the 18th IEEE Symposium on Computers and Communications, ISCC 2013*, pp. 535–540, Croatia, July 2013.

[41] S.-H. Chen, J.-S. Pan, and K. Lu, “Driving Behavior Analysis Based on Vehicle OBD Information and AdaBoost Algorithms,” in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2015.

[42] P. Smyth, D. Heckerman, and M. I. Jordan, “Probabilistic independence networks for hidden Markov probability models,” *Neural Computation*, vol. 9, no. 2, pp. 227–269, 1997.

[43] S. Djahel, R. Doolan, G.-M. Muntean, and J. Murphy, “A Communications-Oriented Perspective on Traffic Management Systems for Smart Cities: Challenges and Innovative Approaches,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 125–151, 2015.

[44] D. J. Rose and R. A. Willoughby, “University of Florida (UF) David Gleich’s Sparse Matrix Page,” https://www.cise.ufl.edu/research/sparse/matrices/Gleich/.

[45] “New York City OpenData Portal, NYPD Motor Vehicle Collisions,” https://data.cityofnewyork.us/Public-Safety/NYPD-Motor-Vehicle-Collisions/h9gi-nx95.

[46] “Matlab from Mathworks (software),” https://www.mathworks.com/products/matlab.html.

[47] PCI Geomatics (software), “PCI Geomatics (software),” http://www.pcigeomatics.com/software/geomatica/professional.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 1347967, 15 pages
https://doi.org/10.1155/2018/1347967

Research Article

Power Profiling of Context Aware Systems: A Contemporary Analysis and Framework for Power Conservation

Umar Mahmud,[1] **Shariq Hussain**,[1] **and Shunkun Yang**[2]

[1] *Department of Software Engineering, Foundation University Islamabad, Pakistan*
[2] *School of Reliability and Systems Engineering, Beihang University, Beijing, China*

Correspondence should be addressed to Shunkun Yang; ysk@buaa.edu.cn

Received 14 March 2018; Revised 18 July 2018; Accepted 29 August 2018; Published 16 September 2018

Academic Editor: Mubashir H. Rehmani

With the advent of smart, inexpensive devices and a highly connected world, a need for smart service discovery, delivery, and adaptation has appeared. This interconnection is composed of sensors within devices or placed externally in the surrounding environment. Our research addresses this need through a context aware system, which adapts to the users' context. Given that the devices are mobile and battery operated, the main challenge in a context awareness approach is power conservation. The devices are composed of small sensors that consume power in the order of a few mW. However, their consumptions increase manifold during data processing. There is a need to conserve power while delivering the requisite functionality of the context aware system. Therefore, this feature is termed as 'power awareness.' In this paper, we describe different power awareness techniques and compare them in terms of their conservation effectiveness. In addition, based on the investigations and comparison of the results, a power aware framework is proposed for a context aware system.

1. Introduction

Context awareness is termed as the awareness of a system to external and internal stimuli gathered through internal and external sensors. Such a system classifies this as an activity to adapt the services present within an environment. A context aware system is typically found in a handheld device, such as a smart phone [1–3]. Context is simply the description of the current situation as bound by the environment, while context awareness utilizes the context to adapt and change the services present in the environment. The environment becomes smart and delivers better services to the end-user [4]. A simple example is the change in orientation of a handheld device from portrait to landscape, based on how a person holds the phone.

Among various issues and challenges of context awareness is the management of battery power [5]. Battery power is used by different sensors that generate data. For instance, a device orientation sensor generates orientation data, whereas an accelerometer generates acceleration data. This data is then processed by the context aware system to classify the current context or situation. These sensors typically consume low power in the order of 1-2 mW. However, they consume 180–300 mW when processing is carried out [6, 7]. This significant increase in consumption requires sufficient power conservation such that the functionality or effectiveness of a context aware system is not reduced. The term 'power conservation' is interchangeably used as 'energy conservation' in the literature. This paper uses the former terminology. Power is related to energy, wherein the use of power over a period of time is considered energy consumption. The reduction in the use of energy for the same task is considered as power conservation.

This paper presents a literature review of different techniques of power conservation that have been proposed by researchers over the years. The literature, in general, considers power and energy to be synonymous. However, this paper follows the convention of power conservation. Furthermore, we discuss the effectiveness of these techniques and present a framework for power conservation. We describe two approaches that can be used to achieve substantial power conservation in context aware systems: (a) network communication-based conservation and (b) context processing-based conservation.

The rest of the paper is organized as follows: Section 2 discusses the related work. Section 3 gives an introduction to context awareness. Section 4 presents an overview of power consumption profiling. In Section 5, first we provide a brief discussion on power awareness and then compare the contemporary power conservation techniques in terms of their capabilities and effectiveness. Section 6 describes the proposed framework for power conservation applicable in context aware systems. Finally, Section 7 provides the concluding remarks.

2. Related Work

A number of researchers have investigated various approaches to model, analyze, and evaluate the efficiency of power consumption and have proposed diverse conservation techniques. This section reviews several closely-related investigations and provides a survey of various power optimization methods, as summarized in Table 1. The available literature can be categorized based on the focus areas. These areas are handheld mobile and cloud-based power optimization, application power optimization, and network communication power optimization.

The device power optimization method considers the device and provides means to conserve battery power within the device. Naik [8] claimed that the size and weight of the handheld mobile devices are the key constraining factors. This opens avenues for software based energy efficiency. The author provided a survey of different software based techniques, including sleep mode support, energy efficient GUI design, and power efficient communication methods. However, this survey neither provides solutions at the device level nor considers context aware systems. Ravindarnath [9] and Arun [10] have independently provided similar surveys for mobile and cloud computing based devices. Li et al. [11] has provided a survey of cloud power optimization based on the tasks given to a cloud system. Mittal [12] has surveyed various power conservation techniques for embedded systems. These embedded systems conserve power as a whole.

Jofri et al. [13] surveyed energy-aware profiling approaches for mobile devices and placed them into five classes and discussed their interrelationships. Power conservation of applications is one of them. Bernal et al. [14] argued that developing a power efficient context aware system is challenging, as it depends on the context sensor configuration. The authors have used an energy profiler to measure the power consumption of context aware applications and have provided recommendations for sensor configuration. Rault et al. [15] have published a survey of energy efficiency in wearable sensors. The authors argued that continuous sampling and communication deplete the battery power fast. They surveyed the existing energy efficient approaches in context aware activity recognition for healthcare applications. Ismail et al. [16] presented a survey of multimedia content adaptation techniques with respect to power consumption. The amount of power utilized in a multimedia content view is high, which can be adjusted based on the device's battery power. The survey neither considered context awareness nor discussed application level fine-tuning for power conservation. Wang et al. [17] argued that energy can be managed efficiently both at the hardware level and at the software level. The authors have selected DBMS for energy profiling and subsequently recommend various conservation measures.

The noisy nature of mobile networks has led to a high network power consumption. This provides a basis for surveys for power optimization in network communication. Brienza et al. [18] have outlined a power efficiency survey for P2P systems. This survey is focused on file distribution and content streaming features of P2P applications. Bolla et al. [19] published a survey of power efficiency in networks. The leading source of power consumption in mobile devices is network communication. However, the authors did not consider the energy overhead involved in mobile processing. Celenlioglu et al. [20] have published a survey of vertical handoff in 3G and WLAN devices.

3. Context Awareness

The objective of context awareness is to provide better services and applications to the users [21–24]. This includes discovering services among available options within an environment as well as fine-tuning the selected services. Thus, the context describes a complete situation wherein multiple users interact with different services bounded by a common environment [25–30]. The context process has five phases as shown in Figure 1. The phases are context aggregation, context representation, context history, context inference, and service adaptation [26].

3.1. Context Aggregation Module. Initially, the context is gathered from the sensors present in the environment and on board of the devices. Each sensor provides context values that are then aggregated as the contextual data captured at an instant. The contextual data gathered is raw and may need further processing to convert it into a meaningful form [31, 32]. Traditionally, this process is concerned with invoking the external sensors present in the environment as well as collecting data from the internal sensors present on the device. An associated controlled vocabulary is also provided to identify the same context attributes but with different names.

3.2. Context Representation Module. The contextual data is then stored as per the data organization of the context aware system, which is referred to as the current context [33]. Among various mechanisms, ontology is the preferred representation technique [34]. Ontology is based on eXtensible Markup Language (XML). It has the added advantage of providing a heterogeneous method of storage so that a complete context can be shared across diverse platforms.

3.3. Context History Module. The task of this module is to maintain the history of contexts associated with the appropriate labels. The history is generally maintained as records in an SQL table, which can be converted to No-SQL using

Table 1: Surveys for power optimization.

Ref	Contents	Objects	Focused Techniques
[8]	Device power optimization	Handheld mobile devices	Software based energy efficiency
[9]	Enhancing energy efficient	Mobile and cloud computing based devices	Energy aware offloading techniques
[10]	Minimizing energy consumption	Mobile cloud computing	Effective task scheduling method
[11]	Cloud power optimization	Cloud applications	Energy consumption models
[12]	Power conservation	Embedded systems	Power management techniques
[13]	Energy efficiency	Mobile devices	Energy-aware profilers
[14]	Power consumption	Context aware applications	Energy profiler for sensor configuration
[15]	Energy efficiency	Wearable sensors/healthcare application	Approaches for context aware activity recognition
[16]	Power consumption	Multimedia	Content adaptation techniques
[17]	Energy profiling	Software and hardware level	DBMS
[18]	Power consumption and optimization	P2P/Network communication	File distribution and content streaming
[19]	Power efficiency	Networks/mobile devices	Power consumption in network communication
[20]	Energy efficiency	3G and WLAN devices	Vertical handoff algorithm

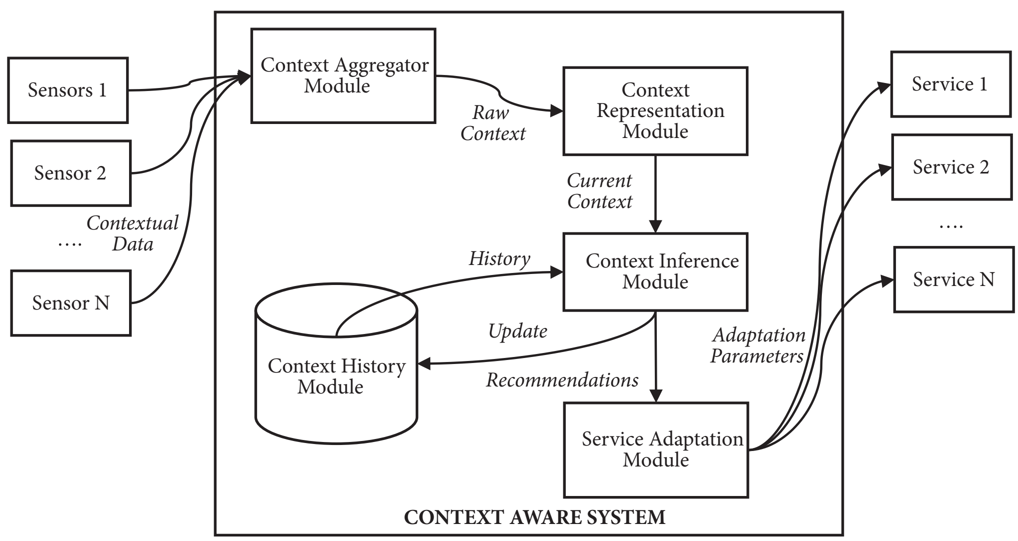


Figure 1: System architectural of a context aware system.

XML. This history information can be purged to reduce the size as well as to identify the user preferences [35].

3.4. Context Inference Module. Through this module, the activity or situation is classified through a context inference mechanism, and labels are given to current contexts [36]. This can include multiple overlapping activities within an environment as well multiple role-based interactions by each user. Here, recommendations for better service discovery and delivery as well as adaptation are made. The context inference mechanism can be designed as rule based or classification-based [37]. The software engineer is thus concerned with selecting and improving known classification algorithms to achieve context awareness.

3.5. Service Adaptation Module. Finally, by this module, the services are contacted and adapted as per the recommendations of the inference phase. This has two aspects: (a) service discovery and delivery: this is concerned with finding and selecting better services for the user and (b) service adaptation: this deals with changing the parameters of a selected service as per the current context. The service could be an external web service that can be invoked upon the change in the state of the current context. The selection of web service may employ certain criteria for QoS-based web service selection [38, 39].

4. Power Consumption Profiling

Power conservation optimization is a challenging area that requires novel approaches to solve the underlying issues. Power is a vital resource for battery-operated systems. Recent technological advancements have paved the way for miniaturization in terms of size and weight of such systems. However, power consumption remains to be a dominant issue in the design process. Both aspects of a system, i.e., software and hardware, are to be considered in evaluating the true power consumption. Typically, the hardware part is considered to be more resource hungry. Very little attention is

Table 2: Comparison of power profiling techniques.

Power profiling		Advantage	Disadvantage	Requirements/suitability
Power consumption profiling methods	Model-based	Easy to implement	Affected by model's accuracy	Suitable at design times
	Real-measurement based	Real time prediction for dynamic power consumption	High overhead for tracking system events	Suitable for process level
Power consumption profiling finesse	Component level	Produce more accurate results	Requires detailed specifications	Suitable for device vendors at design times
	Device level	Easy to implement	Appropriate way	Usage patterns need to be identified

usually given on software development where the operating algorithms and programming styles may influence the overall power consumption of the system [40].

Apart from the hardware, limited considerations have been given to optimization of software applications [41–43]. For power optimization, it is necessary to correctly assess the power usage of the software components of a system. A number of approaches have been presented in the literature that focus on profiling the techniques in order to optimize the power consumption of the software applications [42–47]. Abkenar et al. [48] have employed a mechanism to effectively measure the consumption during a group activity recognition by monitoring the battery drain of a mobile device.

Thermal aspects are also of high interest while profiling the power consumption, as these can offer useful insights. Marcu et al. [49] used thermal profiling and defined benchmarks to identify the impact of the CPU work load (i.e., the executed applications) using the thermal response of the CPU cores. However, this approach is not generic enough, as it is only applicable to a specific test-bed. Profiling is not a trivial task as it involves identifying miniature-level functional relationships and associated power consumption among the complex processes. Duan et al. [50] proposed a power estimation model for mobile devices that considers the interconnected relationship among the three layers, i.e., the hardware, the operating system, and the application under execution. To get the maximum device uptime, optimum resource utilization, in terms of battery energy, is crucial. Hence, true power profiling is the only critical success factor in accurate lifetime prediction.

Prior to understanding energy profiling, the power consumption states used by the devices for power management need to be determined. A brief overview of the power consumption states is presented in the following section, and different power profiling techniques are compared in Table 2.

4.1. Power Consumption States. Generally, three states are available in modern devices to support power conservation: (a) 'active' or 'working,' (b) 'idle' or 'low power,' and (c) 'suspend' or 'sleep.' These states correspond to the device, and do not scale-down to the processes or applications running on the device. These states are shown in Figure 2. Their descriptions are given below:

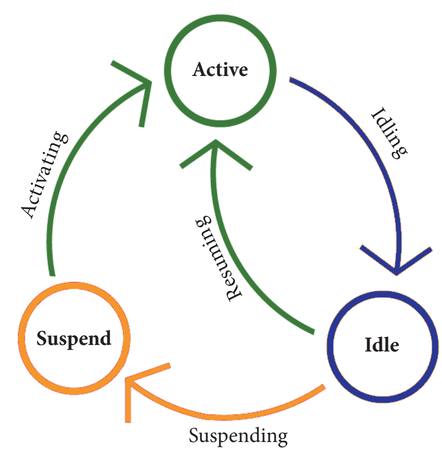


Figure 2: Power consumption states.

(i) **Active (Working):** In this state, the device is fully functional and is processing the user's tasks. Power consumption in this state depends on the usage scenario and involvement of various components that are associated with the task. These components can be the internal sensors present on the device.

(ii) **Idle (Low Power):** When there are no running tasks, the device is switched to idle mode and the core components as well as the associated peripheral components are switched to minimum energy levels. Only the basic functionality of a mobile device is available, e.g., sending and receiving messages or phone call.

(iii) **Suspend (Sleep):** In suspend mode, all unused components are switched-off to save the power. Only the core components are kept alive at predefined minimum energy levels. Only a minimal amount of activity is allowed with which the device can perform the essential tasks.

4.2. Power Consumption Profiling Methods. Generally two methods are considered for power consumption profiling: (a) model-based and (b) real measurement-based [51]. These methods are discussed below:

(i) **Model-Based:** In model-based method, a model is built to analyze the power consumption of the events in the system. Then, through a series of measurements, the power consumption for each of the system's component is determined, and the model is calibrated based on the measured data. Thus, the power consumption for the whole system can be evaluated by monitoring the rendering of the events through profiling. This method is easy to implement. However, the model's accuracy is a key factor in a true evaluation of the power consumption.

(ii) **Real Measurement-Based:** In a real measurement-based method, the power consumption of the system is measured in real-time. The power consumption profile of the system is generated after harmonizing the power data with the system events. This scheme is useful to predict the dynamic power consumption of the system. However, it requires a higher degree of synchronization between both entities, i.e., power records and event records. Another concern while tracking the system events is that it creates a high overhead. Due to this reason, most of the power consumption profiling techniques, as proposed by the researchers to date, focus on the process-level power consumption.

4.3. Power Consumption Profiling Finesse. The finesse of power consumption profiling is broadly based on either coarse-grained or fine-grained methods. The coarse-grained method considers the overall device, while the fine-grained method considers only the processes on the device. The power consumption profiling methods can be applied either at the component level or the device level [52].

(i) **Component Level method:** Under this method, the power consumption for each component of the system is measured independently. By summing up the power consumption values for different components, the total power consumption of the system can be measured. This method produces more accurate results. However, it requires detailed specifications of both the hardware and the software for a particular component. The device vendors can benefit from this approach, as they have detailed specifications and can possibly implement this at the design stage.

(ii) **Device Level method:** This method, to be more appropriate, can be termed as the device usage scenario, as it measures the power consumption of a device for different use-cases [53]. This method is easy to implement, and is widely used in power consumption profiling. For this method, two considerations need to be considered. Firstly, the number of required test runs need to be optimized to achieve coherence in results. Secondly, the usage patterns need to be identified in order to synchronize the power consumption with the use-cases. An example of possible usage scenarios in mobile devices where a user can use any number of applications concurrently is shown in Figure 3.

5. Power Awareness

Among the issues within context awareness, power conservation is of prime importance. Since a context aware system can be mobile, and it relies on the internal sensors present on the device as well as the external sensors present in the environment, there is a need to conserve power. Furthermore, the context recognition is itself a power-consuming task. Once the context is gathered, it must be processed, and processing consumes power [54].

Depending on the classification approach, the algorithms employed have different power profiles. For example, some algorithms, such as artificial neural network (ANN) and genetic algorithm (GA), train prior to classification and have higher training times. Algorithms, such as K-Means and K nearest neighbors (KNN), have zero training but maximum classification times [55]. The design of algorithms allows the scientists to calculate the time and space complexity only. Currently, no convention exists to measure the power conservation of an algorithm. Context aware applications consume power in three phases: context aggregation, context processing, and context adaptation.

Power awareness is a mechanism through which a context aware system can optimize itself to conserve power while providing acceptable functionality. Researchers have proposed various techniques to effectively conserve the power of a context processing mobile device as discussed in this section. Most of these recommended techniques are based on the power consumption profile of a mobile device. Without power consumption, the power conservation technique is useless.

5.1. Contemporary Power Conservation Techniques. In the following sections, various techniques to monitor the power are discussed. These techniques are primarily based on investigations performed by different researchers.

5.1.1. Threshold-Based Conservation. This technique is a simple rule where a threshold is preset for power conservation, e.g., the activation of power saving mode at 15% battery power. This technique, on a whole, is applied to mobile devices. In the power saving mode, the data connections are cut-off. This results in longer battery life—at the cost of context awareness.

Threshold-based conservation, though being the simplest, has the lowest intelligence as the threshold is preset in factory configurations. This mechanism does not adjust the power conservation to the application and is used at the device level.

5.1.2. Demand-Based Conservation. An improvement over the threshold-based technique is to adaptively adjust the threshold based on the demand. The system must now proactively predict the demand and conserve when the demand is low. This is possible at the operating system (OS) or at the device level, where the system predicts the overall demand and adjusts the threshold. The power awareness at the OS level can be termed as 'battery-aware' as it considers the

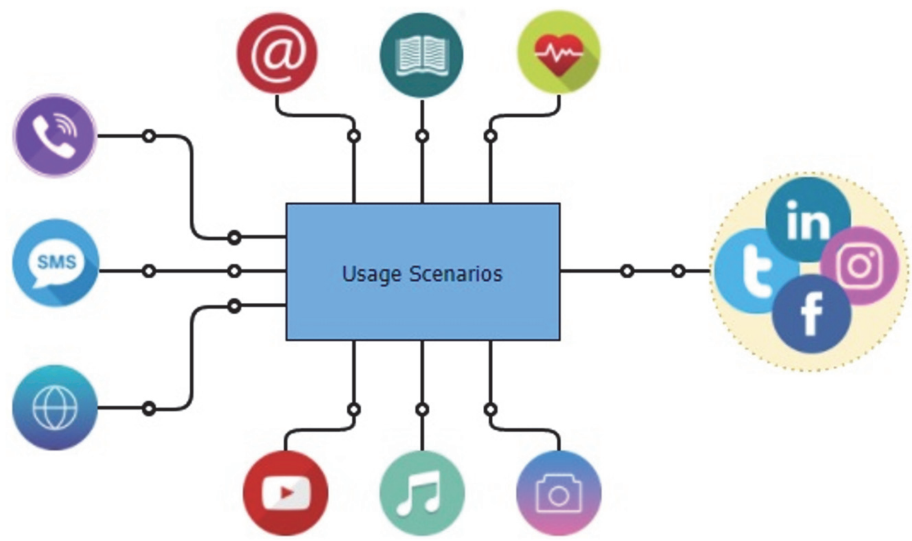


FIGURE 3: Usage scenarios for power consumption at the device level.

discharge management by the complete system rather than the applications [56]. This technique is implemented as a rule based system (RBS) using mapping and is limited to fine-tuning at the application level [57, 58].

Elmalaki et al. have pointed out that using the OS layers for sensor communication as well as context processing would consume less power as the layers (and subsequently the number of lines in the code) are reduced [59]. The researchers further placed the context awareness layers within the Android framework of a smart phone. The operating system must perform power consumption profiling on all applications, i.e., at the device level.

Moghimi et al. approached the problem by creating a Fuzzy RBS based power manager [60]. The power manager is responsible for inferring the rate of sensor access. For example, if the smart phone is idle then the update rate could be reduced to an hourly basis.

Fei et al. have proposed the use of software based dynamic power management as a trade-off between power conservation and an application's Quality of Service (QoS) parameters [61]. They have identified the power requirement to be a measure of the quality of applications.

A demand-based conservation technique cuts off all processing when the threshold is reached just like the threshold-based conservation. Therefore, it lacks fine-tuning at the application level. The implementation as a rule based system, though simple, has its challenges due to increases in the rule space. The system must measure the current conservation through software-hardware interfaces, which may be propriety [62].

5.1.3. Cloud-Based Conservation. Some researchers have proposed the use of cloud as an alternative to using local architecture for context awareness [6, 63]. This mechanism effectively conserves power by not using the local system for processing and awareness-oriented tasks. A broker is implemented as a centralized middleware, which performs context processing over the cloud if the threshold is reached. However, energy profiling for cloud environments is another challenge. A number of researchers have proposed various approaches to address this issue [64, 65]. In our work, we consider power profiling for mobile devices only and leave the cloud environment for future work.

This technique does not use the local resources for context processing. Thus, it conserves power by utilizing the cloud. The challenge is to adaptively establish a threshold after which the computations are sent over the cloud. In effect, this technique is a load balancer. The power consumption profile has higher power consumption cost for cloud communication as compared to the power consumption cost of context processing. The mobile devices only use the internal sensors and communicate with the external sensors present in the environment as well as the cloud.

5.1.4. Context Change-Detection Based Conservation. The context is classified by continuous monitoring of the contextual data in a context aware system. This data is acquired from internal and external sensors as discussed in Section 3. Kang et al. have proposed proactive monitoring on the basis of change detection [66]. The sensors are continuously monitored but the context processing is suspended until a change is detected. The context processing is now carried out proactively if the change in sensor data results in change of context.

This technique though fine-tunes conservation at the application level requires a method to predict the change in the context a priori. An associated history manager must be provided in the context aware module of the complete system.

5.1.5. Statistics-Based Conservation. Sathan et al. have proposed the use of statistics to predict when to poll sensors [67]. The effect of continuous sensing is a constant drain on the battery. If the change in sensor can be predicted, the sensors can be invoked in a noncontinuous manner. The context processing time is also reduced. An innovation is to provide a sensor layer between the sensor and the context gathering module. The sensor layer could easily predict the sensor change based on the history of accesses.

Yuryur proposed a lightweight online and unsupervised mechanism to classify the context using the accelerometer data [7]. To conserve power, a Markov reward process is used to profile the power consumption. Yuryur has identified that a 1 mW accelerometer consumes 370 mW power when the context is being processed. The reward process monitors the power consumption as triggered by a change in the sensor data.

This technique requires a priori information to effectively predict when to poll the sensors, and when to process the context if continuous data is provided from the sensors. If the classification of the context is carried out as a predictive algorithm, it would support the power awareness task. Otherwise the system would use two different learning algorithms, i.e., one for classification of the context and other for sensor polling [35, 37].

5.1.6. Selective Sensor Access-Based Conservation. The context classification tasks require data from the sensors present in the environment as discussed in Section 3. The sensors correspond to variables, which are provided to the context aware system. Hermann et al. argued that the classification of all context situations does not require the use of all variables. For instance, to identify the activity being performed in a cafeteria might require the location and time values but may not need the outside barometric pressure value for correct classification [68]. The researchers propose the use of selective sensors for the classification purposes. If an accurate classification cannot be achieved, then further sensors can also be invoked. The context aware system then starts, suspends, and changes the sampling rate of the sensors according to the user's known context.

This technique demands a minimum list of sensors that must be used for classification of the context. The number of sensors, both internal and external, may be large. In addition, not all of them need to be invoked for a target context. The list must be made adaptive to include unknown contexts having different minimal list of sensors.

5.1.7. Source Code Optimization-Based Conservation. In addition to active power conservation techniques presented in above sections, researchers have suggested that source code optimization is another key approach for power conservation. This technique is based on the concept of power aware software development and considers both the programming languages as well as the associated compilers.

Compiler-based optimizations were given little considerations as it has meagre effect in power consumption optimization on the software. Several methods can be identified from the literature that target power conservation by fine-tuning the compilers in terms of instruction rearrangements, optimized use of memory banks, and reduction of the operands in an instruction to a minimum [69–72].

Apart from compiler optimizations, methods such as loop unrolling, software pipelining, recursion elimination, and algorithm evaluation may also prove to be power aware. However, their effectiveness need to be evaluated under a given problem-set [73]. Further, power conservation for the software depends on coding styles, selection of language constructs, choice of algorithms, and selection of data types. Capra et al. compared a number of open source software for power consumption in terms of design and functional aspects and also examined the effects of the application development environments' usage on power conservation [74].

Hao et al. proposed a technique for energy consumption in mobile devices based on instruction-level energy modeling. Many researchers have proposed models for energy consumption either for some specific components, or for all components of a mobile device [75].

Tsao et al. focused on I/O functions in mobile devices and analyzed the energy consumption of I/O events [76]. All these factors exhibit an association with the source code-based power conservation techniques.

Source code optimization-based conservation is carried out during compile time. Developers create less lines of code that yield the same outcome allowing less power consumption during processing. This technique is cumbersome as the development effort is increased many folds.

5.2. Comparison of Contemporary Techniques. In order to compare the representative contemporary techniques comprehensively, we develop a framework that consists of the following attributes:

(i) **Intelligence:** A technique exhibits intelligence if it has a learning mechanism or at least is better than a simple threshold-based systems. The comparison does not consider the algorithm used for power awareness.

(ii) **Adaptive threshold:** This refers to establishing a dynamic threshold that can be changed as opposed to factory configurations. This requires a better intelligence to adaptively select when to change the power state of a device. Adaptive threshold is a smarter version of the threshold-based conservation technique.

(iii) **Network communication power conservation:** The context aware system communicates with external sensors present in the environment mostly through wireless links. Wireless links being noisy lead to multiple retries for communication, and thus power is wasted. A technique heavily dependent on the network needs to conserve power for network communication.

(iv) **On-device context Processing:** Whether the context classification or context processing is carried out on a mobile device or some other middleware is used. A cloud-based technique uses the external servers while all other methods utilize the device for context processing.

(v) **Cut-off all applications:** This refers to a device level conservation method, and does not adapt to application level adjustments. Cutting off all applications effectively halts the context aware processing as well as all other processing in favor of the basic or minimal tasks on the device.

(vi) **Rule based system:** This method refers to using simple rules for power conservation where rules are prewritten in the applications as well as in the device. A rule based system is efficient in classification, but it is resource-heavy due to the need for large rule space. The designers have suggested rule based systems for small rule spaces.

(vii) **Load balancing:** This is primarily used where external processing is utilized as in the case of cloud-based conservation. The system thus balances the load of context processing on the device as well as on the external middleware.

(viii) **Use of history:** History is used for predictive conservation as well as to identify a priori information. A history module is part of the context aware system as discussed in Section 3. Using history information for power awareness is considered in this facet.

Based on the aforementioned attributes, a comparative matrix for contemporary power conservation techniques is presented in Table 3. The goal of the comparison is to identify supportive and non-supportive criterion for each technique.

A detailed comparison is presented in Table 4, where factors, such as advantages, disadvantages, requirements/suitability, and technical aspects, are included. We hope the comparison will provide useful insights for each conservation technique.

6. Framework for Power Awareness in Context Aware Systems

In this section, we present the proposed framework for power awareness and describe its characteristics. The motivation behind building such a framework is to provide power optimization in context aware systems in a power aware fashion. The proposed framework is illustrated in Figure 4. Awareness is achieved once the power consumption is modeled. The power conservation can then be enhanced. Power conservation can be achieved in processing, storage, and communication phases that comprise a context aware system.

6.1. Power Consumption Model. This section discusses how power is consumed and utilized in every module of a context aware system. This will allow us to better understand the power requirements and to propose conservation tactics at different levels.

6.1.1. Power Consumption in Context Aggregation Module. The context aggregation module is responsible for communicating with internal sensors as well as external sensors present in the environment as discussed in Section 3. The power consumption of external and internal sensors is of the order of a few mW. However, the processing consumes much more power [6]. For external sensors, the power is also consumed during network communication. Being wireless nodes, the number of retries due to noisy channels and dense networks is large. Hence, a number of retries in communication can increase the power consumption many fold. This module does not perform any processing, and simply organizes the current context as an attribute-value tuple.

Denoting the power consumption by the sensors as p_s, the power consumption by the network as p_{nw}, the number of retries as t, and the number of sensors as n, the power consumption of context aggregation module P_{cam} is given by (1). For all internal sensors, the power consumption by the network is zero.

$$P_{cam} = \sum_{i=1}^{n} \left(p_s + \left(p_{nw} \times t\right)\right) \quad (1)$$

6.1.2. Power Consumption in Context Representation Module. The raw current context as acquired by context aggregation module is then processed and represented in a hierarchical and platform independent mechanism [53]. This also involves some processing to convert raw context to a meaningful form. As an example, the temperature in Celsius is to be listed as 'Warm' or 'Cold.'

Denoting the power consumed in converting a single value to the appropriate form as p_{cnv} and the power consumed in processing low level task as p_{lt}, there are n sensors and the power consumed by the representation module p_{crm} is given by

$$P_{crm} = \sum_{i=1}^{n} \left(p_{cnv} + p_{lt}\right) \quad (2)$$

6.1.3. Power Consumption in Context History Module. The current context and the previously known states associated with the classification state are kept in a data store. Thus, the consumed power information is accessed into the history. For the systems where predictive or statistical power awareness is not carried out and the history is absent, the power consumed is thus zero.

Denoting the power consumed in a single record access of history as p_{accrec} and the number of records as r, the power consumed by the context history module P_{chm} is given by

$$P_{chm} = \sum_{i=1}^{r} p_{accrec} \quad (3)$$

6.1.4. Power Consumption in Context Inference Module. The main task of context inference module is to process the current context and classify it as a context situation or activity. Being the core of a context aware system, maximum amount of processing is carried out in this module. Therefore, maximum power is also consumed by this module. The power consumed in this module is primarily dependent on the classification technique. The techniques have two general steps: (a) training and (b) classification. Some techniques, such as ANN and GA, have higher training times but are quick in classifying, whereas techniques such as KNN have low training time and high classification time. These general steps indicate the consumption of power during training as well as during classification phases. However, for a given technique only one step will be dominant.

Denoting the power consumed during training as p_{trg} and the power consumed during classification as $p_{classify}$, the

Table 3: Comparative matrix for power conservation techniques.

	Threshold-Based Conservation	Demand-Based Conservation	Cloud-Based Conservation	Context Change Detection-Based Conservation	Statistics-Based Conservation	Selective Sensor Access-Based Conservation	Source Code Optimization-Based Conservation
Intelligence	×	×	√	√	√	√	√
Adaptive Threshold	×	√	√	√	√	√	√
Network Energy Conservation	√	√	×	√	×	√	×
On Device Context Processing	√	√	×	√	√	√	√
Cut-off All Applications	√	√	×	×	×	×	×
Rule Based System	√	√	×	×	×	×	×
Load Balancing	×	×	√	×	×	×	×
Use of History	×	×	×	√	√	×	×

Table 4: Comparative overview of contemporary power conservation techniques.

Conservation Techniques	Advantages	Disadvantages	Requirements/Suitability	Technical Aspects
Threshold-Based Conservation	(i) Simple and easy to implement	(i) Lack of adaptability (ii) Device level conservation	(i) Battery operated devices that need to shut down in a stable manner	(i) Non-adaptive and hard threshold used for stable power-on and power-off procedures in battery operated devices
Demand-Based Conservation	(i) Adaptive threshold	(i) Device level conservation	(i) Battery operated devices that need to predict consumption based on current usage	(i) Rule based system (ii) Predictive analysis
Cloud-Based Conservation	(i) Use of cloud (ii) Adaptive threshold (iii) Reduction of power awareness overhead at device level (iv) Power conservation at application level	(i) High communication overhead	(i) Suitable for cloud assisted environment	(i) Cloud-based system (ii) Can allow classification techniques for adaptive thresholding
Context Change Detection-Based Conservation	(i) Adaptive threshold (ii) Context monitoring (iii) Proactive approach (iv) Power conservation at application level	(i) Context change detection is required a-priori	(i) Context aware systems	(i) Context awareness (ii) Context differential
Statistics-Based Conservation	(i) Adaptive threshold (ii) Predictive conservation (iii) Power conservation at application level	(i) History management overhead (ii) Statistical calculation overhead	(i) Low power consumption history management module	(i) Predictive analysis (ii) Regression
Selective Sensor Access-Based Conservation	(i) Selective sensor approach (ii) Power conservation at sensor level	(i) Minimal list of sensors for each activity classification (ii) Limited context classification scenarios	(i) Suitable for large number of sensors present in an environment	(i) Power conservation for energy hungry sensors
Source Code Optimization-Based Conservation	(i) Source code optimization keeping in view power constraint	(i) Cumbersome to implement (ii) Compile time conservation (iii) Non-adaptive thresholding	(i) Suitable for embedded systems (ii) Non-generic devices and applications	(i) Source code optimization (ii) Power consumption measurement for programming constructs and procedures

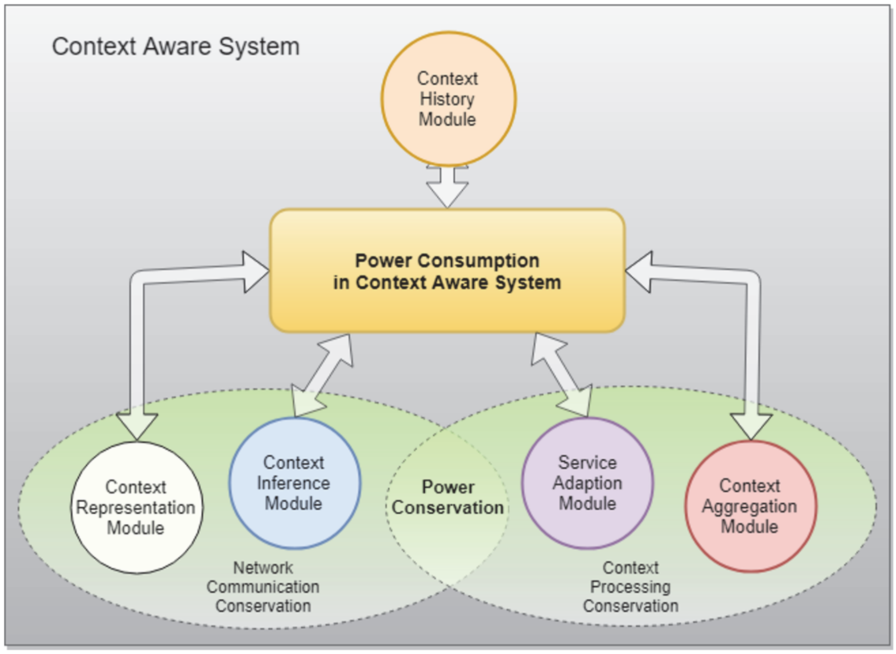


Figure 4: Framework for power awareness in context aware systems.

power consumed by context inference module P_{cim} is given by

$$P_{cim} = p_{trg} + p_{classify} \tag{4}$$

6.1.5. Power Consumption in Service Adaptation Module. This module adjusts both internal and external services present in the environment based on the recommendation from the context inference module. This module communicates with the external services over the network. Retries are required for successful transmission due to noisy and error-prone wireless network.

If the power consumed by the service is p_{svc}, the power consumed by the network for each external service is p_{nw}, the number of retries is t, and the number of services is v, the power consumed by the service adaptation module P_{sam} is given by

$$P_{sam} = \sum_{i=1}^{v} \left(p_{svc} + \left(p_{nw} \times t\right)\right) \tag{5}$$

6.2. Power Conservation Approach. To conserve power numerous methods and safeguards have been proposed. Based on the consumption characteristics, the power consumption centers within a context aware system are classified as network communication-based conservation and context processing-based conservation. A brief description of these is given below.

6.2.1. Network Communication-Based Conservation. The major usage of this method is in network communication, which is required in the context aggregation module and the service adaptation module. It is evident from (1) and (5) that the channel may result in a number of retries, which not only consumes energy but also takes more time. It should be pointed out that the cloud-based conservation techniques have more network utilization than the rest.

6.2.2. Context Processing-Based Conservation. The context representation module and the context inference module both consume power as part of the context processing. This is dependent on the hardware as well as the OS support. The cloud-based conservation techniques have low context processing as it is carried out over the cloud. The context processing is also algorithm dependent where the size of the data set increases the classification or training time and thus consumes more power.

7. Results and Discussions

In this section, we describe experimental setup used for evaluation of proposed framework then provide our observations as well as corresponding results. The tests are conducted on Dell Latitude E5430 and Samsung R450R4V machines. The tests simulates context data gathering, storage, and processing. The power consumption is measured as change in battery level. Object-oriented Java program using NetBeans version 8.2 is developed that measures the battery level through Java Native Access (JNA) library. The test results are summarized in Table 5, which shows the power consumption in percentage (%) and time elapsed in nanoseconds for context aware system. The power consumption is measured as the change in battery level of a test machine. The relationship between power consumption and time of execution is shown in

Table 5: Test results of power consumption and time elapsed for different machines.

Test No	Dell Latitude E5430		Samsung R450R4V	
	Battery Consumption Level (%)	Time Elapsed (nanoseconds)	Battery Consumption Level (%)	Time Elapsed (nanoseconds)
1	1	102871511678.00	1	107283040801.00
2	2	201146140580.00	2	202699872266.00
3	2	301197287625.00	2	305333596221.00
4	5	401126581201.00	3	402140583122.00
5	5	501171081067.00	4	505597135941.00
6	5	601158529665.00	5	604191542742.00
7	6	701601643796.00	6	702121513477.00
8	7	801798038474.00	6	802937759878.00
9	8	901387821639.00	7	908428301040.00
10	9	1005727418757.00	8	1002842365866.00

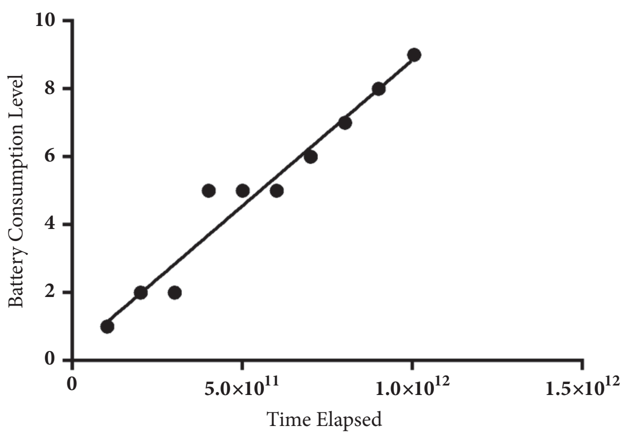


Figure 5: Linear least square line for battery consumption and time elapsed on machine no. 1.

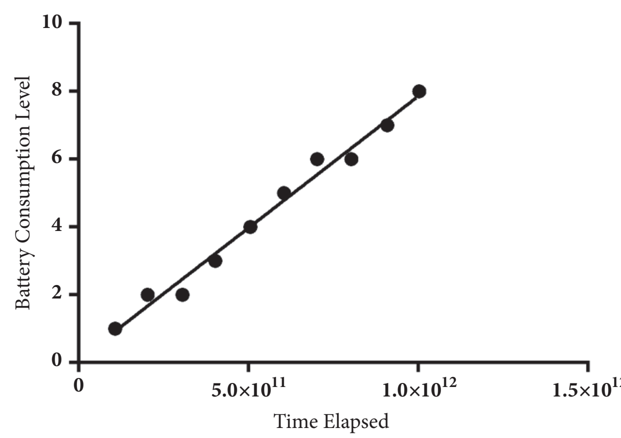


Figure 6: Linear least square line for battery consumption and time elapsed on machine no. 2.

Figures 5 and 6. The linear least square line has been calculated and is shown in

$$BatteryConsumptionLevel = 8.589^{-12} \times TimeElapsed + 0.25930 \quad (6)$$

$$BatteryConsumptionLevel = 7.762^{-12} \times TimeElapsed + 0.09720 \quad (7)$$

Both (6) and (7), as well as Figures 5 and 6, conform that the increase in activity that includes communication and processing causes a linear increase in power consumption. This observation conforms to the proposed framework in Section 6.1.

8. Conclusion

This paper presents a comparison of various contemporary power awareness techniques within the scope of context aware systems. Different techniques to achieve power awareness are classified in this work. The review also includes power consumption profiling to identify how power consumption can be measured. Several recommendations for power conservation are also provided that will help in identifying areas of improvement within a context aware system. A framework is also presented which models the power consumption at different stages of a context aware system. The results conform to the framework which shows that the power consumption is a linear function of execution time. This can then be optimized using network communication conservation and context processing conservation.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work reported in this paper was supported by the National Natural Science Foundation of China (Grant no. 61672080).

References

[1] N. A. Malik, U. Mahmud, and M. Y. Javed, "Future challenges in context aware computing," in *WWW/Internet 2007*, pp. 306–310, 2007.

[2] T. Strang and C. Linnhoff-Popien, "A context modeling survey," in *Proceedings of the Workshop on Advanced Context Modelling, Reasoning and Management (UbiComp '04)*, pp. 34–41, 2004.

[3] P. Daponte, L. De Vito, F. Picariello, and M. Riccio, "State of the art and future developments of measurement applications on smartphones," *Measurement*, vol. 46, no. 9, pp. 3291–3307, 2013.

[4] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, "A review of smart home applications based on Internet of Things," *Journal of Network and Computer Applications*, vol. 97, pp. 48–65, 2017.

[5] T. Cioara, I. Anghel, I. Salomie, G. Copil, B. Pernici, and D. Moldovan, "A context aware self-adapting algorithm for managing the energy efficiency of IT service centers," *Ubiquitous Computing and Communication Journal*, vol. 6, pp. 619–630, 2011.

[6] S. L. Kiani, A. Anjum, N. Antonopoulos, and M. Knappmeyer, "Context-aware service utilisation in the clouds and energy conservation," *Journal of Ambient Intelligence and Humanized Computing*, vol. 5, no. 1, pp. 111–131, 2014.

[7] O. Yuryur, *Energy Efficient Context-Aware Framework in Mobile Sensing [Ph.D. Thesis]*, University of South Florida, Florida, FL, USA, 2013.

[8] K. Naik, *A Survey of Software Based Energy Saving Methodologies for Handheld Wireless Communication Devices*, Department of Electrical and Computer Engineering, University of Waterloo, 2010.

[9] K. Ravindranath and K. R. S. Rao, "A survey on energy aware offloading techniques for mobile cloud computing," *International Journal of Computer Trends and Technology*, vol. 4, pp. 2081–2086, 2013.

[10] C. Arun and V. Jaiganesh, "Survey on minimizing energy consumption in mobile cloud computing," *International Journal of Computer Applications*, vol. 150, no. 3, pp. 5–8, 2016.

[11] Z. Li, S. Tesfatsion, S. Bastani et al., "A survey on modeling energy consumption of cloud applications: deconstruction, state of the art, and trade-off debates," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 3, pp. 255–274, 2017.

[12] S. Mittal, "A survey of techniques for improving energy efficiency in embedded computing systems," *International Journal of Computer Aided Engineering and Technology*, vol. 6, no. 4, pp. 440–459, 2014.

[13] M. H. Jofri, M. F. M. Fudzee, and M. N. Ismail, "A survey on energy-aware profiler for mobile devices," in *Advances in Intelligent Systems and Computing*, vol. 331, pp. 295–305, Springer, 2015.

[14] J. F. M. Bernai, L. Ardito, M. Morisio, and P. Falcarin, "Towards an efficient context-aware system: Problems and suggestions to reduce energy consumption in mobile devices," in *Proceedings of the 9th International Conference on Mobile Business/9th Global Mobility Roundtable (ICMB and GMR '10)*, pp. 510–514, June 2010.

[15] T. Rault, A. Bouabdallah, Y. Challal, and F. Marin, "A survey of energy-efficient context recognition systems using wearable sensors for healthcare applications," *Pervasive and Mobile Computing*, vol. 37, pp. 23–44, 2017.

[16] M. N. Ismail, R. Ibrahim, and M. F. Md Fudzee, "A survey on content adaptation systems towards energy consumption awareness," *Advances in Multimedia*, vol. 2013, Article ID 871516, 8 pages, 2013.

[17] J. Wang, L. Feng, W. Xue, and Z. Song, "A survey on energy-efficient data management," *ACM SIGMOD Record*, vol. 40, no. 2, pp. 17–23, 2011.

[18] S. Brienza, S. E. Cebeci, S. S. Masoumzadeh, H. Hlavacs, Ö. Özkasap, and G. Anastasi, "A survey on energy efficiency in P2P systems: File distribution, content streaming, and epidemics," *ACM Computing Surveys*, vol. 48, no. 3, 2015.

[19] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, "Energy efficiency in the future internet: a survey of existing approaches and trends in energy-aware fixed network infrastructures," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 223–244, 2011.

[20] M. R. Celenliôglu, D. Gözüpek, and H. A. Mantar, "A survey on the energy efficiency of vertical handover mechanisms," in *Proceedings of the International Conference on Wireless and Mobile Networks (WiMoN '13)*, 2013.

[21] U. Mahmud, N. Iltaf, A. Rehman, and F. Kamran, "Context-Aware Paradigm for a Pervasive Computing Environment (CAPP)," in *WWW/Internet 2007*, pp. 337–346, Villa Real, Portugal, 2007.

[22] M. Knappmeyer, S. L. Kiani, E. S. Reetz, N. Baker, and R. Tonjes, "Survey of context provisioning middleware," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1492–1519, 2013.

[23] M. Hashemi and A. Sadeghi-Niaraki, "A theoretical framework for ubiquitous computing," *International Journal of Advanced Pervasive and Ubiquitous Computing*, vol. 8, no. 2, pp. 1–15, 2016.

[24] U. Alegre, J. C. Augusto, and T. Clark, "Engineering context-aware systems and applications: A survey," *The Journal of Systems and Software*, vol. 117, pp. 55–83, 2016.

[25] U. Mahmud and N. A. Malik, "Flow and threat modelling of a context aware system," *International Journal of Advanced Pervasive and Ubiquitous Computing*, vol. 6, no. 2, pp. 58–70, 2014.

[26] U. Mahmud, "UML based model of a context aware system," *International Journal of Advanced Pervasive and Ubiquitous Computing*, vol. 7, no. 1, pp. 1–16, 2015.

[27] U. Mahmud, "Organizing contextual data in context aware systems: A review," in *Handbook of Research on Human-Computer Interfaces, Developments, and Applications*, P. A. Hershey, Ed., pp. 273–303, IGI Global, 2016.

[28] G. W. Musumba and H. O. Nyongesa, "Context awareness in mobile computing: A review," *International Journal of Machine Learning and Applications*, vol. 2, no. 1, 2013.

[29] J.-Y. Hong, E.-H. Suh, and S.-J. Kim, "Context-aware systems: a literature review and classification," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8509–8522, 2009.

[30] S. L. Kiani, A. Anjum, M. Knappmeyer, N. Bessis, and N. Antonopoulos, "Federated broker system for pervasive context provisioning," *The Journal of Systems and Software*, vol. 86, no. 4, pp. 1107–1123, 2013.

[31] U. Mahmud, U. Farooq, M. Y. Javed, and N. A. Malik, "Representing and organizing contextual data in context aware environments," *Journal of Computing*, vol. 4, pp. 61–67, March 2012.

[32] J. Grudin, "Desituating action: Digital representation of context," *Human-Computer Interaction*, vol. 16, no. 2-4, pp. 269–286, 2001.

[33] L. Feng, P. M. G. Apers, and W. Jonker, "Towards context-aware data management for ambient intelligence," *Lecture Notes in Computer Science*, vol. 3180, pp. 422–431, 2004.

[34] U. Mahmud, N. Iltaf, and F. Kamran, "Context congregator: gathering contextual information," in *Proceedings of the 5th Frontiers of Information Technology (FIT '07)*, pp. 134–141, Islamabad, Pakistan, 2007.

[35] N. A. Malik, M. Y. Javed, and U. Mahmud, "Estimating user preferences by managing contextual history in context aware systems," *Journal of Software* , vol. 4, no. 6, pp. 571–576, 2009.

[36] U. Mahmud and M. Y. Javed, "Context inference engine (CiE): inferring context," *International Journal of Advanced Pervasive and Ubiquitous Computing*, vol. 4, no. 3, pp. 13–41, 2012.

[37] U. Mahmud and M. Y. Javed, "Context Inference Engine (CiE): Classifying activity of context using minkowski distance and standard deviation-based ranks," in *Systems and Software Development, Modeling, and Analysis: New Perspectives and Methodologies*, pp. 65–112, IGI Global, 2014.

[38] S. Hussain, Z. Wang, and I. K. Toure, "An approach for QoS measurement and web service selection sureness," *High Technology Letters*, vol. 19, no. 3, pp. 283–289, 2013.

[39] S. Hussain, Z. S. Wang, and I. K. Toure, "Performance analysis of web services in different types of internet technologies," *Applied Mechanics and Materials*, vol. 513-517, pp. 1431–1436, 2014.

[40] O. Landsiedel, K. Wehrle, and S. Götz, "Accurate prediction of power consumption in sensor networks," in *Proceedings of the 2nd IEEE Workshop on Embedded Networked Sensors (EmNetS-II '05)*, pp. 37–44, May 2005.

[41] J. Flinn, *Extending Mobile Computer Battery Life Through Energy-Aware Adaptation [Ph.D. Thesis]*, Carnegie Melon University, Pittsburgh, PA, USA, 2001.

[42] A. Kansal, F. Zhao, J. Liu, N. Kothari, and A. A. Bhattacharya, "Virtual machine power metering and provisioning," in *Proceedings of the 1st ACM Symposium on Cloud Computing (SoCC '10)*, pp. 39–50, Indianapolis, IN, USA, June 2010.

[43] A. Pathak, Y. C. Hu, and M. Zhang, "Where is the energy spent inside my app?: Fine grained energy accounting on smartphones with eprof," in *Proceedings of the 7th ACM European Conference on ComputerSystems (EuroSys '12)*, pp. 29–42, Bern, Switzerland, April 2012.

[44] J. Kulk and J. Welsh, "A low power walk for the NAO robot," in *Proceedings of the Australasian Conference on Robotics and Automation (ACRA '08)*, pp. 1–7, December 2008.

[45] C. Seo, G. Edwards, S. Malek, and N. Medvidovic, "A framework for estimating the impact of a distributed software system's architectural style on its energy consumption," in *Proceedings of the 7th IEEE/IFIP Working Conference on Software Architecture (WICSA '08)*, pp. 277–280, February 2008.

[46] N. Vallina-Rodriguez, J. Shah, A. Finamore et al., "Breaking for commercials: Characterizing mobile advertising," in *Proceedings of the ACM Internet Measurement Conference (IMC '12)*, pp. 343–356, November 2012.

[47] Y. Li, H. Chen, and W. Shi, "Power behavior analysis of mobile applications using Bugu," *Sustainable Computing: Informatics and Systems*, vol. 4, no. 3, pp. 183–195, 2014.

[48] A. B. Abkenar, S. W. Loke, W. Rahayu, and A. Zaslavsky, "Energy considerations for continuous group activity recognition using mobile devices: The case of GroupSense," in *Proceedings of the 30th IEEE International Conference on Advanced Information Networking and Applications (AINA '16)*, pp. 479–486, Crans-Montana, Switzerland, March 2016.

[49] M. Marcu, "Powerthermal profiling of software applications," *Microelectronics Journal*, vol. 42, no. 4, pp. 601–608, 2011.

[50] L.-T. Duan, B. Guo, Y. Shen, Y. Wang, and W.-L. Zhang, "Energy analysis and prediction for applications on smartphones," *Journal of Systems Architecture*, vol. 59, no. 10, pp. 1375–1382, 2013.

[51] Y.-F. Chung, C.-Y. Lin, and C.-T. King, "ANEPROF: Energy profiling for android java virtual machine and applications," in *Proceedings of the 17th IEEE International Conference on Parallel and Distributed Systems (ICPADS '11)*, pp. 372–379, Tainan, Taiwan, December 2011.

[52] M. Tanabian, "Mobile device/application power profiling—testing and design considerations in mobile devices & apps for power performance and battery life," *Intuigence Group, White Paper*, 2011.

[53] Y.-D. Lin, E. Rattagan, Y.-C. Lai et al., "Calibrating parameters and formulas for process-level energy consumption profiling in smartphones," *Journal of Network and Computer Applications*, vol. 44, pp. 106–119, 2014.

[54] M. Lee, D.-K. Kim, and J.-W. Lee, "Analysis of characteristics of power consumption for context-aware mobile applications," *Information*, vol. 5, no. 4, pp. 612–621, 2014.

[55] T. Mitchell, *Machine Learning*, McGraw Hill, 1997.

[56] S. Elmalaki, M. Gottscho, P. Gupta, and M. Srivastava, "A case for battery charging-aware power management and deferrable task scheduling in smartphones," in *Proceedings of the 6th USENIX Conference on Power-Aware Computing and Systems (HotPower '14)*, pp. 1–4, Broomfield, CO, USA, 2014.

[57] G. Anastasi, S. Brienza, G. L. Re, and M. Ortolani, "Energy-efficient protocol design," in *Green Communications: Principles, Concepts and Practice*, K. Samdanis, P. Rost, A. Maeder, M. Meo, and C. Verikoukis, Eds., pp. 339–360, John Wiley & Sons, Ltd, Chichester, UK, 2015.

[58] J. Flinn and M. Satyanarayanan, "Energy-aware adaptation for mobile applications," in *Proceedings of the Seventeenth ACM Symposium on Operating Systems Principles*, pp. 48–63, Charleston, SC, USA, December 1999.

[59] S. Elmalaki, L. Wanner, and M. Srivastava, "CAreDroid: Adaptation framework for android context-aware applications," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*, pp. 386–399, Paris, France, September 2015.

[60] M. Moghimi, J. Venkatesh, P. Zappi, and T. Rosing, "Context-aware mobile power management using fuzzy inference as a service," in *Proceedings of the International Conference on Mobile Computing, Applications, and Services (MobiCASE '12)*, vol. 110, pp. 314–327, Seattle, WA, USA, 2012.

[61] Y. Fei, Z. Lin, and N. K. Jha, "An energy-aware framework for coordinated dynamic software management in mobile computers," in *Proceedings of the IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS '04)*, pp. 306–317, Volendam, Netherlands, October 2004.

[62] J. Chen and G. Venkataramani, "EnDebug: A hardware-software framework for automated energy debugging," *Journal of Parallel and Distributed Computing*, vol. 96, pp. 121–133, 2016.

[63] P. Simoens, C. Mahieu, F. Ongenae et al., "Internet of robotic things: context-aware and personalized interventions of assistive social robots," in *Proceedings of the 5th IEEE International*

Conference on Cloud Networking (CloudNet '16), pp. 204–207, Pisa, Italy, October 2016.

[64] Q. Zhang, G. Metri, S. Raghavan, and W. Shi, "RESCUE: An energy-aware scheduler for cloud environments," *Sustainable Computing*, vol. 4, no. 4, pp. 215–224, 2014.

[65] N. Akhter and M. Othman, "Energy aware resource allocation of cloud data center: review and open issues," *Cluster Computing*, vol. 19, no. 3, pp. 1163–1182, 2016.

[66] S. Kang, J. Lee, H. Jang, Y. Lee, S. Park, and J. Song, "A scalable and energy-efficient context monitoring framework for mobile personal sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 686–702, 2010.

[67] D. Sathan, A. Meetoo, and R. K. Subramaniam, "Context aware lightweight energy efficient framework," *World Academy of Science Engineering and Technology*, vol. 52, pp. 64–70, 2009.

[68] R. Hermann, P. Zappi, and T. S. Rosing, "Context aware power management of mobile systems for sensing applications," in *Information Processing in Sensor Networks*, pp. 1–5, 2012.

[69] T. Simunic, L. Benini, and G. De Micheli, "Energy-efficient design of battery-powered embedded systems," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED '99)*, pp. 212–217, San Diego, Calif, USA, August 1999.

[70] T. Ŝimunić, L. Benini, and G. De Micheli, "Energy-efficient design of battery-powered embedded systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 1, pp. 15–28, 2001.

[71] V. Tiwari, S. Malik, A. Wolfe, and M. T.-C. Lee, "Instruction level power analysis and optimization of software," in *Technologies for Wireless Computing*, pp. 139–154, Springer, 1996.

[72] V. Tiwari, S. Malik, and A. Wolfe, "Power analysis of embedded software: a first step towards software power minimization," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 437–445, San Jose, Calif, USA, 1994.

[73] H. Mehta, R. M. Owens, M. J. Irwin, R. Chen, and D. Ghosh, "Techniques for low energy software," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED '97)*, pp. 72–75, Monterey, Calif, USA, August 1997.

[74] E. Capra, C. Francalanci, and S. A. Slaughter, "Is software 'green'? Application development environments and energy efficiency in open source applications," *Information and Software Technology*, vol. 54, no. 1, pp. 60–71, 2012.

[75] S. Hao, D. Li, W. G. J. Halfond, and R. Govindan, "Estimating mobile application energy consumption using program analysis," in *Proceedings of the 35th International Conference on Software Engineering (ICSE '13)*, pp. 92–101, San Francisco, Calif, USA, May 2013.

[76] S.-L. Tsao, C.-K. Yu, and Y.-H. Chang, "Profiling energy consumption of I/O functions in embedded applications," in *Proceedings of the International Conference on Architecture of Computing Systems (ARCS '13)*, vol. 7767, pp. 195–206, Prague, Czech, 2013.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 6421717, 19 pages
https://doi.org/10.1155/2018/6421717

Research Article

Congestion Control and Prediction Schemes Using Fuzzy Logic System with Adaptive Membership Function in Wireless Sensor Networks

Phet Aimtongkham,[1] **Tri Gia Nguyen**,[2] **and Chakchai So-In**[1]

[1] *Applied Network Technology (ANT) Laboratory, Department of Computer Science, Faculty of Science, Khon Kaen University, Khon Kaen, Thailand*
[2] *Department of Information Technology, Duy Tan University, Vietnam*

Correspondence should be addressed to Chakchai So-In; so-in@ieee.org

Received 27 March 2018; Revised 21 June 2018; Accepted 4 July 2018; Published 1 August 2018

Academic Editor: Al-Sakib K. Pathan

Network congestion is a key challenge in resource-constrained networks, particularly those with limited bandwidth to accommodate high-volume data transmission, which causes unfavorable quality of service, including effects such as packet loss and low throughput. This challenge is crucial in wireless sensor networks (WSNs) with restrictions and constraints, including limited computing power, memory, and transmission due to self-contained batteries, which limit sensor node lifetime. Determining a path to avoid congested routes can prolong the network. Thus, we present a path determination architecture for WSNs that takes congestion into account. The architecture is divided into 3 stages, excluding the final criteria for path determination: (1) initial path construction in a top-down hierarchical structure, (2) path derivation with energy-aware assisted routing, and (3) congestion prediction using exponential smoothing. With several factors, such as hop count, remaining energy, buffer occupancy, and forwarding rate, we apply fuzzy logic systems to determine proper weights among those factors in addition to optimizing the weight over the membership functions using a bat algorithm. The simulation results indicate the superior performance of the proposed method in terms of high throughput, low packet loss, balancing the overall energy consumption, and prolonging the network lifetime compared to state-of-the-art protocols.

1. Introduction

Recently, according to Gardner [1], over 8 billion devices have connected and interacted to form the Internet of Things (IoT) network, most of which perform sensing and monitoring tasks [2] and report their data over a cloud-based platform for further analysis using big data technology [3]. One key technology involved in this trend is the use of wireless sensor networks (WSNs) [4].

In WSNs, many sensing devices or sensor nodes (SNs) are connected, typically via wireless transmissions (e.g., radio frequency (RF)) to overcome the limitation of wire-based infrastructure. These SNs can form an ad hoc network (multihop transmissions via relays or intermediate SNs) that is self-computing, self-contained, and self-organized and that can interact with nodes that are outside the network toward the gateway (sink node or base station (BS)) [5].

Although SNs involve multifunctional devices, including sensing, computing, storage, and transmission units, the key limitation is in their energy consumption. The self-power supply of the battery determines the SN lifetime. In addition to the several advantages of WSNs, there are multiple challenges, such as routing, localization, quality of service (QoS), and security [6, 7]. In particular, the traffic over the network is massive in large-scale WSNs, causing problems with data transmission in determining the optimal path to the sink node as well as heterogeneous traffic in various topologies, especially nonuniform traffic once randomly deployed to places such as forests and disaster areas [8, 9], thereby shortens the overall network lifetime, and results in congestion control issues in WSNs [10, 11] for applications such as thermal rating monitoring of conductors in power systems, life sensor monitoring in health-care applications, and earthquake warning systems [12, 13].

Traditionally, congestion is one of the main issues affecting network outperformance, especially with high density nodes from wired [14] to wireless [15] and particularly WSN. Different from others, WSN has specific characteristics, such as very low energy via battery operation and low bandwidth, which are of great concern for congestion [16]. In general, two main approaches are used to mitigate congestion in WSNs: traffic control (TC) and resource control (RC) [17, 18]. The former is used to adjust the transmission rate, e.g., additive increase multiplicative decrease (AIMD). This technique regularly measures the current bandwidth and adjusts the rate based on the presence of transmitted SNs with consideration of fairness. The key advantage of TC is the sudden effect of congestion alleviation with the decrease of sending rates; however, the main limitations are probable low throughput and packet loss during the initial stage of congestion.

Generally, there are three types of congestion [19]: at the source node (caused by the simultaneous transmission of data by SNs that are close to one another), at an intermediate node (caused by data aggregation and the parent node section strategy, in particular, with buffer overflow), and at the sink node (caused by too many children sending traffic to the only sink node at the same time). These types of congestion can cause packet loss, delay, and low throughput.

The latter approach, RC, mainly refers to an approach that is not limited to decreasing the sending rate; however, in wireless communications, more than one path is usually directed to the destination (BS). Alternate path selection (in which a detour to the destination is constructed with available relays or SNs) is one type of RC. In this method, the key advantage is to balance the resources in the network (i.e., the available SNs), which results in higher throughput. This approach is our focus in the present research.

With an RC-based approach, several factors can affect the network performance, particularly the congestion mitigation, such as buffer occupancy, transmission rate, distance between the sender and the receiver, and most importantly, energy. There is no linear relationship among these factors. Therefore, despite the advantages of RC-based approaches, packet loss and low throughput are still probable due to unsuitable path selection, given those factors.

Thus, we focus on enhancing the RC by considering a probable congestion prediction; in particular, we seek the best alternate path before congestion using a fuzzy logic system (FLS) for adaptive path determination in consideration of buffer occupancy, forwarding rate, hop count, and remaining energy, called the Optimized Fuzzy Logic-based Congestion Control Scheme with Exponential Smoothing Prediction in WSNs (OFES). Our contributions are as follows:

(i) Propose an architecture for congestion mitigation in WSNs.

(ii) Adopt exponential smoothing to predict the buffer occupancy.

(iii) Apply FLS to determine appropriate weights for path determination in consideration of hop count, remaining energy, buffer occupancy, and forwarding rate.

(iv) Optimize FLS with a bat algorithm (BAT) to tune the membership function.

The remainder of the article is organized as follows: In Section 2, we review related work on congestion controls and focus on resource-control-based approaches and fuzzy integration. Section 3 provides the assumptions and definitions of our wireless sensor network models. In Section 4, our proposed approach, which is based on a fuzzy-based congestion control and prediction scheme, is presented. Section 5 presents and discusses the performance evaluation through simulation results. Finally, the conclusions and the future research direction are discussed in Section 6.

2. Related Work

In this section, we focus on RC-based congestion controls in WSNs. We also discuss work on FLS-based congestion controls and predictions.

The pioneering approach to RC is Topology-Aware Resource Adaptation for Congestion Avoidance (TARA), which was proposed by J. Kang et al. [20]. The authors proposed a topological control that applies the concept of distributed and merged pathways. Once congestion is detected, a particular node will split the path (into multipaths) to detour the direct transmission to the congested SN. Then, the multipaths are combined at the merged SN. However, the alternate path can also encounter congestion. More paths can lead to high energy consumption by the network.

Different approaches were considered by only focusing on the best-possible path, such as the Hierarchical Tree Alternative Path (HTAP), proposed by C. Sergiou et al. [21]. Starting from the sink node, a hop-by-hop-based approach is applied, which uses the buffer occupancy information as a threshold. This approach forms a hierarchical tree structure from the source node. However, it may cause a delay since the SN must determine the possible path in each hop-by-hop transmission according to the congestion.

To overcome the limitations of HTAP, the same set of authors proposed a modified approach, the Dynamic Alternative Path Selection Scheme (DAiPAS) [22], by initially constructing the topology to avoid congestion in the hop-by-hop-based scheme. DAiPAS also works with soft and hard states. A soft state tends to have only 1 communication link between each SN and its parent, which is formed by announcing the presence of a transmission. Therefore, the other SNs to be transmitted (if any) should seek other paths before congestion occurs. In contrast, in a hard state, the buffer occupancy will be used to determine the acceptance of an SN by its parent transmission if there is no option.

W. Chen et al. [23] developed the Congestion Control and Energy-balanced scheme based on the Hierarchy (CCEbH), which has an awareness of energy balance. CCEbH adopts the initial topology construction from DAiPAS. However, there is a threshold (quota) for each SN transmission to its parent, i.e., a time interval for each transmission (based on the buffer occupancy and the forwarding rate). There is also an energy balance method in two dimensions, i.e., vertical and horizontal. The SN will typically ask its vertical and horizontal neighbors to determine the paths to the sink node. Here, time synchronization may become an issue in high traffic and dense deployment.

In addition to its use in RC-based methods, FLS is typically used for TC-based approaches. For example, S. Jaiswal and A. Yadav proposed Fuzzy-Based Adaptive Congestion Control (FBACC) [24]. Some of the key parameters of FBACC are related to probable congestions. For example, buffer occupancy, forwarding rate, and the number of connections of a (to be) congested SN are used as the inputs for FLS to determine the appropriate sending rate. However, here, there is no consideration of the energy of SNs as a key factor for prolonging the network lifetime of the WSN.

Y. Zhang et al. [25] considered the energy of SNs, which includes the energies of a particular SN and its neighbors and the path information, in the level-based path determination as the inputs of FLS for selecting the cluster head (CH); this method is named the Energy-Efficient Distributed Clustering algorithm based on fuzzy approach with nonuniform distribution (EEDCF). The CH transmits to the next CH toward the sink node. The main limitation is that, in the case of sparse deployment, EEDCF tends to underperform.

Based on the pioneering work on the TC-based scheme Congestion Detection and Avoidance (CODA) [26], while applying backward congestion notification to the source node to adjust the sending rate to mitigate the effects of congestion, research has also been conducted on congestion prediction to avoid congestion. For example, M. Zawodniok et al. [27] proposed Dynamic Predictive Congestion Control (DPCC). DPCC utilizes queuing, including the embedded channel estimator algorithm, as the metric for adjusting the sending rate; however, this approach incurs high overhead. A. U. Rajan et al. proposed the Rate Regulation Split Protocol (RRG) [28] with the concept of congestion prediction, which uses a probabilistic method on data traffic and buffer occupancy to determine a congestion probability.

3. Models and Assumptions

In this section, we describe networking and energy models, including all required messages, as follows.

3.1. Network Models. In this research, similar to other recent work on congestion control in WSN [20–28], there is only one base station or sink node, which is typically placed at the end of the network (to be connected to the Internet). There is a power source at this node. At a specific time, sensor nodes that are activated by certain events (such as sensing and becoming ready for data transmission), also known as source nodes, will send data toward the sink node via the next or neighboring nodes (hop-based), nonuniform deployments. We make the following network assumptions:

(i) Sensor nodes (SNs): in addition to the sink node, SNs are randomly deployed in a uniform manner within a network square area, each of which has the same initial energy.

(ii) After the SNs are deployed, there is no mobility involved and no explicit location information.

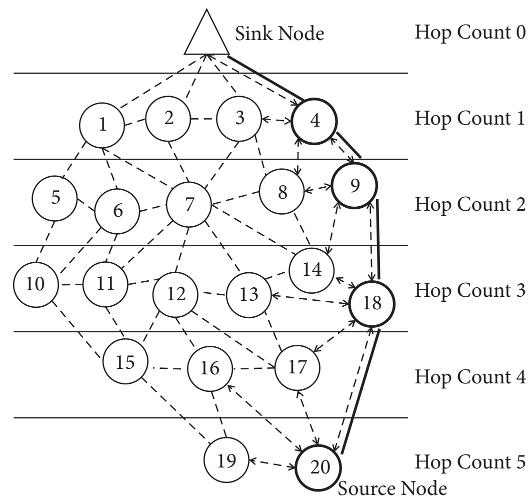


Figure 1: Example: network model.

(iii) Each SN has the same communication range.

(iv) All SNs within the network have been given the time-synchronization protocol for communication purposes [20–28].

(v) Any SN can leave the network if and only if the energy is depleted (it is no longer in operation).

Figure 1 shows an example of our network model, which consists of a sink, a source, and other SNs. The dashed line defines the communication range and its connectivity from each SN to its neighboring node (NN). The arrow head is determined by the broadcast message. The network will be generally divided into layers based on the number of hop counts. Once the path has been defined (as the solid line), the source node will send the data to the sink node accordingly (from the 20th SN, go over 18, 9, and 4, and finally go to the sink node).

3.2. Energy Models. To communicate within WSNs, we follow a standard IEEE 802.15.4 Low-rate Wireless Personal Area Network (LR-WPAN) protocol, in which the energy depends on the number of packets that are sent and received by a specific SN. Equation (1) shows the energy model for WSNs [29], where the transmission energy E_{Tx} (Joules/bit) of the ith bit over a distance d in the free space propagation model is expressed as follows:

$$E_{Tx}(i,d) = i \cdot E_{elec} + i \cdot E_{tran} \cdot d^2 \quad (1)$$

where E_{elec} denotes the data processing energy and E_{tran} is the energy for the transmission of 1 bit over a distance d.

Similar to the transmission, (2) expresses the receiving energy E_{Rx} (Joules/bit) of i bits, where E_{Reci} is the energy for receiving 1 bit.

$$E_{Rx}(i) = i \cdot E_{Reci} \quad (2)$$

Note that, based on the specification of MicaZ [30] with 3 volts (2 AA batteries), $E_{elec} = 0.96\ \mu$J/bit, $E_{tran} = 0.144\ \mu$J/bit, and $E_{reci} = 0.96\ \mu$J/bit.

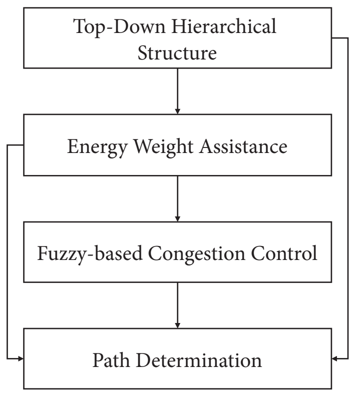


Figure 2: Overview of OFES.

3.3. System Messages. There are eight types of messages in both broadcast and unicast for our OFES architecture:

(i) *Route-Request-Msg(Node_ID, Hop Count)*. This broadcast message is used to make a route query to neighbors, and it includes the node identifier and the hop count from the sink node.

(ii) *Route-Reply-Msg(Node_ID, Hop Count)*. Once *Route-Request-Msg* has been received, the neighbors reply with this broadcast message, which includes their identities and hop counts.

(iii) *Route-End-Msg(Node_ID, Hop Count)*. When no *Route-Reply-Msg* is sent back within a specific time-out, a specific node will send this broadcast message to indicate the end of the hop tree (to determine a maximum hop count).

(iv) *Route-Update-Msg(Node_ID)*. If a node does not have enough energy to perform an operation (within a threshold), that node will send this broadcast message to its neighbors, which includes its identity.

(v) *Request-Stage-Msg(Node_ID)*. This message is broadcasted to an SN's neighbors for weight inquiry.

(vi) *Reply-Stage-Msg(Node_ID, EW, AW)*. After *Request-Stage-Msg* has been received, each SN will send this unicast message back, which includes the identity and two additional (locally computed) weights: the Energy-Assisted Routing Weight (EW) and the Availability Weight (AW).

(vii) *Data-Sent-Msg(Node_ID, Data)*. This unicast message is used for data transmission after the final path has been determined.

(viii) *Data-Ack-Msg(Node_ID)*. This unicast message is used for data transmission confirmation.

4. Optimized Fuzzy Logic-Based Congestion Control Scheme with Exponential Smoothing Prediction in WSNs (OFES)

Figure 2 shows an overview of the OFES architecture. There are 4 main components:

Table 1: Example: neighbor table in Route Create.

Node ID	Hop Count

(i) **Top-Down Hierarchical Structure.** This component is first used to generate the layer-based structure with hop count as the key metric. We adopt the Ad hoc On-Demand Distance Vector (AODV) [31], which is used for noninfrastructure wireless networks, to generate a many-to-one structure from SNs to the sink node. Two main steps are applied as (1) Route Create and (2) Route Update phases.

(ii) **Energy-Assisted Routing Weight.** The second component is used for weight generation with FLS. Two fuzzy inputs are considered here: hop count and remaining energy.

(iii) **Congestion Control and Prediction.** This component is mainly used for congestion control to aid in congestion prediction. Here, we adopt exponential smoothing, with its optimization over FLS tuned by BAT, and the buffer occupancy and the forwarding rate as fuzzy inputs.

(iv) **Path Determination.** After weight derivation, this final component is used to determine a final path toward the sink node. Generally, with the performance versus energy consumption trade-off, OFES can select one of the three components to achieve the highest performance: applying only the first component, the first and the second components, or all three components.

4.1. Top-Down Hierarchical Structure. In this component, OFES starts to construct the topology for path determination by further applying the layering concept with hop count determination (one for each additional layer) from the sink node to the possible source nodes (all SNs) as the key metric. Two steps are carried out: the Route Create and Route Update phases.

4.1.1. Route Create Phase. We adopt AODV as a baseline for determining a layer-based approach, using hop count as a main metric. Similar to the tree-based concept, here, the sink node is considered a root of the tree and an additional layer after the root is traversed hop by hop (in a top-down manner) until reaching the source node.

First, the sink node (hop count = 0) broadcasts *Route-Request-Message* to the nearby nodes (neighbors). Once each neighboring node receives this message, it stores the information (Node ID and Hop Count) in the neighbor table (see Table 1) and increases the hop count by 1 to implicitly specify how far it is from the sink node.

After receiving and storing the information, this neighboring node sends the reply message *Route-Reply-Msg* to confirm its receipt. This node continues this process (*Route-Request-Message* and *Route-Reply-Msg*) until there are no additional neighboring nodes (no responses to *Route-Reply-Msg*). In this case, this node (the end node of a particular

Sink node broadcasts *Route-Request-Msg* to SN_i
while (*SN* is an intermediate node) do
SN_i receives *Route-Request-Msg* and creates [$NeighborTable_i$]
SN_i sends *Route-Reply-Msg* and broadcasts *Route-Request-Msg*
  if SN_i is last hierarchy node **then**
  SN_i *sendsRoute-End-Msg*
  Break
  end if
end while

Algorithm 1: Topology construction in Route Create.

Table 2: Example: Neighbor table of SN 4 in Route Create.

Node ID	Hop Count
(Sink Node)	0
3	1
8	2
9	2

Table 3: Example: neighbor table of SN 20 in Route Create.

Node ID	Hop Count
18	3
16	4
17	4
19	5

path along the tree, Timeout 3,000 ms) sends *Route-End-Msg* back to the previous node (lower hop count). This process continues until the sink node is reached. Note that this message is used for maximum hop count determination of each path.

Note that, in case a particular node may receive more than one *Route-Request-Message*, the node will select (or update) the table based on the number of hop counts (lowest is preferred).

Table 2 shows a neighbor table of the 4th SN (see also Figure 1); here, there is only 1 neighbor (SN = 3rd) for the first broadcast and there are 2 neighbors (SNs = 8th and 9th) in the next broadcast. Note that, here, we assume the architecture is hop-by-hop communication. When the broadcasting pattern has been completed, the end node of the tree is node 20. The neighbor table is shown as Table 3.

Figure 3 also shows an example of this case; here, there are six possible paths from the 20th SN to the sink node. Algorithm 1 also shows the steps of the Route Create phase. The final path determination will be discussed again in Section 4.4.

4.1.2. Route Update Phase. To avoid a disconnectivity, the Route Update phase is used to switch the route. Generally, the disconnectivity (disjoint path) is due to a lack of energy (when the battery has run out). Here, we specify an energy threshold for each SN. If the energy falls below the threshold

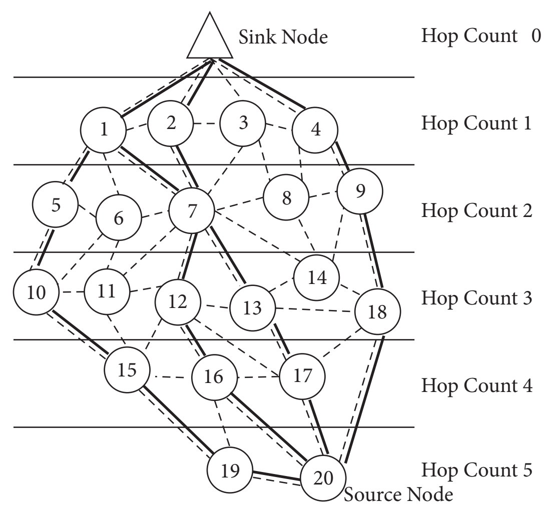


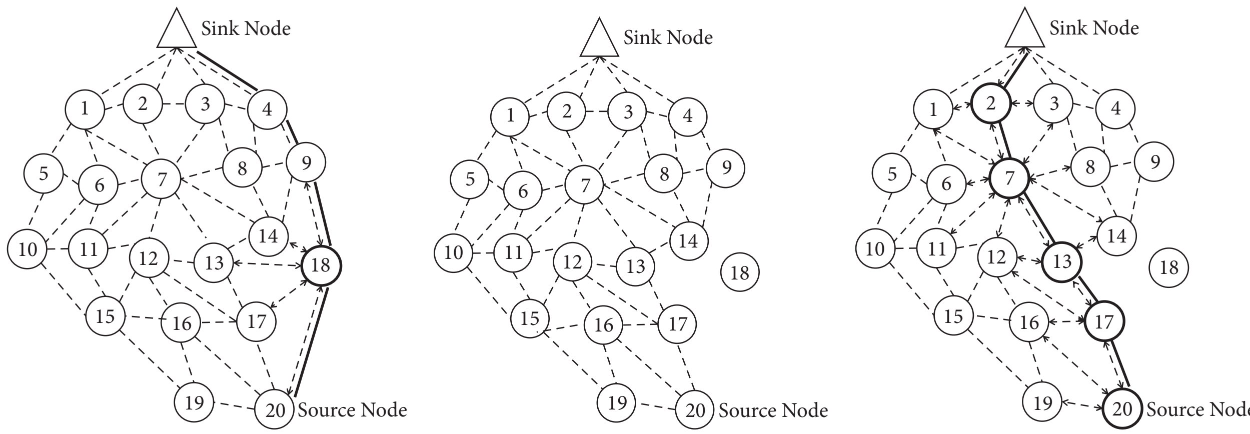
Figure 3: Example: possible path selection in Route Create.

(here, we set it to 5% of the initial energy [32]), the SN will send *Route-Update-Msg* to its neighbors to update the neighbor tables.

Figure 4 shows an example of this phase. Figure 4(a) shows the final determined path (the shortest path from 20 ⟶ 18 ⟶ 9 ⟶ 4 ⟶ sink node) from the possible paths (see Figure 3), which include (20 ⟶ 19 ⟶ 15 ⟶ 10 ⟶ 5 ⟶ 1 ⟶ sink node), (20 ⟶ 16 ⟶ 12 ⟶ 7 ⟶ 1 ⟶ sink node), (20 ⟶ 16 ⟶ 12 ⟶ 7 ⟶ 2 ⟶ sink node), (20 ⟶ 17 ⟶ 13 ⟶ 7 ⟶ 1 ⟶ sink node), (20 ⟶ 17 ⟶ 13 ⟶ 7 ⟶ 2 ⟶ sink node), and (20 ⟶ 18 ⟶ 9 ⟶ 4 ⟶ sink node). However, when the energy of the 18th SN is low, it will broadcast *Route-Update-Msg*, and its neighbors will update the table (shown in Figure 4(b)). Figure 4(c) shows the selected path: 20 ⟶ 17 ⟶ 13 ⟶ 7 ⟶ 2 ⟶ sink node (the next-shortest path). Table 4 shows the neighbor table in the absence of the 18th SN (see also Table 3 for the neighbor table in the presence of the 18th SN).

Figure 5 shows a detailed sequence diagram of both the Route Create and Route Update phases.

4.2. Energy-Assisted Routing Weight (EW). This component is used to determine routing criteria (as a weight or Energy-Assisted Routing Weight (EW)) between hop count and



(a) SN 18 sends *Route-Update-Msg* once the energy falls below the threshold

(b) Neighbors of SN 18 remove the connectivity from the table

(c) Final path determination from 20 ⟶ 17 ⟶ 13 ⟶ 7 ⟶ 2 ⟶ sink node

Figure 4: Example: Route Update at SN 18.

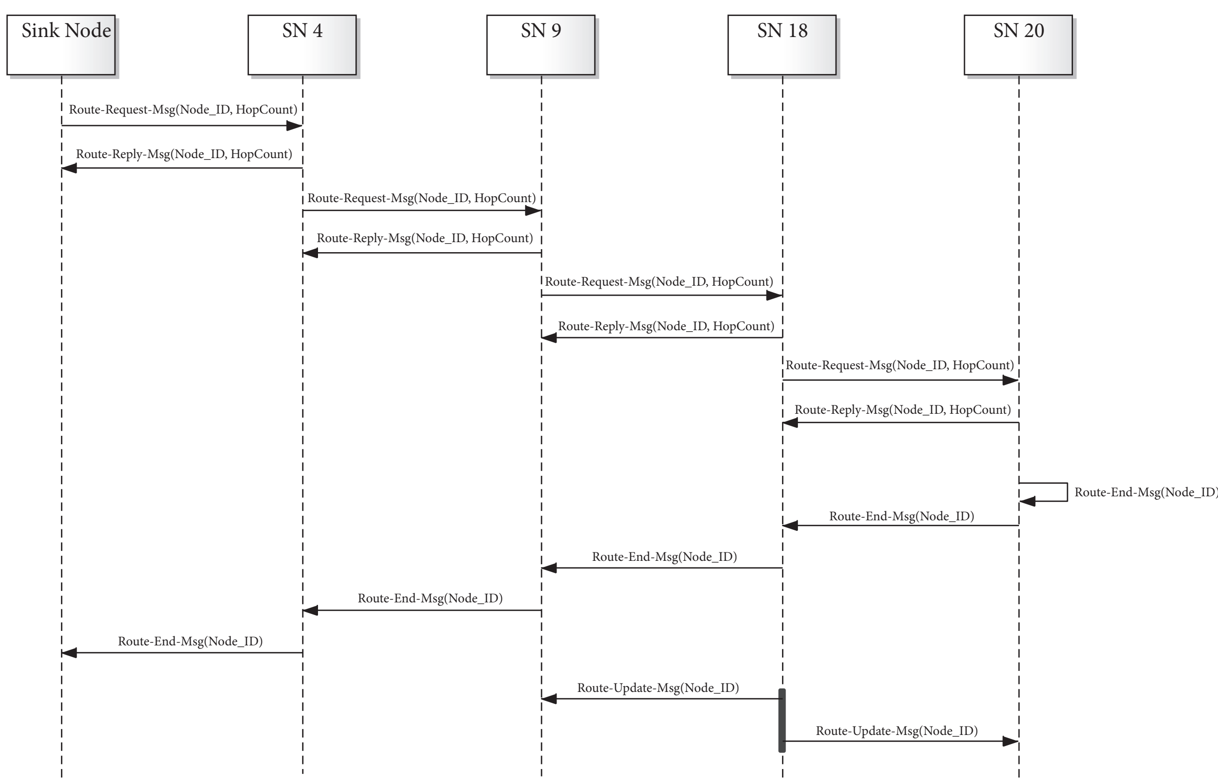


Figure 5: Sequence diagram of OFES in Route Create and Update phases.

Table 4: Example: Neighbor Table in SN 20 in the absence of the 18th SN.

Node ID	**Hop Count**
16	4
17	4
19	5

remaining energy. We applied FLS to determine a final weight based on the membership function.

(A) Fuzzy Logic System (FLS). FLS is a technique for logical reasoning with an approximation-based approach as opposed to an exact solution. FLS identifies solutions within the range between 0 and 1 (not exactly 0 or 1) due to the fuzziness. Based on our evaluation process, FL with Mamdani outperformed

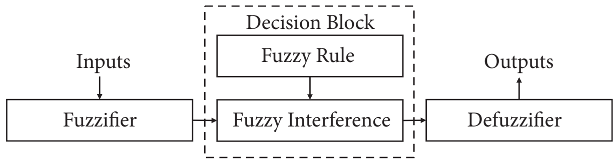


Figure 6: Block diagram: adaptive fuzzy logic system.

Table 5: Examples of 9 EW fuzzy rules.

IF (Hop Count)	AND (Remaining Energy)	THEN (Weight)
L	L	M
L	M	M
L	H	H
M	L	L
M	M	M
M	H	H
H	L	L
H	M	L
H	H	M

FL Sugeno; thus, we select FL with Mamdani, which has four main steps [33] (see Figure 6).

Fuzzifier is used to transform the input (i.e., hop count and remaining energy) to a numerical format, which is typically normalized to a range between 0 and 1 from the current over the maximum hops and similarly the remaining over the initial energy. Then, the input is mapped to determine the crossing points of each function (corresponding to fuzzy rules). Based on our intensive evaluation, we applied triangular (low, medium, and high for output weight) and trapezoidal (low, medium, and high for the two inputs) functions, which outperformed the others, such as Gaussian, generalized bell, and sigmoid. Figure 7 shows an example of a membership function of hop count and remaining energy, including the output weights.

Fuzzy rule is used to generate the input-to-output mapping based on the membership function (if/then rule). Table 5 shows a rule construction procedure with 9 rules, including a combination of high (H), medium (M), and low (L).

Fuzzy inference engine is used for output (weight) derivation given the inputs with a fuzzy rule-based system using an aggregation method, i.e., the intersection operation. Figure 8 shows an example of EW fuzzy with 9 rules (See Table 5). There are two normalized inputs, i.e., hop count and remaining energy, here with 0.38 and 0.66 as examples. With these inputs, each fuzzy set is considered with "AND" operation; i.e., the minimum among the twos will be selected to cross over the weight, resulting in output. Next, the aggregation method is applied over the 9 rules before applying the defuzzifier in the next component.

Defuzzifier is used to compute the final output based on the center of gravity (CoG) concept over the derived output (weight). This final output (in the range of 0 to 1) provides the weight for the routing stage; here, 0.577 is an example after the fuzzy inference engine. Note that the output is the energy-assisted routing weight (EW).

Table 6: Example: neighbor table of SN 20 with EW.

Node ID	Hop Count	EW (Weight)
16	4	0.89
17	4	0.75
19	5	0.70
20	5	0.83

(B) EW Derivation with FLS. A weight (EW) will be derived for the FLS (i.e., the final output from Defuzzifier). Then, that weight will be computed once *Request-Stage-Msg* has been received. Subsequently, SN will reply with this weight, in addition to its own identity, via *Reply-Stage-Msg*. With the reply message, the corresponding weight will be stored in the neighbor table. Table 6 shows an example of the update of the 20th SN. Here, this SN broadcasts the request message to its neighbors, SNs 16, 17, 19, and 20. These SNs send the reply to update the base, for example, with weights of 0.89, 0.75, 0.70, and 0.83, respectively. Note that the selection of the next hop will be discussed again in Section 4.4.

4.3. Congestion Control and Prediction. Previously, two main factors, namely, hop count and remaining energy, were used as the main criteria for determining a path. However, in dense deployments, another key factor is network congestion, which typically lowers the overall performance, e.g., through packet loss, low throughput, and high latency. Thus, we also investigated this constraint using a statistical congestion prediction model (in consideration of buffer occupancy) with FLS-based congestion controls. Similar to Section 4.2, however, here, we additionally tuned the membership function of FLS with BAT for the two inputs, namely, predicted buffer occupancy and forwarding rate, to derive another weight factor (Availability Weight (AW)).

4.3.1. Congestion Prediction with Exponential Smoothing

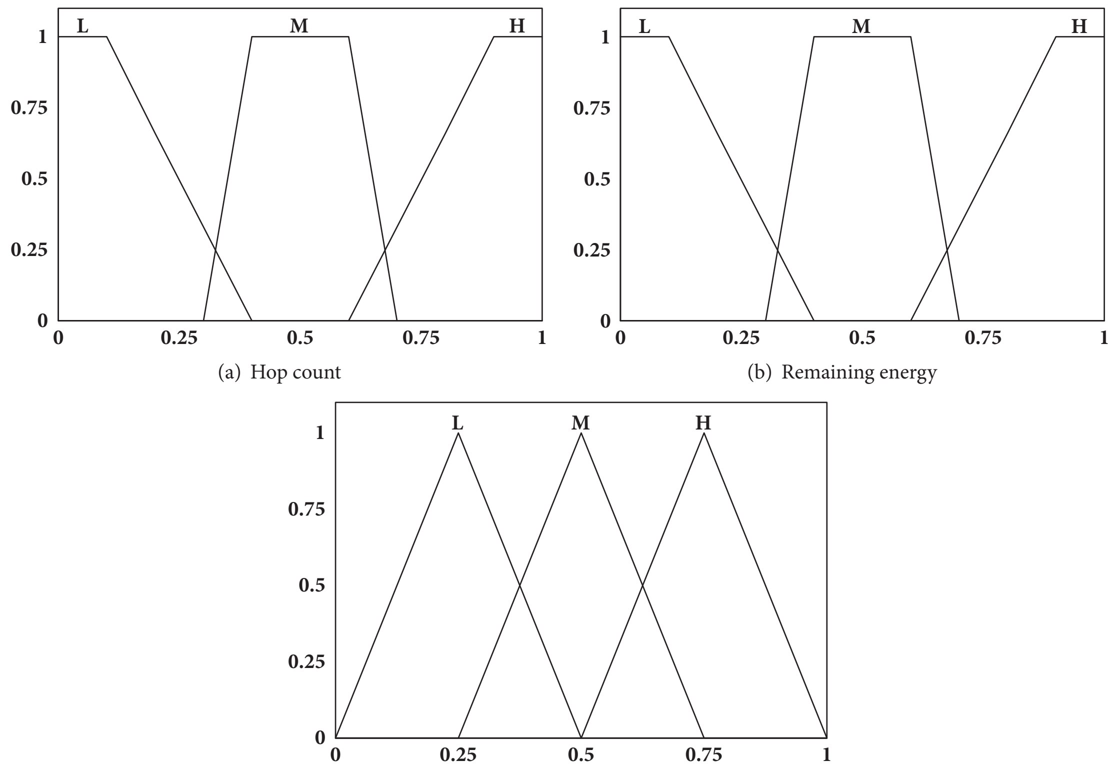
(A) Exponential Smoothing. In this research, we applied exponential smoothing, which is one of the most potent time-series prediction models. Its key advantage over short-term prediction was appropriately used for WSNs [34] to aid in path selection to avoid congestion in consideration of buffer occupancy. Equations (3) to (5) show the exponential smoothing derivation:

$$\widehat{Z}_t(1) = \frac{S_t}{W_t} \tag{3}$$

$$S_t = \sum_{\gamma=0}^{t-1} \alpha^{\gamma} Z_{t-\gamma} \tag{4}$$

$$\widehat{Z}_{t+1} = (1-\alpha) Z_t + \alpha \widehat{Z}_{t-1} \tag{5}$$

where $\widehat{Z}_t$ is the predicted value at time t (buffer), W_t denotes a geometric series, and α is a given constant



(a) Hop count

(b) Remaining energy

(c) Output weight

Figure 7: EW membership function: Hop count, remaining energy, and output weight.

(smoothing factor) in the range of 0.02 and 0.3. Equation (6) shows how to derive the predicted buffer, namely, BO over $\widehat{Z}_t$.

$$BO_{t+1} = (1-\alpha)\, BO_t + (\alpha)\, BO_{t-1} \tag{6}$$

(B) Adaptive Membership Function Using BAT. BAT, introduced by X. S. Yang et al. [35], is a naturally inspired approach used as a metaheuristic algorithm against an NP-complete problem [36] to find a good solution within a constraint-like resource (such as computational time complexity). According to a survey provided by A. H. Gandomi et al. [37], BAT is a potent algorithm in terms of performance compared with particle swarm optimization (PSO) and genetic algorithms (GAs).

BAT is also used for WSNs in several areas, such as localization [38] and radial optimization [39]. Thus, we adopt BAT to tune the membership function of FLS.

In general, BAT imitates bat behavior, in particular, the food scavenging behavior of microbats. A bat generates a high-frequency pulse to navigate toward a food resource; it also informs other bats of the location of the food. There are three main steps:

(i) Each bat sends a pulse (echolocation) to identify the position and distance for food, prey, and obstacle differentiation. The objective function $f(x)$ is defined in terms of the bat population x_i with velocity v_i and transmission frequency f_i to determine a better solution $f(x)$ in each round N. In each round, a better solution will be determined by adjusting the pulse rate (r_i) and loudness (l_i).

(ii) Each bat has no specific pattern for flying toward the food resource at velocity v_i from position x_i. The initial frequency is defined as f_{min}. During flying, wavelength λ and its amplitude or loudness l_i will be adjusted to seek the target. The bat can also change the pulse rate r_i in the range of $[0, 1]$ such that the rate is 0 in the initial pulse. The bat also increases the pulse rate once it is close to the target.

(iii) The loudness l_i is initially set to l_0 (maximum amplitude). During flying, the bat can search for the optimal solution. In contrast to the pulse rate, once the bat is close to the target, the loudness tends to be lowered, with l_{min} as the minimum amplitude.

In this research, we applied the bat behavior to adjust the membership function (MF_i) of FLS (see Algorithm 2). Here, $f(x) = MF_i$. In each round, we select the best MF, i.e., the MF with the maximum Euclidean distance (ED) between a specific MF and an initial MF. Typically, the population (number of bats = n) affects both the computational complexity and the accuracy gain [40]. We intensively investigated this trade-off and observed that, with 5 bats ($x_i = 5$), the accuracy was not significantly different from that with a larger number of bats. We defined an iteration threshold (N_i) as a scope of membership function adjustment to maintain the original shape of the function (either from trapezoidal to trapezoidal

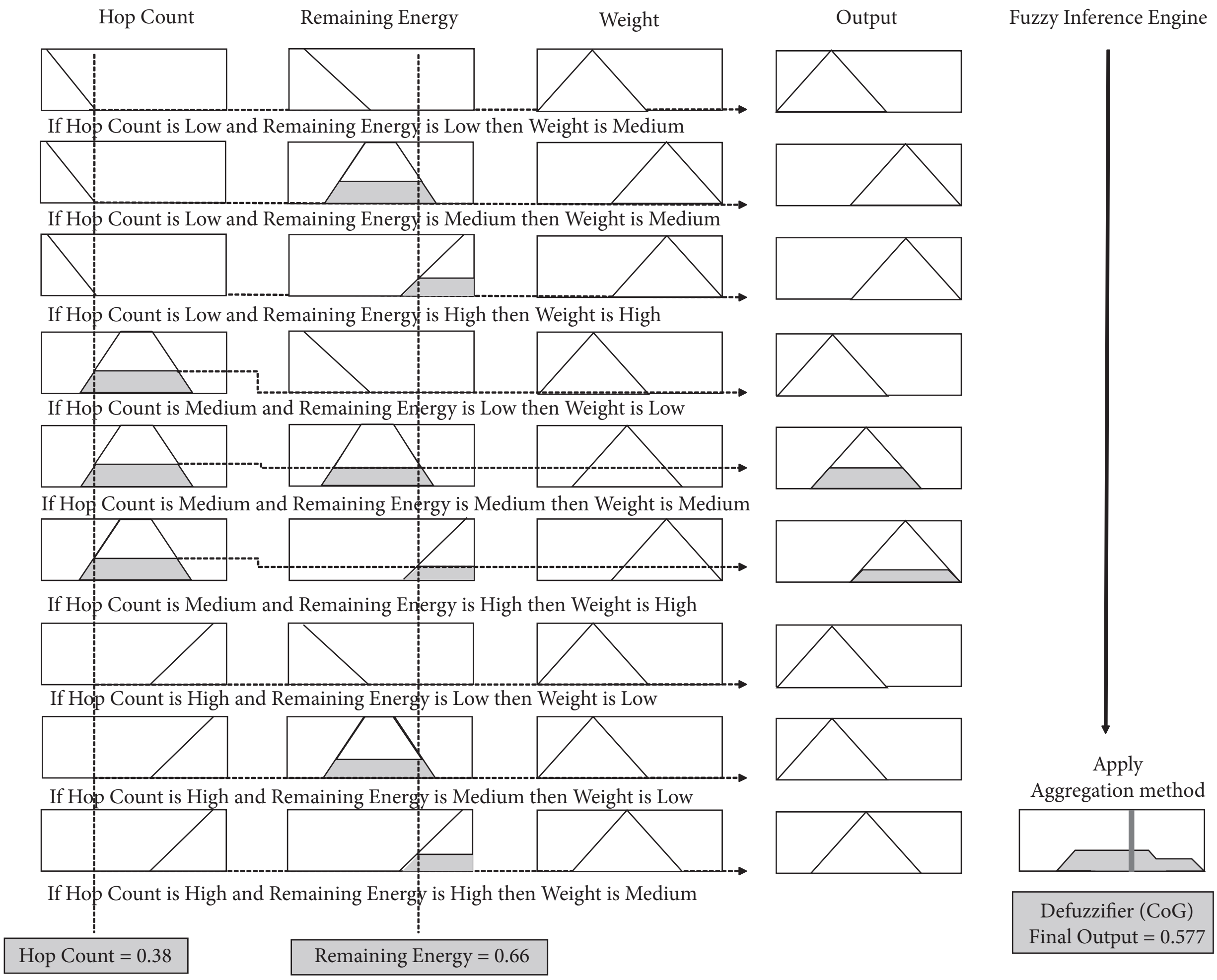


FIGURE 8: Example: EW fuzzy.

Objective function $f(x) = MF_i$
Initial population size $x_i \mid i \in 1, 2, \ldots, n$
Initial maximum iterations N_i, pulse rate r_i, loudness l_i, velocity v_i.
Define pulse frequency f_i
while (N_i)
Generate new solutions by adjusting f_i
Update vi and MF_i
  if (rand() $\longrightarrow r_i$)
    Select a solution and generate a local solution
  end if
    Generate a new solution with a random flying pattern
  if (rand() $< l_i$ & $f(x_i) < f(x)$)
    Accept new solution MF_i
    Increase r_i and decrease L_i
  end if
  Rank the bats and find the optimal MF_i using ED
end while

ALGORITHM 2: Membership adjustment using BAT.

Table 7: Example: fuzzy rule of α.

IF (Hop Size)	AND (Hop Count)	THEN (Weight)
L	L	H
L	M	H
L	H	M
M	L	L
M	M	M
M	H	M
H	L	L
H	M	L
H	H	M

Table 8: Example: fuzzy rules of AW.

IF (Buffer Occupancy)	AND (Forwarding Rate)	THEN (Weight)
L	L	L
L	M	H
L	H	VH
M	L	L
M	M	M
M	H	H
H	L	VL
H	M	L
H	H	M

or trapezoidal to triangular; see also Figures 9(a) and 9(c); 9(b) and 9(d) for α as smoothing factor; or Figures 10(a) and 10(c); 10(b) and 10(d) for AW as availability weight). We followed the configuration described by X. S. Yang et al. [35] for r_i, l_i, and f_i.

(C) Fuzzy Logic System with Adaptive Membership Function for Optimized Smoothing Factor. With the exponential smoothing discussed previously, the value of smoothing factor (α) can affect the prediction precision. Thus, we focus on the α derivation with regard to the relationship between hop size (number of hops from the source node) and hop count (number of hops from the sink node) using FLS.

Similar to the previous section (Section 4.2), we applied FLS in addition to membership tuning using BAT in different inputs; here, the two relationships (hop size and hop count) are used as inputs. Note that similar steps are applied.

For example, with Fuzzifier, we examined different membership functions (i.e., the two inputs with a triangular function for low and a trapezoidal function for medium and high) and used a triangular function for the output weight (see Figures 9(a) and 9(b)). Figures 9(c) and 9(d) show the membership after tuning with BAT. Table 7 shows an example with fuzzy rules; here, there are 9 rules.

Note that the same concepts of Fuzzy Inference Engine and Defuzzifier are applied (aggregation method and CoG; see also Figure 8) but with different inputs. Subsequently, the weight after defuzzification is used to determine α as the final output.

From Sections 4.3.1(A) and 4.3.1(B), we applied the optimized α (α_{opt}) for exponential smoothing with buffer as the parameter in a linear relationship, as described in

$$BO_{t+1} = \alpha_{opt} BO_t + \left(1 - \alpha_{opt}\right) BO_{t-1} \tag{7}$$

(D) Fuzzy Logic System with Adaptive Membership Function for Optimized Availability Weight. We considered the exponential smoothing for buffer occupancy (BO) prediction that was derived previously and the forwarding rate (FR) as the two input parameters over FLS for determining the availability weight (AW). Note that BO is used to determine the constraint of packet processing (queuing = drop if buffer is full), and FR is for the transmission capability. Note that these two factors are used to avoid a path that can lead to a high probability of packet loss (due to congestion, which results in queue unavailability and a low transmission rate).

Similar to the previous section (Section 4.2), we applied FLS in addition to membership tuning using BAT in different inputs: the predicted buffer occupancy and the forwarding rate. Similar steps were also applied. For example, with Fuzzifier, we examined different membership functions (i.e., the two inputs with triangular functions for low and high and a trapezoidal function for medium) and a triangular function for the output weight (to achieve high precision for high performance achievement, we used five membership levels instead of three, very low, low, medium, high, and very high; see also Figures 10(a) and 10(b)).

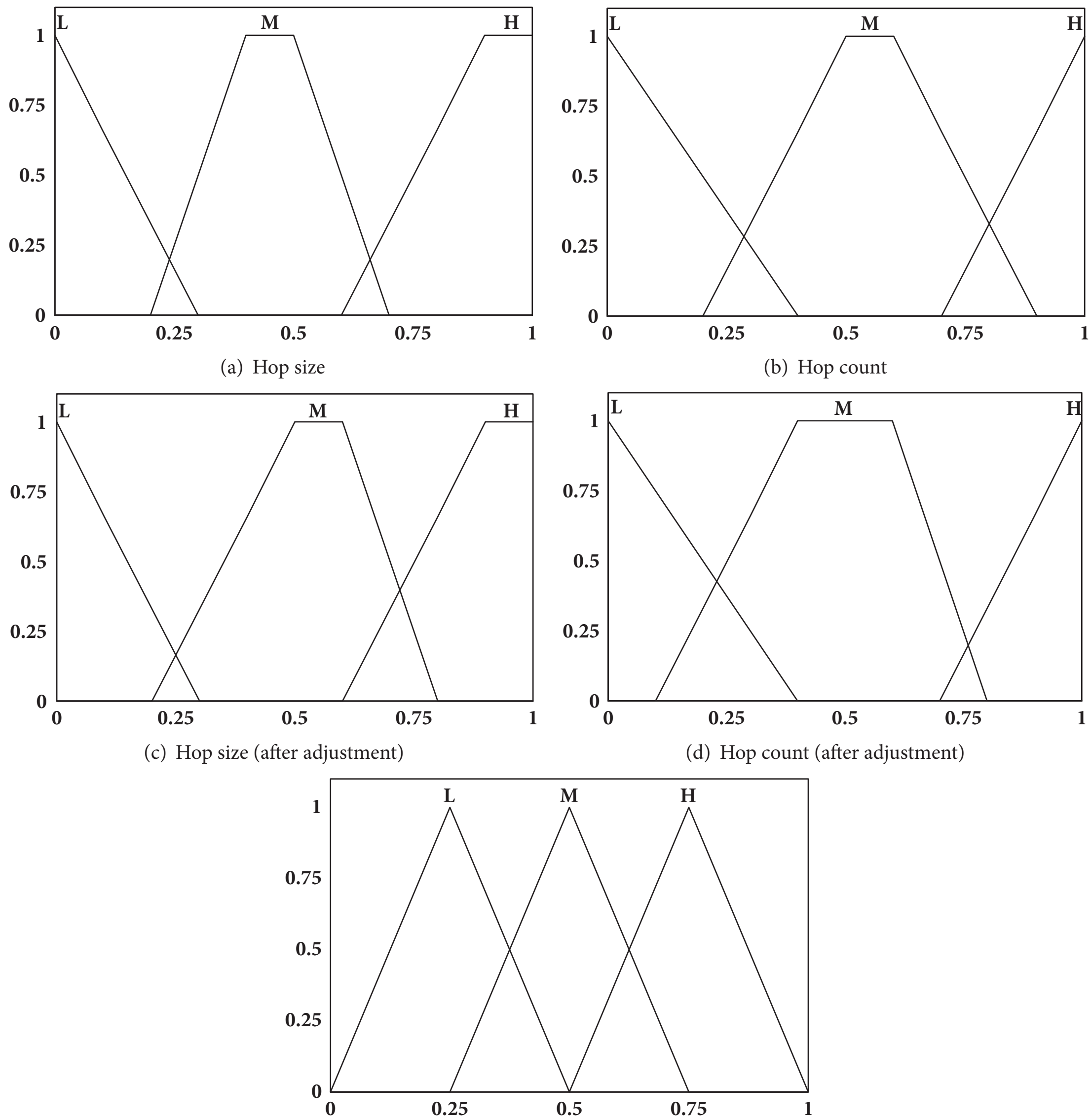
Figures 10(c) and 10(d) show the membership after tuning with BAT. Table 8 shows an example of fuzzy rules; here, there are 9 rules. The same concepts of Fuzzy Inference Engine and Defuzzifier are applied (aggregation method and CoG; see also Figure 8) but with different inputs. Subsequently, the weight after defuzzification is used to determine the AW as the final output.

Figure 11 also shows a detailed sequence diagram of our proposal, OFES.

4.4. Path Determination. As an example, shown in Figure 12, OFES can select the path from source to sink from 3 classes based on the user requirements and the presence of weight information.

(i) **Case I:** Top-Down Hierarchical Structure (Section 4.1). After the Route Create phase, all SNs determine their hop counts from the sink node. To send the packet upward, an SN can simply select the parent with the lowest hop count (stored in the neighbor table) to the sink node. In case there is more than one possible next hop, the preferred next hop is the lowest node identification (this is also applied for the other two cases below).

(ii) **Case II:** Energy-Assisted Routing Weight (Section 4.2). With an additional process for weight determination given the two factors (hop count and remaining energy), a particular SN can select the parent with the maximum EW from the neighbor table.



(a) Hop size
(b) Hop count
(c) Hop size (after adjustment)
(d) Hop count (after adjustment)
(e) Output weight

FIGURE 9: α membership function (before and after BAT tuning): hop count, hop size, and output weight.

(iii) **Case III:** Congestion Control and Prediction (Section 4.3). Similar to the second case, but with an additional weight (AW), here, we defined link score (LS) as a path selection criterion over EW and AW, as shown in (8). Here, γ is a weighted constant for determining a relationship between these two factors, which ranges between 0 and 1:

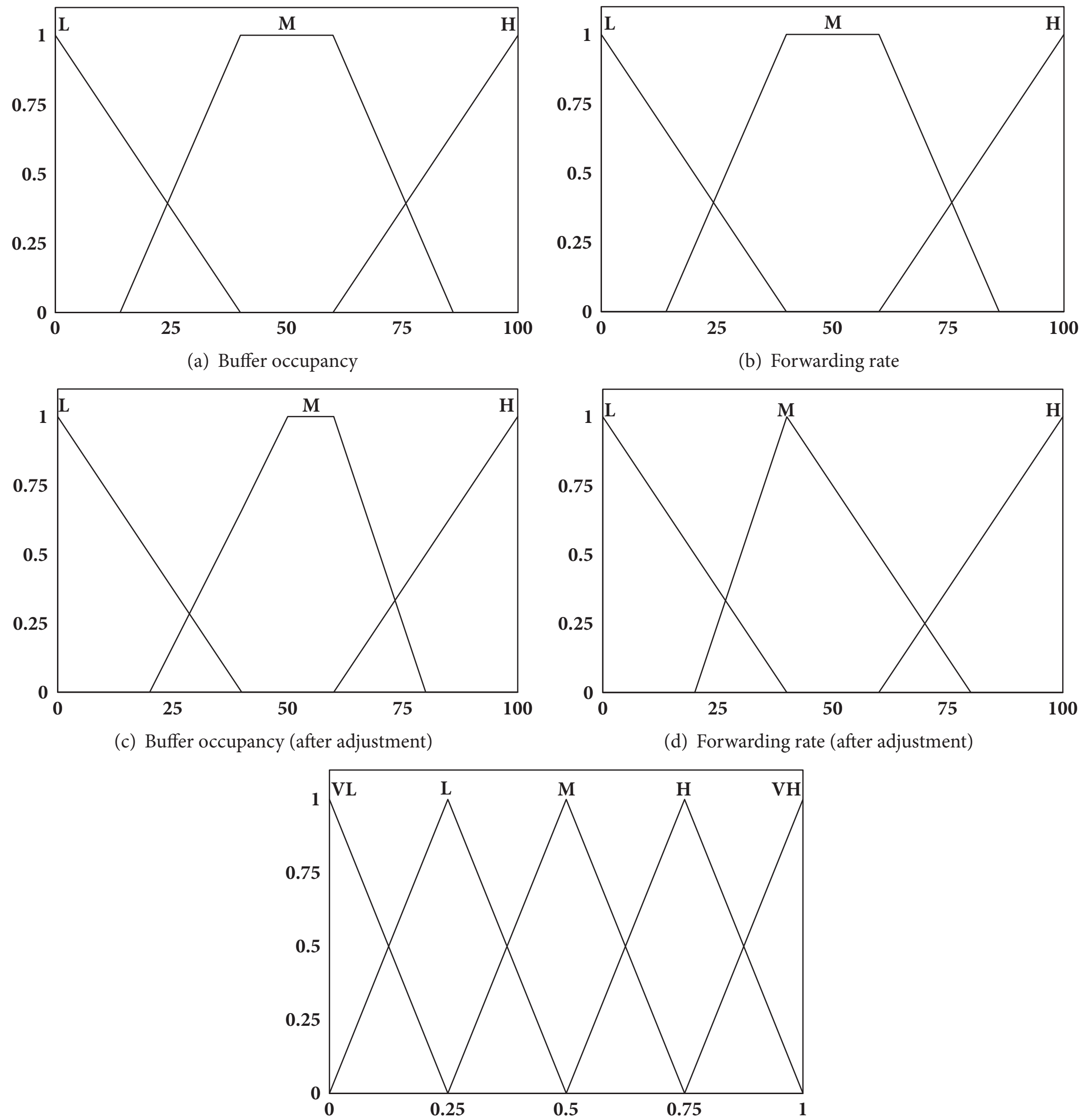
$$LS = \gamma AW + (1-\gamma) EW \tag{8}$$

Once LS is determined, a particular SN can select the parent with the maximum LS.

Figure 12 shows an example of path determination with different cases. Here, we only show the way in which the 20th SN selects a path given the neighbor table. Figure 12(a) shows Case I; here, the lowest hop count can be determined from 20 ⟶ 17 ⟶ 13 ⟶ 7 ⟶ 2 ⟶ sink node. However, in the second case, Figure 12(b) shows a different path selection (20 ⟶ 17 ⟶ 13 ⟶ 14 ⟶ 9 ⟶ 4 ⟶ sink node) with an additional weight (EW) (see Table 9). We select the maximum EW path. Figure 12(c) shows a different path (20 ⟶ 16 ⟶ 12 ⟶ 11 ⟶ 6 ⟶ 1 ⟶ sink node) while selecting LS (AW and EW) (see also Table 10). Note that after LS determination, the LSs of SNs 16, 17, and 19 are 0.875, 0.8, and 0.835, respectively. Thus, SN 16 is the parent of SN 20.

TABLE 9: Example:neighbor table of SN 20 with EW.

Node ID	Hop Count	EW (Weight)
16	4	0.89
17	4	0.65
19	5	0.83



(a) Buffer occupancy

(b) Forwarding rate

(c) Buffer occupancy (after adjustment)

(d) Forwarding rate (after adjustment)

(e) Output weight

Figure 10: AW membership function (before and after BAT tuning): buffer occupancy, forwarding rate, and output weight.

Table 10: Example: neighbor table of SN 20 with EW and AW.

Node ID	Hop Count	EW (Weight)	AW (Weight)
16	4	0.89	0.86
17	4	0.65	0.95
19	5	0.83	0.84

5. Performance Evaluation

To confirm the superior performance of OFES, we evaluated its performance in comparison with three state-of-the-art methods: HTAP [21], DAIPaS [22], and CCEbH [23], including the traditional ad hoc routing, AODV [31]. In addition, we evaluated our design criteria for the three possible path determinations, including the three cases: Top-Down Hierarchical Structure, Energy-Assisted Routing Weight, and Congestion Control and Prediction (with and without BAT tuning).

5.1. Experimental Design. Our algorithm was simulated using network simulation (NS2) version 2.35 with an extension on WSN [41, 42]. We simulated FLS using Octave 4.2. Our testbed was an Ubuntu 14.04 LTS. The SNs were randomly deployed in a topographical area A of dimension 200 m × 200 m. Once deployed, there was no mobility involved. The number of sensor nodes was 100, with 50 meters as the communication range (see an example shown in Figure 13). Note that the selected parameters follow the specifications

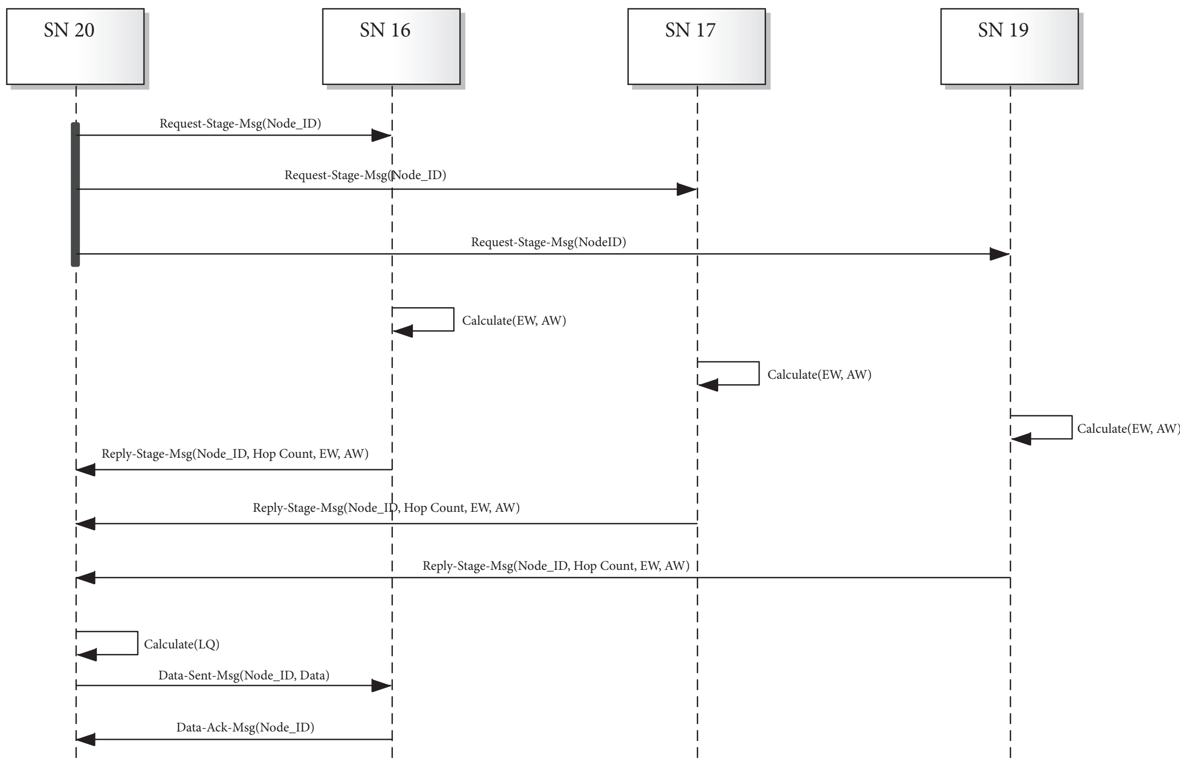
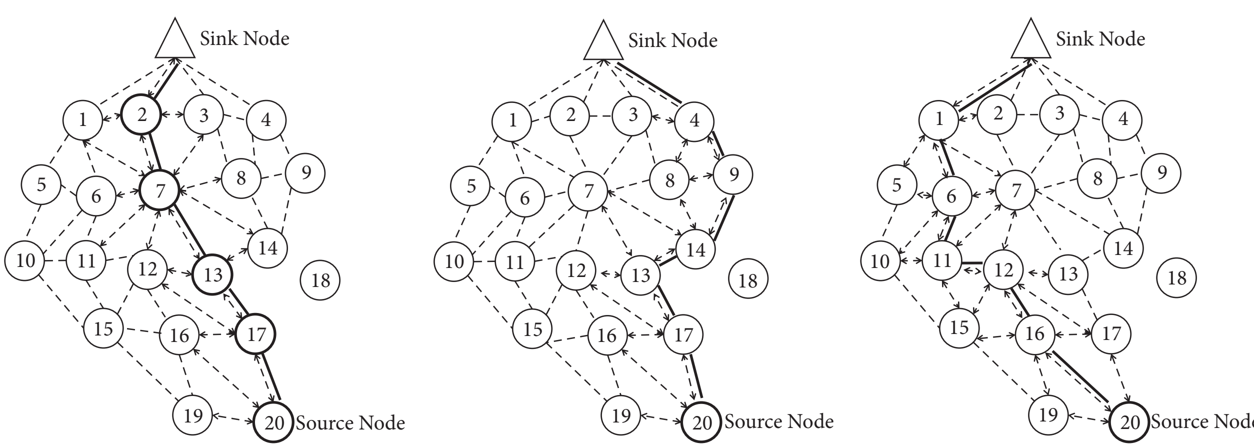


Figure 11: Sequence diagram of OFES.



(a) Case I (b) Case II (c) Case III

Figure 12: Example: path determination.

of MicaZ [30], HTAP [21], DAIPaS [22], and CCEbH [23].

To evaluate the efficiencies of the methods, experiments were conducted on both sparse and dense networks in terms of the number of events (more traffic), i.e., 10% and 20%, where each is from 32 Kbps to 224 Kbps (with a step size of 32 Kbps). The initial energy was 10 J [21–23]. The simulation was over 300 seconds with 10 trials, and the average result was calculated. The parameter values are listed in Table 11.

We also performed the intensive evaluation over the possible gamma (0 to 1) for optimal use in OFES at the two event rates and found that, at 0.48 and 0.62 in 10% and 20%, the event rate can outperform, i.e., 147 and 124 Kbps (see Figure 14), and we used these findings for further evaluation.

For comparison purposes with [21–23], three main metrics are used in the performance evaluation, namely, Average Throughput at Sink (ATS), Packet Delivery Ratio (PDR),

Table 11: Simulation parameters.

Parameter	Value
Sensor Node (Mote)	Micaz
Initial Energy	10 Joules
Transmission range	50 meters
Maximum bandwidth	250 Kbps
Number of sink node	1
Number of SNs	100
Event Rate	10% and 20%
Link Layer	CSMA/CA
Propagation model	Free Space
Antenna	Omni
Interface queue	Droptail
Packet size	128 bytes
Queue size	100 packets
Area	200 m^2
Simulation time	300 seconds

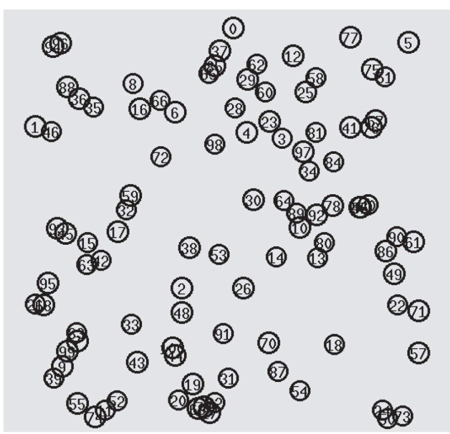


Figure 13: Node distribution.

and Average Remaining Energy (ARE), which are defined as follows:

$$ATS = \frac{\sum_{i=0}^{n} Rx\_sink_i}{\#Event\_Node} \tag{9}$$

$$PDR = \frac{\sum_{i=0}^{n} Packet\ Received}{\sum_{i=0}^{n} Packet\ Sent} \tag{10}$$

$$ARE = \frac{\sum_{i=0}^{n} \left(E_{init(i)} - E_{usage(i)}\right)}{\#SNs} \tag{11}$$

where Rx_sink_i is the received traffic from the ith SN by the sink node and $E_{init(i)}$ and $E_{usage(i)}$ are the initial and usage energies of the ith SN, respectively.

There are two scenarios to justify our performance: (1) we evaluated OFES with our experimental design justification over our three motivations as path determination, Case I (Top-Down Hierarchical Structure, OFES-Case-I), Case II (Energy-Assisted Routing Weight, OFES-Case-II), and Case III (Congestion Control and Prediction with and without BAT tuning, OFES-Case-III-woBAT and OFES-Case-III-withBAT).

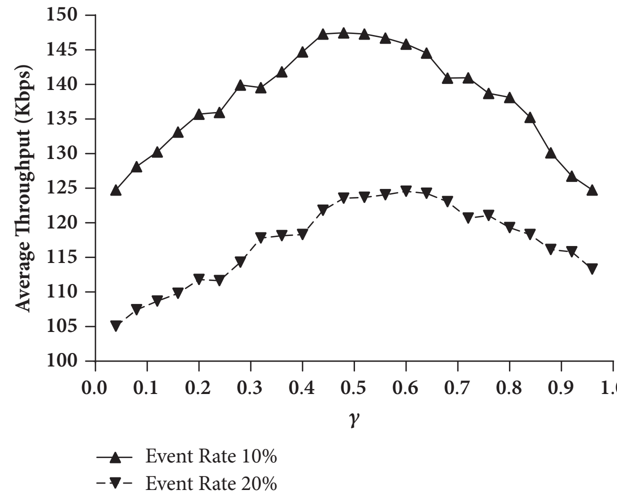


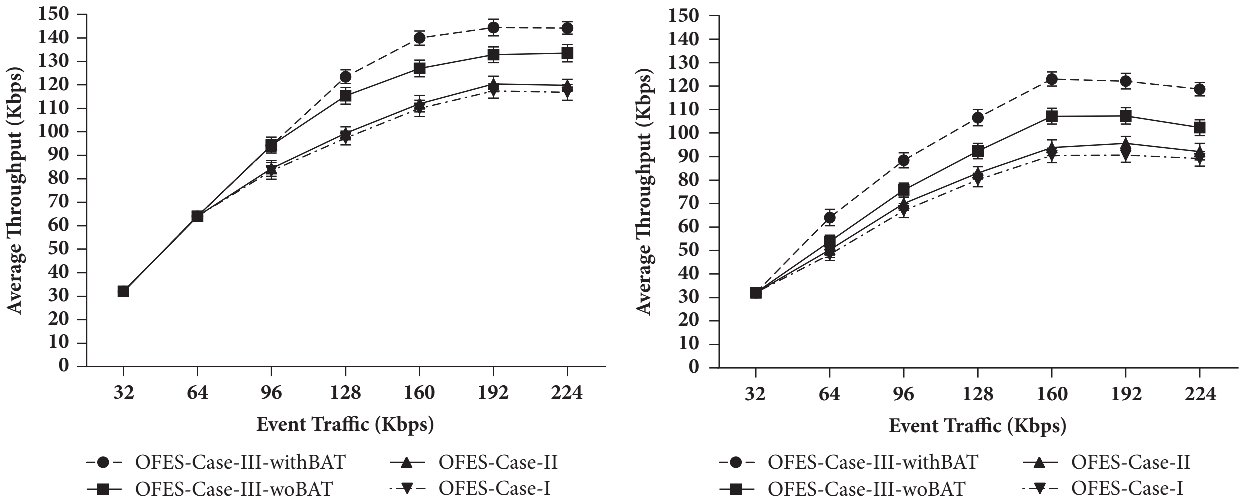
Figure 14: Average throughput over γ.

5.2. Simulation Results and Discussion. For the first scenario, our motivation with regard to the three cases of path determination, Figure 15 shows the performance in terms of ATS (Kbps). In general, as shown in Figure 15(a), our finalized OFES, OFES-Case-III-withBAT, outperforms its three derivations (OFES-Case-III-woBAT, OFES-Case-II, and OFES-Case-I) by factors of 7.25%, 21.61%, and 24.10%, with 10% events (sparse mode), respectively; and 13.28%, 28.44%, and 39.31%, with 20% events (dense mode); although the performance (throughput) is not in such an order. There is no congestion during low traffic, e.g., 32 to 64 Kbps in sparse mode and 32 Kbps in dense mode.

In terms of PDR, Figure 16 shows the results corresponding to ATS; however, the results are the opposite: the higher the traffic, the lower the PDR, for example, from 100% to 66%, 59%, 53%, and 52% in sparse mode and to 48%, 45%, 41%, and 39% in dense mode, for the four cases, respectively. Our finalized OFES, Case-III-withBAT, is superior on average to its derivatives.

Figure 17 shows the ARE values of our OFESs. In general, with more traffic, the remaining energy will decrease. For example, from 32 Kbps to 224 Kbps, traffic decreases the energy from 94% to 53%, 91% to 45%, 89% to 37%, and 85% to 32% in sparse mode and 86% to 27%, 82% to 20%, 80% to 16%, and 79% to 11% in dense mode, respectively. Our finalized OFES (OFES-Case-III-withBAT) is superior, and it outperforms the other three derivatives by 10.83%, 23.61%, and 41.39% in sparse mode and 13.31%, 32.91%, and 55.37% in dense mode.

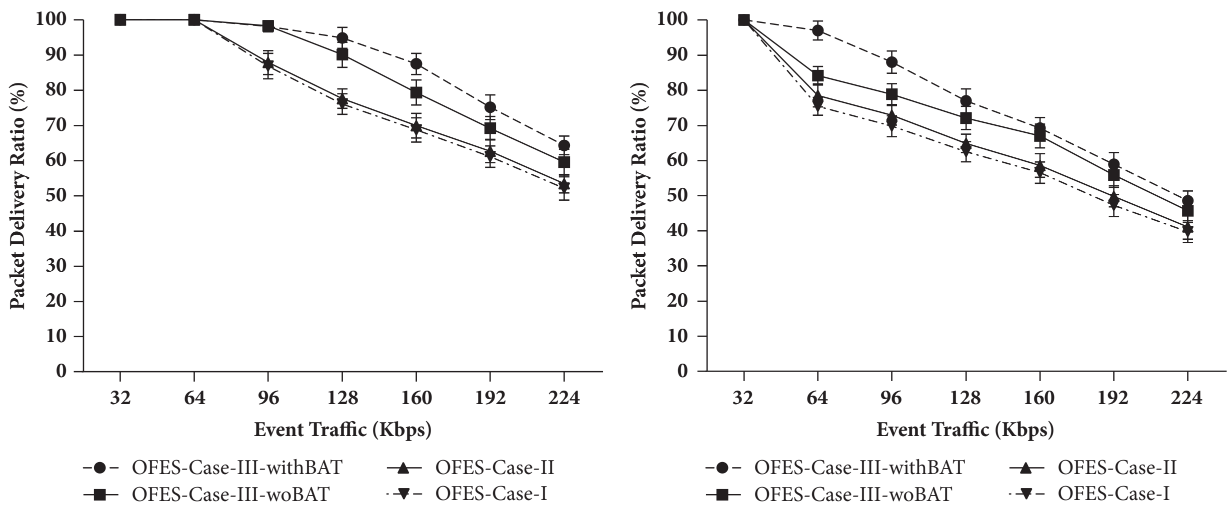
For the second scenario, compared with the recent congestion control algorithms in WSNs, Figure 18 shows the performance in terms of ATS (Kbps). In general, with the increase of event traffic, the average throughput decreases due to network congestion, for example, from 32 Kbps to 147, 133, 126, 119, and 92 Kbps, with 10% events (sparse mode), for OFES (aka OFES-Case-III-withBAT), CCEbH, DAIPaS,



(a) 10% event

(b) 20% event

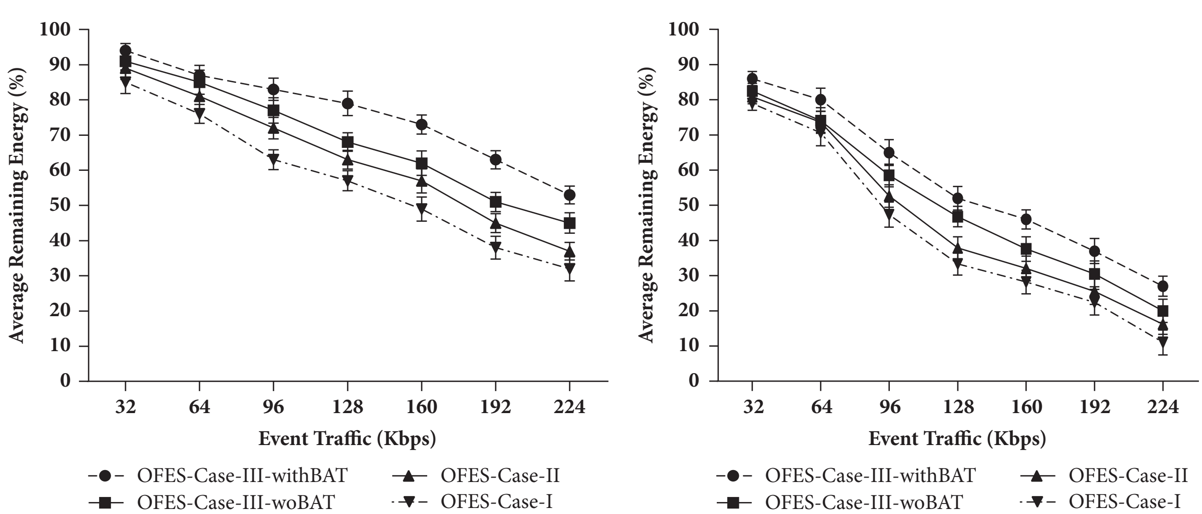
Figure 15: Average throughputs at sink node over different traffic loads.



(a) 10% event

(b) 20% event

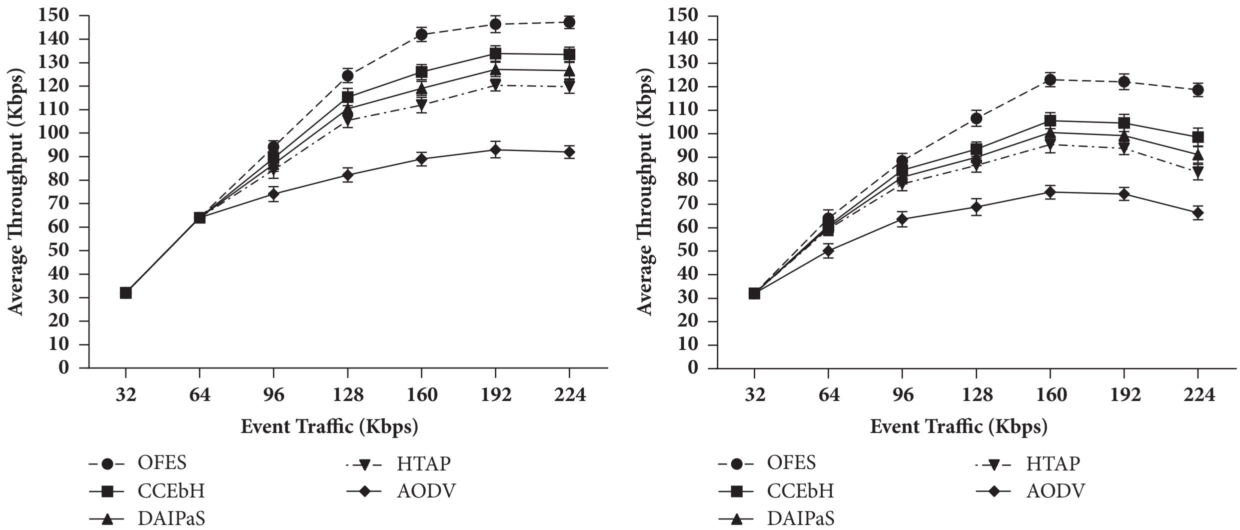
Figure 16: Packet delivery ratios over different traffic loads.



(a) 10% event

(b) 20% event

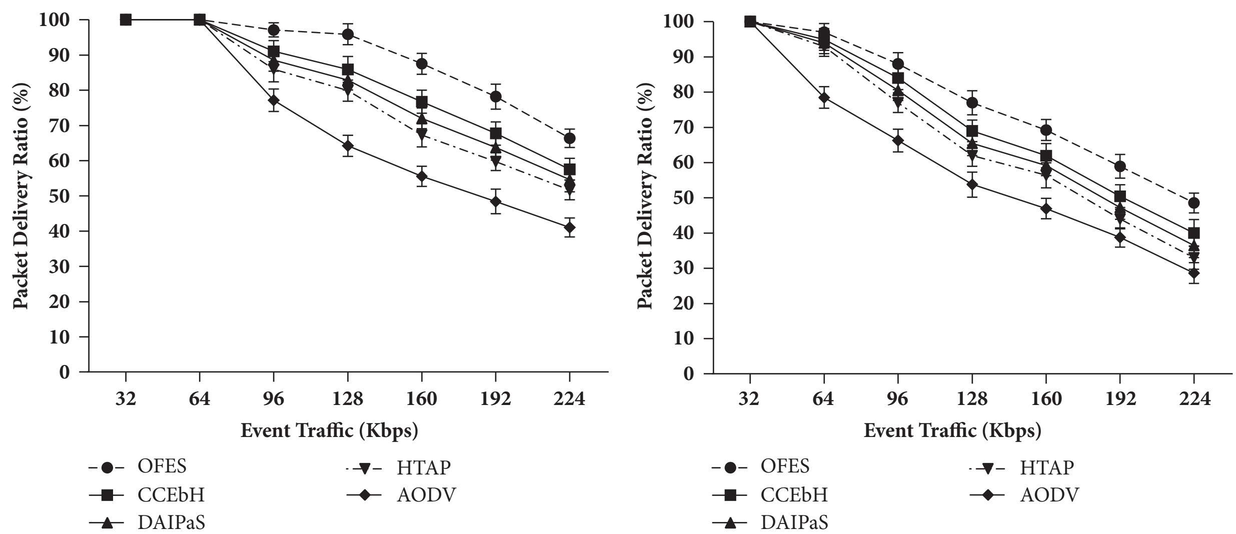
Figure 17: Average remaining energies over different traffic loads of OFESs (OFES-Case-I, OFES-Case-II, OFES-CaseIII-woBAT, and OFES-Case-III-withBAT).



(a) 10% event

(b) 20% event

Figure 18: Average throughputs at sink node over different traffic loads.



(a) 10% event

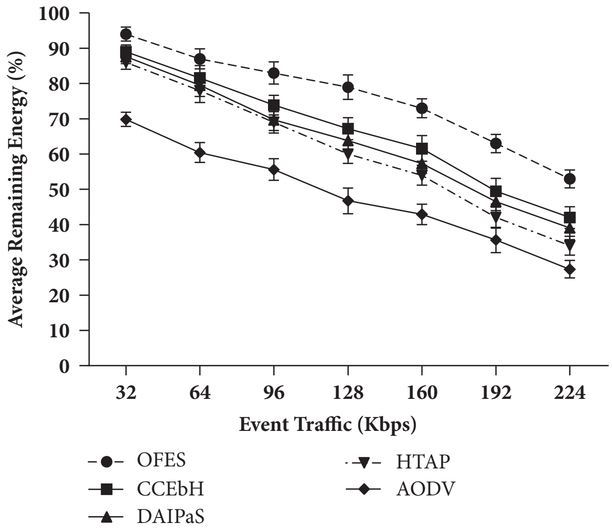
(b) 20% event

Figure 19: Packet delivery ratios over different traffic loads.

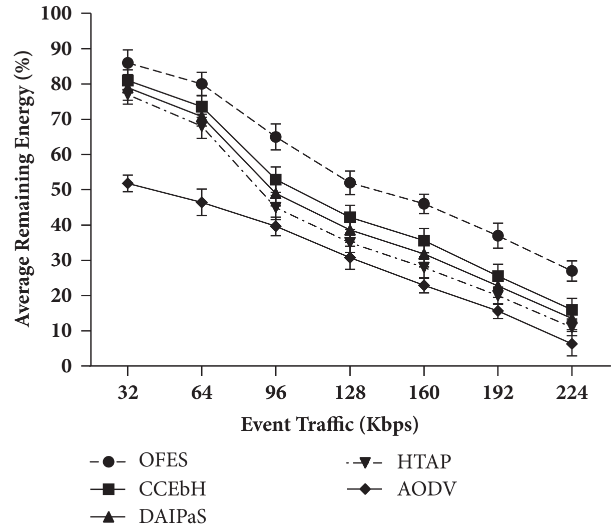
HTAP, and AODV, respectively; and from 32 Kbps to 118, 98, 91, 83, and 66 Kbps, with 20% events (dense mode).

In general, our proposed method outperforms the other four considered methods by factors of 8.32%, 14.37%, 20.18%, and 50.10% higher than the others. Note that, with low traffic, i.e., 32 and 64 Kbps, there is no congestion involved, so there is no loss (ATS is the same as the event traffic); however, with high traffic, the loss becomes obvious (low ATS on average). In dense mode, for 20% events, the ATS was similar despite the increased congestion. Thus, ATS tends to be lower in the dense mode than in the sparse mode. However, OFES is still superior, in particular, at 160 Kbps with ATS = 123 Kbps, and, on average, it outperforms the other four methods by 11.22%, 18.19%, 24.04%, and 54.70%. Again, note that there is no congestion at 32 Kbps traffic.

In terms of PDR, Figure 19 shows the results corresponding to ATS but in the opposite direction: with an increase in event traffic, PDR tends to decrease due to network congestion, for example, from 100% to approximately 66%, 57%, 54%, 51%, and 40%, with 10% events; and 48%, 40%, 36%, 33%, and 28%, with 20% events for OFES, CCEbH, DAIPaS, HTAP, and AODV, respectively. OFES is again superior on average to the other methods. Again, the network becomes congested at approximately 64 Kbps with 10% events. Similarly, in dense mode, where the number of events is 20%, PDR becomes much lower due to high congestion.



(a) 10% event



(b) 20% event

Figure 20: Average remaining energies over different traffic loads of OFES, CCEbH, DAIPaS, HTAP, and AODV.

Figure 20 shows ARE values of the three protocols. In general, as shown in Figure 20(a), with more traffic, the remaining energy will decrease, e.g., from 32 Kbps to 224 Kbps: from 94% to 53%, 89% to 42%, 87% to 39%, 86% to 34%, and 69% to 27%, respectively. Again, OFES is superior, and the overall remaining energy is high. Figure 20(b) also shows the results for the dense mode. OFES remains superior and outperforms CCEbH, DAIPaS, HTAP, and AODV by 13.62%, 22.56%, 30.55%, and 62.48%, respectively (20.81%, 42.37%, 59.58, and 82.27% for dense mode).

6. Conclusion and Future Work

In this research, we propose a path determination architecture for WSNs that takes into account network congestion, which is called the Optimized Fuzzy Logic-based Congestion Control Scheme with Exponential Smoothing Prediction in WSNs (OFES). For flexible use, three main path determination approaches can be applied based on the available resources and information: (1) a simplified level-based hierarchical tree path construction (using a hop count to find a path), (2) a more complex path determination method that uses known information, such as hop count and remaining energy, and (3) a path resolver with congestion prediction that uses exponential smoothing and two factors, such as buffer occupancy and forwarding rate. We also applied FLS to the last method to determine an adjustable weight in addition to weight optimization using BAT to tune the membership function.

To justify our motivation of path determination and to evaluate the performance of OFES, we first compared OFES with the other three cases, i.e., Case I (Top-Down Hierarchical Structure), Case II (Energy-Assisted Routing Weight), and Case III (Congestion Control and Prediction with and without BAT tuning). The results justify our design such that our finalized OFES (Case III with BAT tuning) is superior. Then, we compared the OFES with the three recently proposed methods that include the traditional ad hoc routing, namely, CCEbH, DAIPaS, HTAP, and AODV in terms of throughput, packet loss, and energy consumption. Through simulation experiments, we showed that OFES is superior in terms of all metrics, followed by CCEbH, DAIPaS, HTAP, and AODV.

For example, OFES outperformed the other four methods by factors of 9.77%, 16.28%, 22.11%, and 52.4%, respectively, in terms of average throughput and packet delivery ratio; and factors of 17.21%, 32.46%, 45.06% and 72.37%, respectively, in terms of average remaining energy. Note that although the performance of OFES is high, limitations and considerations for further investigation remain, such as the high volume of data transmission. In future work, comprehensive simulation and analysis could be carried out to investigate properties such as network density and diversity, network dimension, mobility, various signal propagation models, network failure probability, and heterogeneous data traffic, including physical device implementation.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Acknowledgments

This work was supported by grants from the Research Fund for Supporting Lecturers to Admit High Potential Students to Study and Research in Their Expert Program Year 2017 from the Graduate School, Khon Kaen University, Thailand

(Grant no. 601T213), and partially supported by the Thailand Research Fund (TRF) under Grant no. RTA6080013.

References

[1] Gartner. Press Release - Gartner Says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016, 2017. http://www.gartner.com/newsroom/id/3598917.

[2] Intel. A Guide to the Internet of Things Infographic, 2017. https://www.intel.com/content/www/us/en/internet-of-things/infographics/guide-to-iot.html.

[3] ABI Research, "Big Data and Analytics in IoT and M2M, 2016," https://www.abiresearch.com/market-research/product/1024282-big-data-and-analytics-in-iot-and-m2m/.

[4] I. Khan, F. Belqasmi, R. Glitho, N. Crespi, M. Morrow, and P. Polakos, "Wireless sensor network virtualization: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 553–576, 2016.

[5] M. A. Mahmood, W. K. G. Seah, and I. Welch, "Reliability in wireless sensor networks: a survey and challenges ahead," *Computer Networks*, vol. 79, pp. 166–187, 2015.

[6] T. G. Nguyen, C. So-In, and N. Nguyen, "Barrier Coverage Deployment Algorithms for Mobile Sensor Networks," *Journal of Internet Technology*, vol. 18, no. 7, pp. 1689–1699, December 2017.

[7] H. Yetgin, K. T. K. Cheung, M. El-Hajjar, and L. Hanzo, "A Survey of Network Lifetime Maximization Techniques in Wireless Sensor Networks," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 828–854, 2017.

[8] D. Chen, Z. Liu, L. Wang, M. Dou, J. Chen, and H. Li, "Natural disaster monitoring with wireless sensor networks: A case study of data-intensive applications upon low-cost scalable systems," *Mobile Networks and Applications*, vol. 18, no. 5, pp. 651–663, 2013.

[9] S. Halder and S. D. Bit, "Enhancement of wireless sensor network lifetime by deploying heterogeneous nodes," *Journal of Network and Computer Applications*, vol. 38, no. 1, pp. 106–124, 2014.

[10] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, and F. Senel, "Topology management techniques for tolerating node failures in wireless sensor networks: a survey," *Computer Networks*, vol. 58, no. 1, pp. 254–283, 2014.

[11] M. A. Kafi, D. Djenouri, J. Ben-Othman, and N. Badache, "Congestion control protocols in wireless sensor networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1369–1390, 2014.

[12] B. Rashid and M. H. Rehmani, "Applications of wireless sensor networks for urban areas: A survey," *Journal of Network and Computer Applications*, vol. 60, pp. 192–219, 2016.

[13] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *The Journal of Supercomputing*, vol. 68, no. 1, pp. 1–48, 2014.

[14] S.-U. Lar and X. Liao, "An initiative for a classified bibliography on TCP/IP congestion control," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 126–133, 2013.

[15] C. Lochert, B. Scheuermann, and M. Mauve, "A survey on congestion control for mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 7, no. 5, pp. 655–676, 2007.

[16] J. A. Khan, H. K. Qureshi, and A. Iqbal, "Energy management in Wireless Sensor Networks: A survey," *Computers & Electrical Engineering*, vol. 41, pp. 159–176, 2015.

[17] C. Sergiou, P. Antoniou, and V. Vassiliou, "A comprehensive survey of congestion control protocols in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1839–1859, 2014.

[18] S. A. Shah, B. Nazir, and I. A. Khan, "Congestion control algorithms in wireless sensor networks: Trends and opportunities," *Journal of King Saud University - Computer and Information Sciences*, vol. 29, no. 3, pp. 236–245, 2017.

[19] A. Ghaffari, "Congestion control mechanisms in wireless sensor networks: a survey," *Journal of Network and Computer Applications*, vol. 52, pp. 101–115, 2015.

[20] J. Kang, Y. Zhang, and B. Nath, "TARA: Topology-aware resource adaptation to alleviate congestion in sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 7, pp. 919–931, 2007.

[21] C. Sergiou, V. Vassiliou, and A. Paphitis, "Hierarchical Tree Alternative Path (HTAP) algorithm for congestion control in wireless sensor networks," *Ad Hoc Networks*, vol. 11, no. 1, pp. 257–272, 2013.

[22] C. Sergiou, V. Vassiliou, and A. Paphitis, "Congestion control in Wireless Sensor Networks through dynamic alternative path selection," *Computer Networks*, vol. 75, pp. 226–238, 2014.

[23] W. Chen, Y. Niu, and Y. Zou, "Congestion control and energy-balanced scheme based on the hierarchy for WSNs," *IET Wireless Sensor Systems*, vol. 7, no. 1, pp. 1–8, 2017.

[24] S. Jaiswal and A. Yadav, "Fuzzy based adaptive congestion control in wireless sensor networks," in *Proceedings of the 2013 Sixth International Conference on Contemporary Computing (IC3)*, pp. 433–438, Noida, India, August 2013.

[25] Y. Zhang, J. Wang, D. Han, H. Wu, and R. Zhou, "Fuzzy-Logic Based Distributed Energy-Efficient Clustering Algorithm for Wireless Sensor Networks," *Sensors*, vol. 17, no. 7, pp. 1554:1–1554:21, 2017.

[26] H. D. Nikokheslat and A. Ghaffari, "Protocol for Controlling Congestion in Wireless Sensor Networks," *Wireless Personal Communications*, vol. 95, no. 3, pp. 3233–3251, 2017.

[27] H. Z. Shi, R. V. Prasad, E. Onur, and I. G. M. M. Niemegeers, "Fairness in wireless networks: issues, measures and challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 5–24, 2014.

[28] A. Uthra Rajan, S. V. Kasmir Raja, A. Jeyasekar, and A. J. Lattanze, "Energy-efficient predictive congestion control for wireless sensor networks," *IET Wireless Sensor Systems*, vol. 5, no. 3, pp. 115–123, 2015.

[29] T. Rault, A. Bouabdallah, and Y. Challal, "Energy efficiency in wireless sensor networks: a top-down survey," *Computer Networks*, vol. 67, pp. 104–122, 2014.

[30] Crossbow technology inc., "MiCaZ Datasheet," Document Part No. 6020-0060-04, January, 2006.

[31] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," No. RFC 3561, 2003.

[32] G. Han, L. Liu, J. Jiang, L. Shu, and G. Hancke, "Analysis of Energy-Efficient Connected Target Coverage Algorithms for Industrial Wireless Sensor Networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 1, pp. 135–143, December 2017.

[33] P. Nayak and B. Vathasavai, "Energy Efficient Clustering Algorithm for Multi-Hop Wireless Sensor Network Using Type-2 Fuzzy Logic," *IEEE Sensors Journal*, vol. 17, no. 14, pp. 4492–4499, 2017.

[34] R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting with exponential smoothing: the state space approach*, Springer Science & Business Media, 2008.

[35] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature inspired cooperative strategies for optimization*, vol. 248, pp. 65–74, 2010.

[36] X.-S. Yang and A. H. Gandomi, "Bat algorithm: A novel approach for global engineering optimization," *Engineering Computations*, vol. 29, no. 5, pp. 464–483, 2012.

[37] A. H. Gandomi, X. S. Yang, A. H. Alavi, and S. Talatahari, "Bat algorithm for constrained optimization tasks," *Neural Computing and Applications*, vol. 22, no. 6, pp. 1239–1255, 2013.

[38] C. So-In and W. Katekaew, "Hybrid Fuzzy Centroid with MDV-Hop BAT Localization Algorithms in Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 10, article 894560, 2015.

[39] S. P. Kaur and M. Sharma, "Radially optimized zone-divided energy-aware Wireless Sensor Networks (WSN) protocol using BA (Bat Algorithm)," *IETE Journal of Research*, vol. 61, no. 2, pp. 170–179, 2015.

[40] S. Yilmaz and E. U. Küçüksille, "A new modification approach on bat algorithm for solving optimization problems," *Applied Soft Computing*, vol. 28, pp. 259–275, 2015.

[41] K. Fall and K. Varadhan, "The network simulator (ns-2), 2007," http://www.isi.edu/nsnam/ns.

[42] Manna Research Group, "Mannasim framework, 2010," http://www.mannasim.dcc.ufmg.br/index.htm.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 1028578, 17 pages
https://doi.org/10.1155/2018/1028578

Research Article

An Open IoT Platform to Promote Eco-Sustainable Innovation in Western Africa: Real Urban and Rural Testbeds

Corentin Dupont,[1] Massimo Vecchio ,[1] Congduc Pham,[2] Babacar Diop,[3] Charlotte Dupont,[4] and Sename Koffi[5]

[1] *FBK CREATE-NET, Italy*
[2] *University of Pau and Pays de l'Adour, France*
[3] *Gaston Berger University, Senegal*
[4] *Easy Global Market, France*
[5] *WoeLab, Togo*

Correspondence should be addressed to Massimo Vecchio; mvecchio@fbk.eu

Received 12 January 2018; Revised 18 May 2018; Accepted 5 June 2018; Published 1 August 2018

Academic Editor: Al-Sakib K. Pathan

The digital revolution led by the Internet of Things (IoT) is already reshaping several traditional business sectors. Moreover, because of its very nature, the promise of the IoT is also to reduce energy consumption and pollutant emissions in several environmental scenarios. At the same time, it is desirable to keep the development of IoT as sustainable as possible, hence truly realizing the vision of the *green IoT*. In this paper, we show how a full-stack IoT framework can alleviate some real environmental problems afflicting countries in Western Africa. We present three real IoT-based deployments currently hosted in two rural areas of Senegal and Ghana and one metropolitan area of Togo. These testbeds are connected to a Cloud-based software platform, purposely designed and engineered to address some very specific environmental, economic, and social requirements of the region.

1. Introduction

Since its coinage by K. Ashton in 1999 [1], the term "Internet of Things" (IoT) has attracted a multitude of research and industrial interests. Today, the IoT paradigm embraces numerous architectures, protocols, standards, services, and applications for ubiquitous data acquisition and large-scale data analysis. A wide range of domains is impacted: housing, precision agriculture, retail, transportation, infrastructure monitoring, and personal healthcare, just to mention a few [2]. In all these domains, autonomous devices (i.e., "the things") are able to collect data and push them to the Internet. Authoritative sources like Gartner expect that, by 2020, around 26 billion objects will be connected to the Internet (https://www.gartner.com/newsroom/id/2636073) while, according to IDC (https://www.forbes.com/sites/louiscolumbus/2017/12/10/2017-roundup-of-internet-of-things-forecasts/#5a432b8c1480), worldwide spending on IoT is forecasted to reach 772.5 billion USD in 2018, hence with an increase of 14.6% over the 674 billion USD predicted for 2017. These numbers reveal that the IoT digital revolution is already happening: a huge amount of connected objects will be deployed everywhere within a few years, making the world we are living in ultraconnected, hence also smarter.

However, for such a smarter world to be sustainable, its enabling technologies should be characterized by a certain degree of energy efficiency [3]. In other words, these billions of things equipped with sensing, processing, and wireless communication capabilities will require energy to operate, usually provided by on-board batteries [4]. For this reason, energy efficiency will be one of the key elements for the implementation and development of the so-called *green IoT* concept [5].

In recent years, Western Africa registered a very rapid economic development but, unfortunately, this fast development made environmental problems grow proportionally, in a region where dramatic climate changes and natural disasters already represent huge threats in both rural and metropolitan

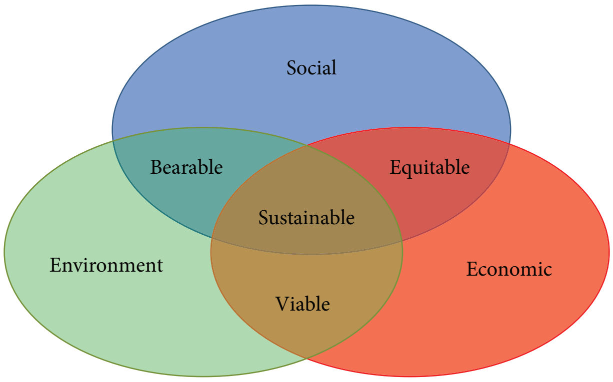


Figure 1: The Venn diagram of sustainable development.

areas. However, the adoption of the IoT in this region has the potential of being disruptive in several sectors [6, 7]. To better understand this implication, consider what happened in Africa with the telecommunications services: currently, 93% of the African population has access to mobile phones, while landlines phones were never massively deployed in the continent! Thus, the African continent has the potential of directly jumping into the open IoT arena, completely skipping any siloed business or industry. Then, it is not coincidence that Sonatel (a subsidiary company of Orange telecommunications provider in Africa) has inaugurated a FIWARE (https://www.fiware.org/) hub in Dakar, Senegal, while multinational corporations such as IBM (https://www.ibm.com/blogs/research/2016/10/forecasting-air-quality-south-africa-iot/), SAP (https://news.sap.com/africa/2016/12/19/iot-connecting-africa/) and Google (http://allafrica.com/stories/201705170571.html) have already established foot on the continent. Furthermore, in Africa, the rising phenomenon of the so-called "tech hubs" is taking a big role in the global and capillary diffusion of IT technologies in general (of IoT, in particular). As we write, for instance, the AfriLabs (http://www.afrilabs.com/) initiative already comprises more than 80 of such innovation centers across 27 African countries.

Generally speaking, as summarized in Figure 1, for a solution to be successful, it should not only be economically viable but also environmentally friendly and socially integrated [8]. In this specific case, socially valid approaches are inherently tied together to a solid growth of the African SME sector [9]. Consider, for instance, that, according to the African Economic Outlook 2017 (a report jointly produced by the African Development Bank Group, the Organization for Economic Cooperation and Development, and the United Nations Development Programme), about 80% of Africans consider entrepreneurship as a good career opportunity, revealing the very lively entrepreneurial culture of this continent [10].

In this paper, we propose a full-stack IoT framework which is able to alleviate some of the unique environmental challenges afflicting Western Africa's countries, while being also socially accessible and economically effective. With the goal of keeping down the costs of the sensing and actuating infrastructure, we adopted a Do-It-Yourself (DIY) approach, integrating only low-power and low-cost off-the-shelf components. This strategy, in fact, allows local partners and end-users to easily reproduce the solution in a sustainable and cost-effective way. Nevertheless, the engineered devices result to be very environmentally friendly, being able to run for at least two years with a couple of standard AA batteries. At software side, the Cloud-based infrastructure offers very peculiar services able to meet some of the unique requirements of this region. Examples of requirements are notifications via SMS and voice messages, resiliency against the low technical expertise of the average end-user, and robustness against frequent Internet disconnections. We deployed and tested our framework in three real testbeds. The first use case tackles water and soil management with a local variety of tomato crops. It is deployed in an experimental farm at Gaston Berger University, Saint-Louis, Senegal. The second use case, deployed in Ghana, shows that low-cost IoT devices can help improve water quality in fish farms. Finally, the third use case shows how IoT devices can be used to effectively perform waste management in African urban scenarios. It is deployed in Lomé, the capital and largest city of Togo.

The remainder of this paper is organized as follows: Section 2 reviews the most recent related works, while Section 3 describes the current major challenges to be faced by the IoT in Western African countries. Then, Section 4 thoroughly describes the proposed platform, while the three real testbeds deployed in Senegal, Ghana, and Togo are thoroughly described in Section 5. Section 6 concludes this paper.

2. Related Works

2.1. From Green IT to Green IoT. According to the IoT paradigm, each day more devices are connected to the Internet. It follows that, in the near future, everything around us will be ultraconnected. The huge amounts of digital information produced need to be transferred from/to the devices through the Internet itself, hence generating high volumes of data traffic. Moreover, such exponential growth in data production requires the application of very advanced digital technologies able to quickly filter, process, and store the data flowing from devices to data centers (e.g., big data, big stream, and data analytics [11]). Thus, both data and communications systems need to implement and adopt some power management policy to reduce their CO2 emissions and become truly environmentally sustainable. This is the challenge that the *green IT* branch sets itself. More in detail, in [12] we find a formal definition of "green IT" as a holistic approach able to benefit the environment, by improving energy efficiency, lowering greenhouse gas emissions, using less harmful materials, and encouraging reuse and recycling. In other words, the *"greening"* process may be realized either by acting at the IT system itself or in the way that such IT system is used. For instance, the authors of [13] analyze some power management policies for both processing and communications subsystems (i.e., holistically), with the goal of wisely trading-off power consumption and quality of service (QoS) delivered to the users.

With reference to data subsystem (and so to the Cloud side of our framework), different strategies to reduce power consumption are currently investigated in the literature. For instance, the authors of [14], after noticing that standard approaches to resource overprovisioning are excessively power hungry, proposed some novel, dynamic virtual machine consolidation techniques. Those techniques are able to reduce energy consumption without compromising any service level agreement, hence without undermining QoS. For a comprehensive review of techniques and approaches recently proposed in the literature to reduce power consumption in Cloud-based environments along the whole technology stack (i.e., from the service request issued by users to the actual Cloud service provider), we invite the interested reader to refer to [15].

With reference to green communications, there are several technological challenges still open [16]. The concept of energy efficiency is gaining back popularity, although mostly to alleviate economic concerns rather than ecological ones. Indeed, the consumption of energy by telecom operators, together with the cost of energy, is showing continuously growing trends [17]. A wide portion of the research community is currently putting efforts in defining and implementing the fifth generation (5G) wireless cellular network technology, so as to support the drastic demands of user subscriptions and data bandwidths. Clearly, also in this case energy consumption represents a concern, as users' terminals are still powered by batteries. For a detailed review of the major challenges in 5G green communications, the reader is referred to [18]. However, the assumption here is that batteries may be recharged when needed, while this is not the case for the vast majority of the billions IoT devices expected to spread out in the near future. In this context, a standard requirement for an IoT application is to run without manual intervention (i.e., autonomously) for years. It follows that the only way forward to engineer this kind of applications is to be as power-aware as possible. For this reason, the IoT is considered as an umbrella of technologies that should be green by design.

2.2. Green IoT in Western Countries. In the literature ([5, 19]) "Green IoT" is defined as "the energy-efficient procedures (hardware or software) adopted by IoT either to facilitate reducing the greenhouse effect of existing applications and services or to reduce the impact of the greenhouse effect of IoT itself. In the earlier case, the use of IoT will help reduce the greenhouse effect, whereas in the latter case further optimization of IoT greenhouse footprint will be taken care. The entire life cycle of green IoT should focus on green design, green production, green utilization, and finally green disposal/recycling to have no or very small impact on the environment". A recent, extensive report from Cisco Systems Inc. provides several precious insights into the envisioned role played by some of the most modern IoT technologies. This report also highlights a number of scenarios where the IoT is practically solving (or, at least, alleviating) some of the world's most challenging development and sustainability issues [7].

According to the Food and Agriculture Organization of the United Nations (FAO), the current world population of 7.6 billion is expected to reach 9.8 billion in 2050 and nearly all of this population increase will occur in developing countries [20]. In order to feed this larger population, food production must increase at least by 70%. Looking at these numbers, it is clear that improvin the farm productivity is an essential factor in the survival of the human race. Within this challenge, the green IoT is envisioned to play a leading role, being it one of the major technologies of the so-called smart farming sector, sometimes also referred to as precision agriculture. The authors of [21] define *precision agriculture* as "a suite of IT-based tools which allow farmers to electronically monitor soil and crop conditions and analyze treatment options", also highlighting the importance of compatibility between enabling technology and farmers' expertise. Recently, the authors of [22] described the design of an IoT-based platform, SmartFarmNet, as being able to collect environmental data coming from fields and store such data in the Cloud for performance analysis and recommendations. Similarly, [23] employed the OpenIoT middleware infrastructure (http://www.openiot.eu) to integrate a Phenonet deployment in Australia (https://data61.csiro.au/en/Our-Work/Monitoring-the-Environment/Sensing-the-environment/Phenonet). Specifically, the latter employs a state-of-the-art sensor network technology to gather environmental data for crop variety trials at a higher resolution than conventional methods. Then, thanks also to the features enabled by the OpenIoT platform, scientists and farmers are able to visualize, process, and extract both real-time and long-term crop performance information from the acquired sensor measurements. Smart farming applications include also farm vehicle tracking, livestock monitoring, greenhouse monitoring and control, and storage monitoring, only to mention few examples. Livestock sensors on-board IoT devices can, for instance, electronically monitor the animals' vital signs [24] or notify ranchers when animals have roamed from the herd. Soil sensors can alert farmers to irregular conditions, giving the farmer time to reconcile the issue and produce better crops. This is the case, for instance, for [25], where the authors perform temperature control in a greenhouse using Zigbee-based wireless communications.

Some recent approaches have also introduced another architectural component, namely, the IoT gateway. IoT gateways are used by the authors of [6, 26], where they present a system architecture for remote agriculture process automation. The peculiarity of these works is that the sensors and actuators are wirelessly connected to an IoT gateway, which is able to perform the monitoring and controlling process.

Getting back to the FAO's most recent forecasts, around 66% of the world's population is expected to live in urban areas by 2050. It is obvious that cities will need to increase the efficiency in which they operate and use their resources, in order to meet the demands imposed by such a dramatic urbanization. In this context, the main challenge is to keep providing basic resources, such as sufficient fresh water, clean energy, public transportation, safety, and security. These resources should be provided while also ensuring economic, social, and environmental sustainability [27]. In other words, the constant improvement and enhancement of the citizens' quality of life will continue to be the main goal of every modern city of the future. In this challenging arena, green IoT is expected to play a crucial role. However, collecting, analyzing, and exploiting large amounts of data generated by many heterogeneous sources spread throughout the city represent very challenging issues [28].

2.3. Green IoT in Africa. The penetration level of IoT in Africa is much lower than in other world regions, as also highlighted in [29, 30]. To quantify this gap, it is sufficient to look at the fixed broadband Internet subscription rates (https://www.statista.com/statistics/370681/fixed-broadband-internet-penetration-region/) of 2017: 1 person every 100 inhabitants in Africa, against 37.2 in Europe. Although this gap is reduced when dealing with the mobile broadband Internet subscription rates (https://www.statista.com/statistics/370694/mobile-broadband-internet-penetration-region/) (26 and 85.2 in Africa and Europe, respectively), it is clear that the Internet availability in this continent is still far behind the world's average figure. Because of this Africa's uniqueness, M. Masinde advocates for innovative African-centric approaches to IoT, rejecting the old-style approach of "transferring of Northern designs to Southern realities" [30, 31]. More in detail, by considering two very concrete use cases as a drought early warning system and an asset tracking system, he argues that, by innovatively exploiting local and unique realities such as indigenous knowledge, the chances of success for an IoT-based business in this continent are increased. The authors of [29] provide a comprehensive survey, country by country, of the undertaking of IoT in Africa, also isolating the major challenges that still hinder the wide adoption of the IoT in the continent.

With reference to the benefits of the IoT technologies for a sustainable development of rural African environments, the authors of [32] introduce some interesting applications in the agriculture domain (i.e., crop farming, weather forecasting, wildlife management, forestry, livestock farming, market identification, and rural financing), focusing on some very specific requirements of rural areas of South Africa and Zambia. This work is similar to our use cases in Senegal and Ghana; see Sections 5.1 and 5.2, respectively. A very inspiring work is described in [33], where the authors suggested the development and the implementation of the so-called "frugal IoT", as a paradigm to improve the monitoring and management of small-scale farms in Africa. In this case, the final goal is to minimize IoT-related hardware deployment, while maximizing the benefits of the adoption of such hardware, so as to bring African small farmers closer to the market and improve the prospects for food security across the continent. A more practical deployment of a standalone Wireless Sensor Network (WSN) for precision irrigation in Malawi is presented in [34]. More in detail, this deployment consists of a ZigBee-based WSN whose base station is connected to the Internet through a GPRS modem. Sensor nodes are powered by rechargeable batteries through solar panels, while they are provided with on-board sensors able to measure soil temperature and soil moisture, as well as irrigation valves able to automate the field watering process.

Finally, with reference to the benefits of the IoT technologies for a sustainable development of African metropoles, the authors of [35] identify several potential applications of IoT in transport that can make a difference to the South African economy. Our urban use case in Togo, see Section 5.3, also tackles urban development. In [36], the authors propose a smart end-to-end IoT architecture able to monitor and control electric water heaters which are responsible for consuming over 30% of household energy budget in South Africa. In [37], the authors apply smart city and smart mobility indicators to determine the level of smartness of the public transport infrastructure in South Africa and Gauteng province. Finally, authors of [38] discuss the current situation of public transportation in the city of Ouagadougou, Burkina Faso. Interestingly, they also provide and evaluate a preliminary distributed application based on users' personal phones, RFID tags mounted on buses and Short Message Service (SMS) of mobile cellular networks for communication, hence addressing the technological and social specificity of the sub-Saharan region.

3. IoT in Africa: Challenges

Any IoT deployment should tackle and solve a common set of technical challenges, such as end-to-end security, data privacy and trust, scalability, and interoperability [2]. However, African scenarios reveal a pool of unique challenges. Especially in remote and rural areas, the main challenges are the high cost of hardware, the complexity of deployment, and the lack of technological ecosystem and background [39].

Despite these difficulties, there are many opportunities for IoT in Africa. The following are but some of the most striking examples: monitoring of water, air, and soil quality, potable water management and optimization, warning systems for environmental threats (e.g., drought, tsunami, flood, earthquake, wildfire), wild animal tracking devices, and deforestation control systems [30, 40].

In this section, we detail seven challenges with IoT deployments in Africa, namely, (i) longer range for rural access, (ii) cost of hardware and services, (iii) dependency to proprietary infrastructures, (iv) local interaction models, (v) low-energy consumption, (vi) ease of deployment and operation, and (vii) resilience against environmental hazards.

3.1. Longer Range for Rural Access. Traditional mobile communication infrastructures (e.g., GSM/GPRS and 3G/4G) are still very expensive to use with IoT devices. Moreover, they are definitely not energy-efficient for autonomous devices that must run on battery for months. Short-range technologies such as IEEE 802.15.4 can eventually be used by implementing multi-hop routing to overcome the limited transmission range but this can only be envisaged with high node density and easy access to power scenarios such as smart-cities environments. They can hardly be considered in isolated or rural environments.

Recent Low-Power Wide Area Networks (LPWAN), Sigfox™ or Semtech's LoRa™ technology, provide a much more adapted connectivity answer for IoT in remote areas as a star topology with a central gateway or base station can be deployed [41]. Most of the long-range technologies can achieve 20km or higher range in LOS condition and about 2km in urban NLOS [42]. LoRa technology that can be privately deployed in a given area without any operator has a clear advantage in the context of developing countries over Sigfox whose coverage is entirely operator-managed.

3.2. Cost of Hardware and Services. Commercial IoT devices are definitely too expensive for very low-income countries. In addition, these highly integrated devices are difficult to repair, and their parts are hard to find locally. The availability of low-cost, open-source hardware platforms such as Arduino definitely pushes for a Do-It-Yourself (DIY) and "off-the-shelves" design approach: the Arduino Pro Mini based on an ATmega328 microcontroller has a high performance/price tradeoff and can be used to build a low-cost generic sensing IoT platform with LoRa long-range transmission capability for less than 10 euro. In addition, these boards also benefit from the support of a worldwide and active community of developers and a large variety of software libraries are available.

Commercial LPWAN gateways use advanced concentrator radio chips to listen on several channels and radio parameters simultaneously. The cost of such concentrator alone is more than a hundred euro. In the context of smaller scale rural applications, simpler "single-connection" gateways can be built using the same radio components than those for end-devices. Again, by adopting "off-the-shelves" embedded Linux platforms (e.g., a Raspberry Pi board) the cost of an LPWAN gateway can be maintained as low as 50 USD.

3.3. Dependency on Proprietary Infrastructures. Along with the worldwide IoT uptake, a large variety of IoT Clouds platforms offers an unprecedented level of diversity which contributes to limit dependency on proprietary infrastructures. Most of these dedicated IoT platforms have free account offers that, despite some limiting features, can largely satisfy the needs of most agriculture/micro and small farm/village business models. In order to take advantage of all these infrastructures, the design of an IoT versatile gateway should highly decouple the low-level gateway functionalities from the high-level data postprocessing features to maximize the customization of the data management part. Furthermore, by privileging high-level scripting languages such as Python, the customization process can be done in a few minutes, using standard REST API interfaces to IoT Clouds. Therefore, rather than focusing on large-scale deployment scenarios, easy integration of low-cost off-the-shelves components with simple, open programming libraries should be the main focus of IoT platforms in developing countries.

3.4. Local Interaction Models. With unstable and expensive accesses to the Internet, data received on the gateway should be locally stored. In addition, a versatile gateway is also an interesting feature where it should be possible to turn the gateway into an end computer by just attaching a keyboard and a display, using visualizing data locally. With standard wireless technologies such as WiFi or Bluetooth, it is also

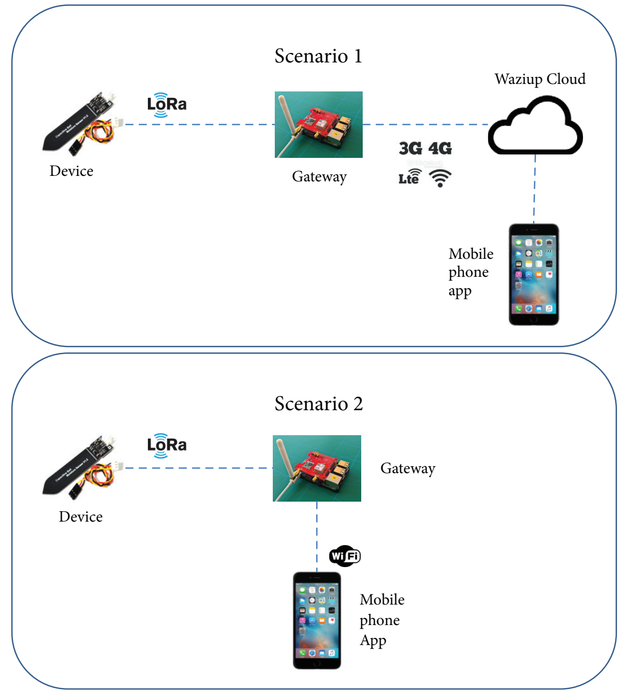


Figure 2: Deployment scenarios in developing countries.

interesting to provide local interaction with the end-user' smartphone/tablet to display captured data and notify users of important events without the need of Internet access. Figure 2 shows the two interaction models.

In the first case, the access to the Internet is provided to the gateway through technologies such as 3G/4G or DSL+WiFi. The second case shows a fully autonomous gateway scenario, without the Internet access. Data coming from sensing devices are collected and stored locally on the gateway. The data can then be accessed locally using smartphones or tablets connected to the gateway through a local WiFi or Bluetooth connection.

3.5. Low-Energy Consumption. The deployed IoT device must be able to run for at least 1 year using regular, easy-to-find batteries. Regular AA batteries can be purchased worldwide, even in remote villages. Our experiments showed that an IoT application running on a barely simple sensor node composed by an Arduino Mini board equipped with temperature and humidity sensors and a LoRa communication transceiver can last at least two years with a couple of standard AA batteries. As detailed in in Section 4, several techniques can be used to reduce the energy consumption, such as lowering the sampling and communication period.

3.6. Ease of Deployment and Operation. The system must be able to work out-of-the-box, or with minimum configuration effort. Important configuration steps, such as system upgrades, must be made available from a user-friendly interface accessible, for instance, through a low-cost smartphone. Remote maintenance is also a crucial point. For example in our rural testbeds, technicians have to drive for several hours to reach the deployment sites. Furthermore, in the case of smart agriculture, the sensors need to be redeployed on regular basis. This means that the devices are removed from the field by farmers before the harvesting and replaced at the end of the harvesting. The device deployment is, for this reason, delegated to low-qualified people.

3.7. Resilience against Natural Environmental Hazards. Durability and maintenance aspects are clearly a challenge for the DIY IoT strategy. The boards used can be fragile, and the casing in which they will be deployed must be studied with care. For instance, in the case of the DIY buoy developed and engineered for the fish ponds trial, the temperature and humidity in the box were very high, due to the sun and the proximity of water. The internal sensor installed inside the box showed that the temperature was reaching 50 Celsius degrees every day, while the relative humidity measured was more than 90% during the day. This resulted in damage to the battery and the board. The LoRa transmissions also suffer from obstacles. In our agriculture trial, once the crop grew up, this shadowed the antennas, resulting in frequent transmission losses.

4. The Proposed Platform

In this section, we describe the solution stack that has been developed to meet the unique requirements of Africa's use cases. It comprises a hardware platform, which is low-cost, low-energy, and long-range. It also includes a Cloud data and application platform, able to offer unique services specially tailored to fit Africa's needs.

4.1. IoT Sensing and Actuating Devices. Considering that the proposed platform is mainly employed in rural environments of developing countries, we adopted a Do-It-Yourself (DIY) approach to the IoT device assembly, which inherits the Arduino philosophy of low-cost, simple to program, yet efficient, hardware platforms. Indeed, the Arduino-compatible ecosystem is extremely wide and heterogeneous, proposing several solutions, from very powerful boards (mainly used for development purposes) to the tiniest and power-aware boards (more suitable for permanent setups). For instance, we used the Arduino small form factor board named "Pro Mini" as our generic tiny sensing IoT device. Such board comes with the minimum of components to keep the cost down (e.g., no on-board USB or pin headers, while still powered by an ATmega328P microprocessor running at 8MHz). Its unique features are the price (it is available for less than 2 USD, if bought directly from China suppliers), the size (18x33mm, less than 2 grams), and the low power it needs to properly work (it is a low-voltage board working at 3.3V, which guarantees very low-power consumption, especially when in deep-sleep mode (when in this state, the board registers $5\mu A$ of current draw.)). Thus, with efficient power management policies (e.g., sampling and reporting with a period of 1 hour) such a device is able to run in autonomy for years, with a couple of standard AA batteries. At the same time, to address the requirements of more demanding IoT applications, we also used boards belonging to the Teensy family (e.g., LC, 3.1, and 3.2). Such family offers state-of-the-art microcontrollers with more memory and advanced power management features with respect to the previous board, yet at a reasonable cost (e.g., about 10 USD for the LC model; indeed LC stands for "Low Cost").

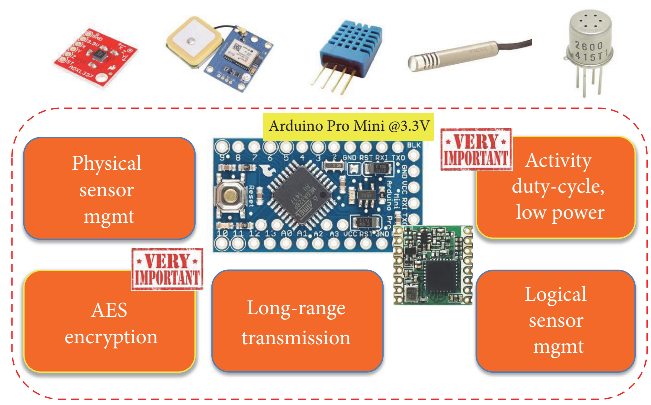


Figure 3: The proposed generic sensing and actuating IoT device, together with the main software building blocks.

Thus, depending on the sensing task, our generic IoT device is powered by one of such boards, also integrating software building blocks into ready-to-use templates for quick and easy customization (see Figure 3).

After the engineering phase, a low-cost hardware production approach is carried forward through a simple and open-source PCB design, which is then checked out as a 1-click order. Figure 4 depicts the whole production process of a sensing device equipped with an on-board integrated LoRa chipset and antenna that, besides providing seamless integration at lower costs, is necessary, for instance, when producing the cattle rustling collar, for obvious reasons of robustness.

Finally, Figure 5 depicts some of the already produced sensing devices, currently deployed in various testbeds in Africa (e.g., cattle rustling in Senegal, fish farming in Ghana) and Asia (e.g., multi-level soil moisture monitoring station for precision agriculture in Pakistan) and connected to the proposed platform.

4.2. Versatile Gateway. The engineered gateway is a single-channel LoRa device that acts as a gateway by forwarding LoRa packets to the network. Such device can also receive (and decrypt, if necessary) LoRaWAN packets on a dedicated channel. As shown Figure 6, the gateway architecture sharply decouples low-level transport functionalities from the high-level data postprocessing features, privileging high-level languages at Cloud side. The versatile gateway is based on a Raspberry Pi board (all models supported), allowing deploying a very affordable, yet highly efficient, gateway proposing many features that are lacking in commercial gateways. Several versions of the gateway can be assembled with the same software distribution, depending on the budget and the deployment requirements and constraints, as illustrated in Figure 7. For instance, on the right-hand side of Figure 7 we show a tiny version of our gateway, based on a Raspberry

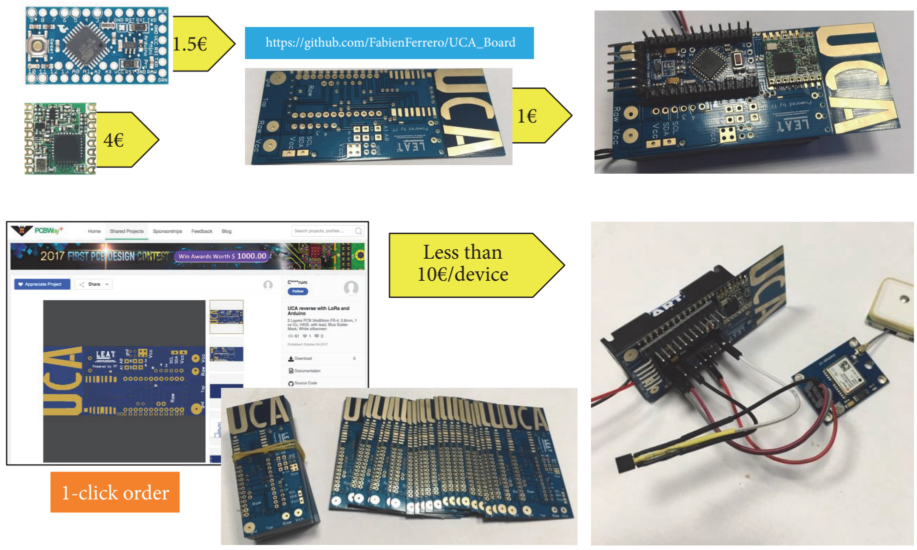


FIGURE 4: Production process of an IoT sensing device comprising also a LoRa chipset and an integrated antenna.

FIGURE 5: Several devices deployed in various testbeds and connected to the proposed platform.

Pi Zero board, coupled with a Loranga board (Loranga is a board officially supported by the WAZIUP European project and launched its own crowdfunding campaign on the Kickstarter platform.). This gateway, being equipped with a 2G/3G modem and a LoRa radio transceiver, is particularly well-suited for deployments in remote areas.

The gateway is able to handle offline scenarios or, more generally, situations in which the Internet connectivity is intermittent or not reliable: SMS can be sent if a cellular 2G dongle is attached, while data sent by IoT devices can be locally stored in a NoSQL MongoDB database. Eventually, being provided with an embedded web server, the gateway can also provide locally stored data that can be graphically displayed on users' smartphones and tablets, through a wireless LAN connection. Ultimately, the gateway can also be used as a general purpose PC, by just connecting a keyboard and a display. All of these interactions mechanisms can be selectively enabled as needed.

Customizing data management tasks can be done in few minutes, using standard tools, simple REST API interfaces

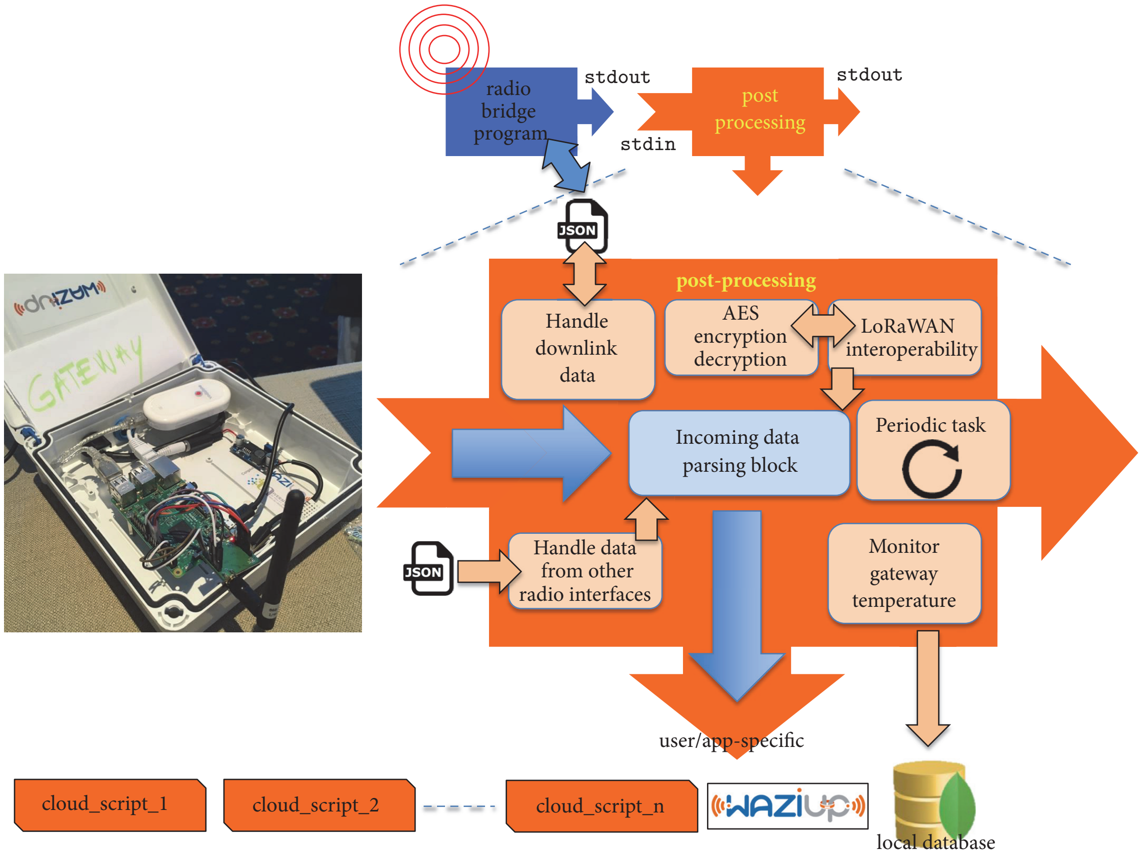


Figure 6: Modular and versatile low-cost gateway instances.

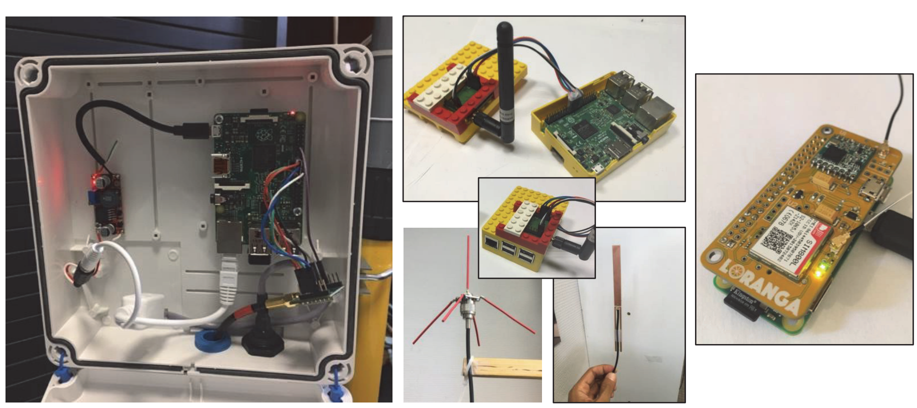


Figure 7: Low-cost DIY versatile LoRa gateway.

and available public Cloud platforms and services. We provide ready-to-use templates that already supports a number of publicly available IoT platform such as Node-Red, Firebase, ThingSpeak, Freeboard, GroveStream, and FIWARE as well as commonly used protocols like the MQTT. For instance, as shown in Figure 8, the gateway can be been seen as a Node-Red node to be integrated into more complex flows developed in the Node-Red graphical programming framework.

4.3. Cloud Platform. In order to provide a whole software stack, we also developed our own Cloud platform. It essentially collects, stores, presents, and analyzes data coming from the gateways. Moreover, it allows users to code and deploy ad hoc applications, as well as to manage their own data and assets. In order to be suitable for the African market, the Cloud platform has some key design objectives:

(i) Easy data processing, storing and visualization

(ii) Support for SMS, USSD, voice, mobile apps

(iii) Support for intermittent Internet connection

(iv) User application hosting

(v) High data security

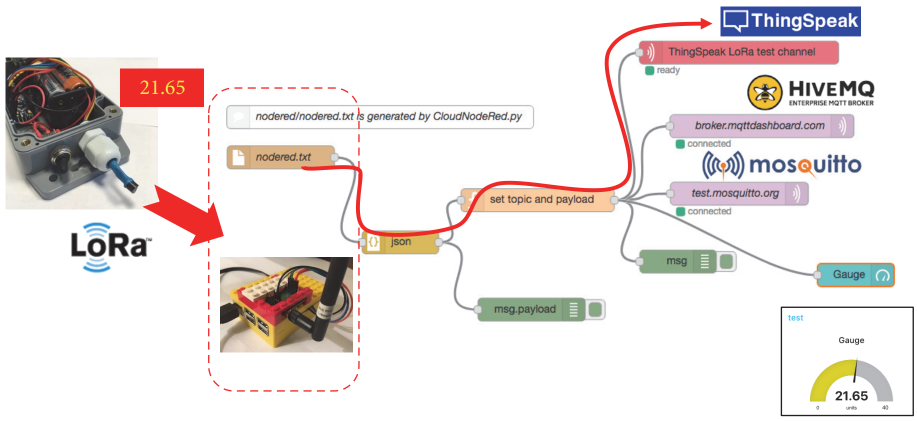
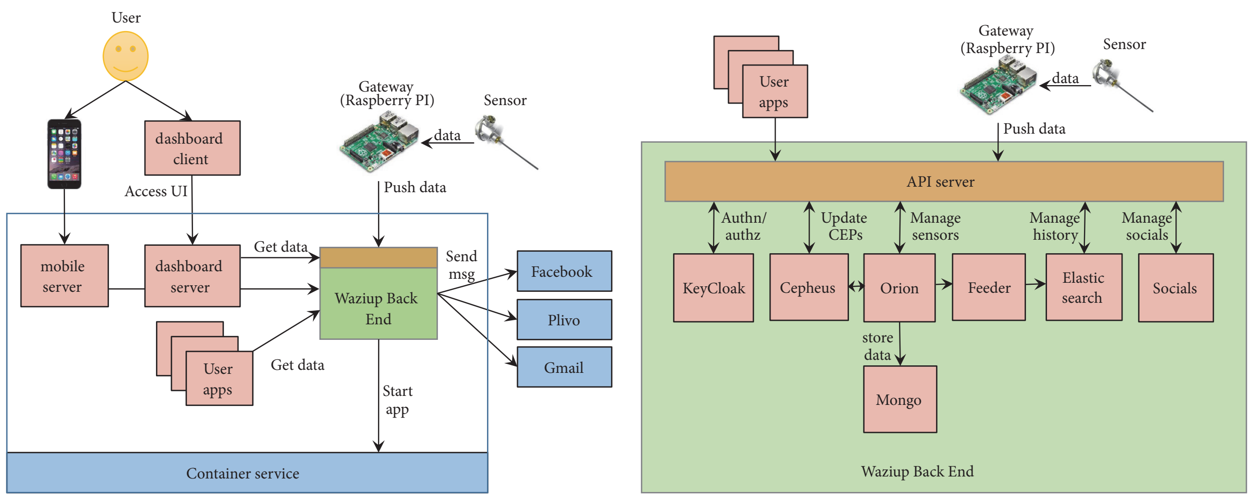


Figure 8: The proposed gateway in a Node-Red programming flow.



(a) (b)

Figure 9: The proposed Cloud platform in terms of (a) block-diagram architecture and (b) backend and main software components.

As depicted in Figure 9(a), users interact with the Cloud platform through a web interface, namely, the dashboard. Moreover, we provide a mobile application since, as mentioned earlier, in Africa mobile phones, represent the primary means to get access to the Internet. Those two interfaces allow users to manage the IoT assets (i.e., gateways and sensing devices). The platform also visualizes data coming from the sensing devices and let users analyze real-time and historical data coming from the gateways (the only kind of asset able to push data directly to the Cloud platform). The heart of the Cloud platform is the *backend*, as it implements and provides all the services necessary to manage the incoming data. Furthermore, it provides access to the social media and communication channels.

In rural zones, GSM phones and coverage are extremely widespread. However, 3G/4G coverage is often not available. For this reason, users adopt SMS and voice calls to communicate. This drove the technology choice for our user interfaces: the platform supports SMS, USSD, automated voice messages, Facebook, and Twitter. SMS and automated voice messages are a must-have in rural Africa zones. USSD is also widely used because it allows a better interaction that SMS. Facebook is the primary social network used in Africa (http://www.internetworldstats.com/stats1.htm), with 170 million users.

Another constraint is the Internet connection: the Internet link can be very intermittent. However, the vast majority of current IoT platforms are very Cloud-dependent, as the endpoint of all data sources is the Cloud. In this situation, if the Internet connection is not available, then users simply cannot access their Cloud-based applications. For this reason, we developed what we call the "Local Cloud" concept: a lightweight version of the platform is able to run locally on a single PC, or even on the IoT gateway itself. Thus, when the Internet connection is not available, the data is stored locally, until the remote connection is reestablished and the data can be uploaded to the Cloud side of the platform. Furthermore, the local Cloud allows the user to access all his applications locally.

We also noticed that application hosting is a problem for the majority of African IoT developers. Indeed, hosting a single IoT application in the Cloud can be for them too expensive. For instance, a single "m4.large" virtual machine instance hosted at Amazon Web Services (AWS) EC2 costs around 80 USD/month, which is often not affordable for a single person entrepreneur, given the average African salaries. Furthermore, it requires a lot of operational competencies and has a complex maintenance. For this reason, we proposed a simple application hosting service based on the Docker container technology (https://www.docker.com/). Then, each user's application is embedded inside a dedicated Docker container, which is pushed directly to the Cloud platform and hosted there.

Finally, data security is a must for any IoT platform. We implemented both authentication and authorization security schemes. Before accessing the Cloud platform, the user needs to authenticate himself. For this, he needs to connect to the dashboard to create an account, as a user-name/password pair. Then, all transport packets are encrypted using the HTTPS protocol. Once logged in, the user can access a variety of resources, based on his roles and attributes. For instance, by default, he has access to the assets that he physically owns (being them gateways or sensing devices). The social media access is also configurable, so as to allow only certain users to access nonfree channels like SMS.

Figure 9(b) shows the implementation of the Cloud backend. All interactions are defined and supported by a purposely developed Application Protocol Interface (API) which defines eight endpoints, namely,

(i) *sensors*: create, read, update and delete sensing devices;

(ii) *data*: perform complex queries on stored data;

(iii) *notifications*: send messages and alerts to users based on sensor values;

(iv) *social*: send direct messages on social network channels like Twitter or Facebook;

(v) *users*: manage users of a particular application, as well as their access rights;

(vi) *entities*: create additional assets to be managed, such as farms, fields, etc.;

(vii) *events*: define complex relationships between entities (e.g., to define a complex event processing engine, CEP);

(viii) *apps*: host containerized applications in the Cloud platform.

All those endpoints are protected using the Keycloak (http://www.keycloak.org/) open-source identity and access management technology. To manage data in a more abstract context, the platform integrated the FIWARE Orion (https://fiware-orion.readthedocs.io/en/master/) context broker. In this way, our platform can manage both simple data (i.e., a measurement value of 25) and more complex contexts (i.e., measurement units, geo-referenced positions, etc.). Data persistence is ensured by a MongoDB NoSQL DBMS, while such data is also sent to an ElasticSearch (https://www.elastic.co) instance, where it can be analyzed and visualized with powerful graphical tools, such as Kibana (https://www.elastic.co/products/kibana). FIWARE Cepheus (https://github.com/Orange-OpenSource/fiware-cepheus) is used to provide CEP functionalities. Finally, a specific component called "Socials" was internally developed to build and deliver notifications to the users through various social networks.

5. Use Cases

In this section, we present 3 use cases: water saving in tomato crops in Senegal, water monitoring in fish ponds in Ghana, and finally waste management in Togo.

5.1. Use Case 1: Water Saving in Tomato Crops in Senegal. While sub-Saharan Africa has ample opportunities to become one of the breadbaskets of the world, it has experienced the most severe land degradation in the world [43]. Soil and fertilizer management is one of the biggest challenges sub-Saharan farmers are facing seasonally. There is a need to clarify the impacts of fertilizing products on the soil. In this section, we present an experiment aiming at analyzing the capacities of natural, nonchemical fertilizers. In particular, we will study the effects of fertilizers on the soil moisture retention mechanisms.

The objectives of the experiment are twofold. The first goal is to highlight the water retention mechanism depending on the type of treatment. We analyze four types of treatments: poultry manure, cow dung, mineral fertilizer, and a 50/50 mix of poultry manure and cow dung. The second objective is to highlight the impact of a specific treatment on the soil quality and the growth of the crop.

5.1.1. Overview. For the deployment, we have an area of half a hectare, in which two varieties of tomatoes crops are sown, namely, "Kiara" and "Mongol", as shown in Figure 10. The area is partitioned into several equally dimensioned patches, such that each patch holds an experiment. A patch represents the elementary space portion where data are collected by sensors and reported to a central gateway. In each column of the patch, one type of crop is sown, either Kiara or Mongol. The color of the patch represents the type of fertilizer that will be used.

About water management, the deployment site is irrigated by means of a dripping system. The proportion of water dropping in each sensor's patch is checked to be the same. In this way, it is possible to measure parameters on the same basis.

5.1.2. Deployment. We deployed five IoT sensor nodes in the field, marked with codes S8 to S34 in Figure 10. Each node incorporates a small set of sensors able to collect parameters such as soil moisture, ambient temperature, and relative humidity. The goal of such a deployment is to be able to capture and analyze soil and environmental data involved in the development of the crop.

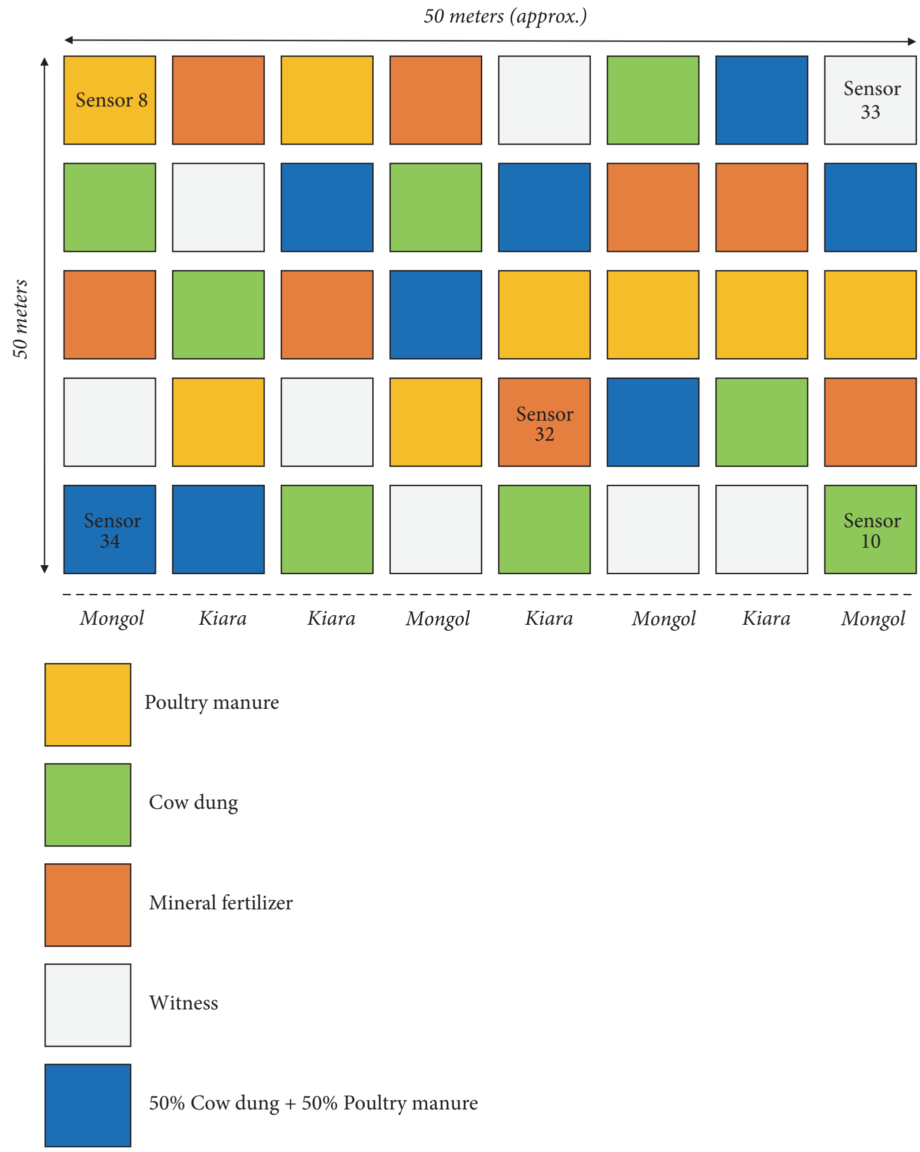


Figure 10: Deployment map of the water use case experiment.

IoT nodes are connected to a local gateway, near the field, to which they forward gathered data. Data are captured from each node at regular intervals of two hours and sent to the gateway via LoRa communication (see Figure 11).

Once sent to the gateway, data can be pushed to the Cloud platform using the APIs described in Section 4. The uploading process is achieved by first posting datapoints to a data broker and storing them in a flexible database following the required format. Then, historical APIs are used to extract data for future analysis, as well as plotting them on visualization tools. Examples of collected and visualized data are depicted in Figure 12: we can see the data captured by the sensors and rendered in the dashboard of the Cloud platform as two-dimensional plots. The next step will consist in acquiring enough data to assess the water retention in the ground, based on each fertilizer.

5.1.3. Results and Environmental Impact. The preliminary result of the experiment showed that poultry manure is the best candidate for water retention: the humidity measured stays high for a longer period of time. However, further experiments are required for this result to be statistically significant. We also identified the correlation between air temperature and soil humidity. Improvements are necessary for device calibration, deployment condition, and gateway stability for collecting data. Phase two of our deployment will take into account those results and carry out more sensors with more parameters.

5.2. Use Case 2: Aquaculture Smart Monitoring in Ghana. In the last decades, aquaculture has grown exponentially in the world, especially in developing countries. For instance, Asian developing countries now produce 89% of the total fish production worldwide [44]. In Africa, fish production

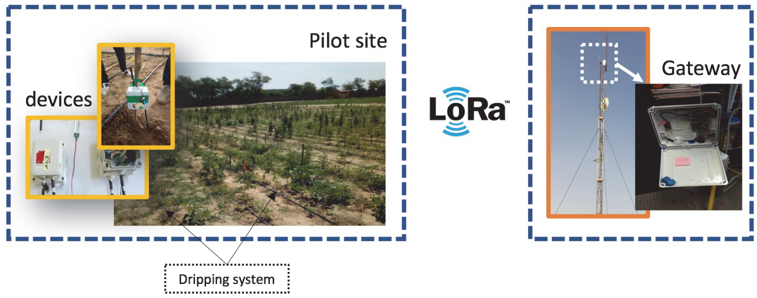


Figure 11: Deployed devices for the experiment.

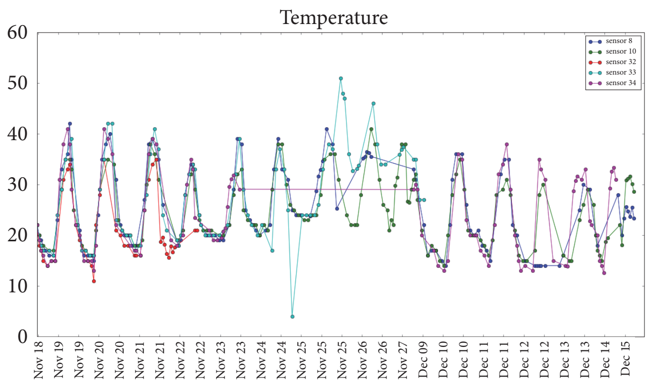


Figure 12: Visualization of the data from sensor nodes.

from farming has reached 2.5% of the global fish production, thanks to many fish farming development projects in the last ten years. However, this is not sufficient as many countries still have to import fish. In Ghana, for example, the combination of the annual supply of fish from capture and from aquaculture does not meet half of the country demand. This shortage in supply is expected to increase further with limited prospects for increasing marine and inland fisheries. Aquaculture has thus become pivotal in the future development but many farmers lack information to properly run a fish farm and to get a good production. This leads the farmers to make decisions based on instinct or empirical methods and may have a bad environmental impact. Indeed, they can use too many products in the water or consume too much water by changing it at the wrong time. The key is to monitor the water continuously because the optimum fish production is totally dependent on the physical, chemical, and biological qualities of the water. Unfortunately, classical water monitoring devices are too expensive for African farmers, as they can cost up to 10,000 Euros (https://in-situ.com/products/aquaculture-management/fish-pond-management/fish-pond-management/aquaculture-pond-buoy/).

We deployed a low-cost water monitoring system in a fish pond in Ghana, as can be seen in Figure 13. The system is measuring pH, temperature, and dissolved oxygen level in water and is autonomous in energy with a solar panel. It is pushing data in real-time so the farmers can see at any time the status of the water in the pond.

5.2.1. Deployment. As shown in Figure 14, we deployed the low-cost water monitoring device in Kumah Farms, in Kumasi, Ghana, in January 2017. This farm has 18 ponds of various size, from 120 square meters to 0.8 hectares. They grow Tilapia and Catfish, which has a life cycle of 6-8 months before being harvested. After designing the prototype and assembling the electronic parts in France, the prototype was finally assembled on site with help of African local partners and potential final users. A low-cost gateway was installed at the entry of the farm with a 3G USB dongle. The distance from the pond to the gateway is about 200m with many obstructions (e.g., vegetation and other buildings). Data was pushed to ElasticSearch servers and visualized with Kibana in the Cloud platform frontend.

5.2.2. Data Analysis. In Figure 15, we can see the variations of the values of pH, dissolved oxygen, and temperature in early February 2017. The first noticeable feature is that pH variation is cyclical within a day. pH falls during the night and reaches its minimum in the morning. Then it rises during the day to reach its maximum at the end of the afternoon. It is normal for pond water to have this kind of behavior. However, the variation of pH per day is up to 2.58. This is way too much as it is recommended that pH does not vary more than 0.5 within a day. With this kind of variation, it can make the fishes shocked, weakened, and stop eating. We can also notice that pH is getting too high every afternoon, above the warning level of 8.5 (in orange) and even above the critical level of 9 (in red). The strong fluctuation of pH during the day combined with high level of pH in the afternoon is symptomatic of a pond with too many algae. Algae and microorganisms use CO2 and can affect the pH of the water. Algae grow and develop quickly when hardness (the amount of CaCO3) of

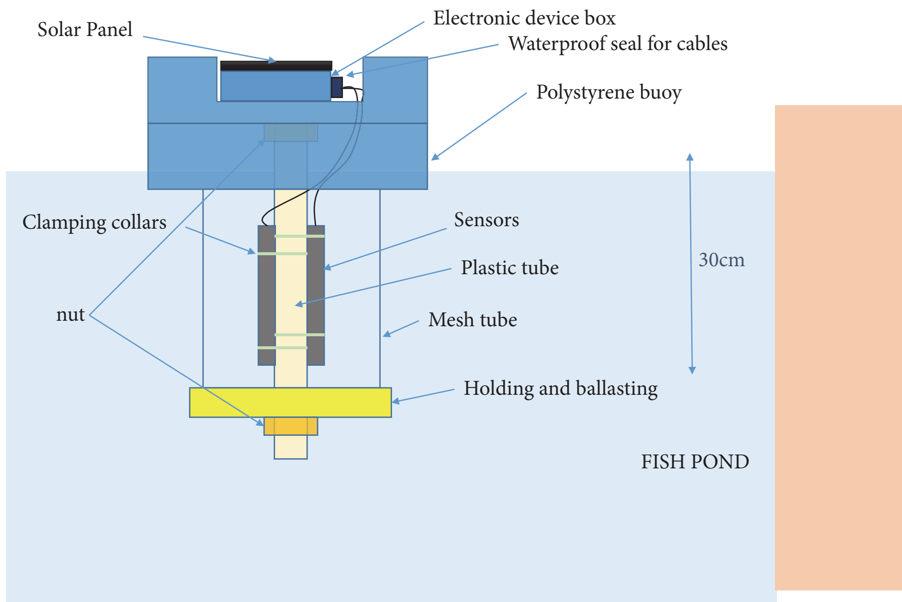


FIGURE 13: Buoy prototype diagram.

FIGURE 14: Buoy prototype deployment.

the water is low. The recommendation in that case [6] is to add dolomite lime (100-200 kg/ha) to increase water hardness and a buffering agent. Water should also be changed to stabilize the growth of algae.

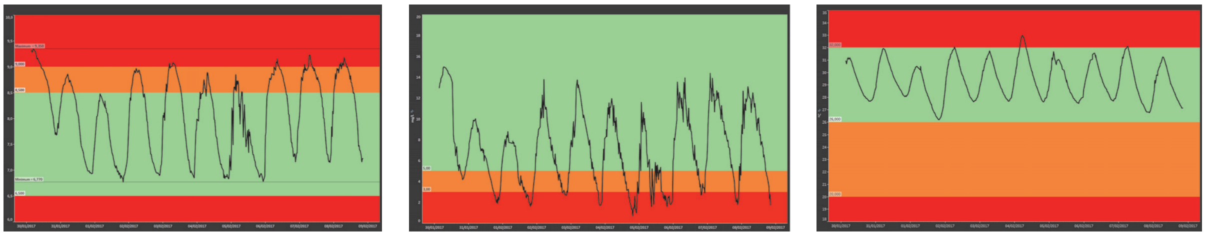
The level of dissolved oxygen is one of the most important factors in aquaculture, as a lack of oxygen can cause fishes to die. In the figure, we can see the cyclical behavior of dissolved oxygen level in the pond. This is an expected behavior: during the day, algae and microorganisms produce oxygen through photosynthesis. During the night, as oxygen is not produced anymore, it is consumed by fish and rapidly decreases until the sun is high enough to resume photosynthesis. In this pond, we can see that the amount of oxygen produced from photosynthesis during the day is not enough to last all night long. Every morning, fish lacks oxygen. This can cause them to grow slowly or even die if the stress lasts for too long. The only way to deal with too low oxygen levels in a pond is to aerate the water with manual aerators that brass water and increase the oxygen level. In this pond configuration, aeration must be done at night to avoid the lack of oxygen in the morning.

Finally, the water temperature is also cyclical with a natural increase in day and decrease at night. In this pond, the temperature stays most of the time in the recommended range.

5.2.3. Results and Environmental Impact. With this first deployment in Kumah Farms, we pointed out two major issues in the fish pond:

(i) The oxygen level is too low in the morning.

(ii) The pH is too high and varies too much, because of excessive growth of macroalgae.



(a) Water pH (b) Dissolved oxygen (c) Water temperature

Figure 15: pH, dissolved oxygen, and water temperature in the fish pond.

Both cause fishes to be stressed. In that case, fishes grow slower, they do not breed, and they can eventually die. Our recommendations to improve water quality (i.e., aerate water at night and remove algae with dolomite lime) should considerably improve the fish production yield of this farm. Moreover, we consider that the recommendations we gave will lead to smart decisions and actions, based on the real status of the fish pond. This will be the first step for a smart fish farm management with low environmental impact.

5.3. Use Case 3: Urban Waste Management in Togo. African cities have the fastest urbanization speed in the world. For instance, Kinshasa will have tripled its population by 2050. The most important challenges in African cities are the household living conditions, improvement of food security, and digitalization of the multiple sectors like waste management. In African cities, most of the informal economy relies on the house (i.e., for snacks everything is prepared for home) or the proximity of the street (i.e., access to shops and workshops front). Furthermore, we see a lack of awareness of resource management and a lack of good citizen reflexes. We see those problems as an opportunity for smart devices to compensate. Finally, the general African context and its sociology are favorable to distributed solutions such as smart grids, peer-to-peer networks, or blockchains. IoT is a key enabler in this context.

The development of the urban lifestyle in African cities is producing more waste every day. This waste accumulation is a major challenge for African cities and a huge source of pollution. This is a high time for having a proper plan for waste management and disposal system. In order to be economically sustainable, a waste collection and recycling infrastructure must also offer business opportunities for all stakeholders in the recycling chain. With this objective in mind, in this section, we discuss an automated system for waste management by using IoT technology.

5.3.1. Overview. The solution proposed is a smart waste bin that is able to send a notification when it is full. It is coupled with a network of waste collection vehicles. In contrast to the other systems worldwide, there are no fixed routes and time schedules. Instead, we follow a demand-driven approach with smaller transport vehicles. Furthermore, the solution needs to be of low cost in order to be deployed in underdeveloped countries. It includes energy-efficient devices and low rate communications.

Figure 16: Waste bin prototype.

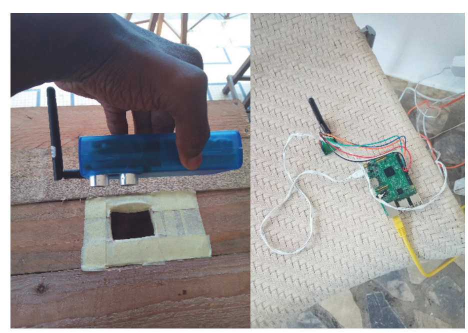


Figure 17: Waste bin sensor node.

5.3.2. Prototype. The smart bin is shown in Figure 16. The prototype is built and distributed by the local start-up SCoPE (https://docs.google.com/presentation/d/18o_ZZ6q089w62sy-I7W3soPxyrKsQ_m1BGM9sMj7Pz8M). It is entirely made from recycled material. The bin liner is also created from recycled drinking plastic bags, which are very common in this region.

The sensor node developed is shown in Figure 17. It consists of one Arduino Pro Mini, one LoRa radio module, and an antenna. It is powered by one 9V rechargeable battery. As detailed in Section 4, this kind of setting can have an autonomy of more than one or two years. The cost of the sensor node is around 30 Euros.

The gateway is composed of a Raspberry Pi board, with a LoRa radio module and an antenna. It runs the software stack presented in Section 4.

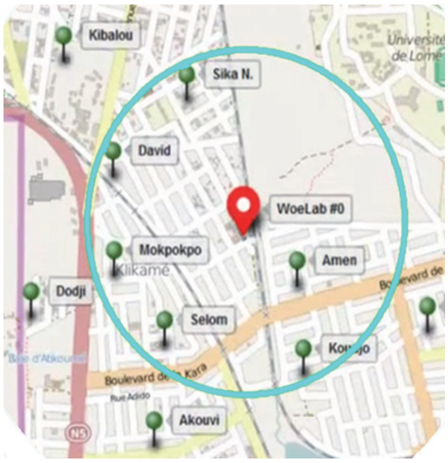


Figure 18: Bins localization.

5.3.3. Deployment. Figure 18 shows the localization of some of the bins in the city of Lomé, in the "Abové" neighborhood. The deployment has two phases. In the first phase, a low-tech bin has been deployed, with no intelligence at the bin. The users must send a signal themselves by phone when they see that the plastic waste bin is full. A simple smartphone application has been developed specially for this purpose. So far, 47 users have been equipped with this model of the smart bin.

In the second phase, we are modernizing the bin, implementing a sensor node in each bin. This allows sending a notification to the platform automatically when the bin is full. We started the pilot phase with 3 users equipped.

The LoRa gateway has been deployed in the premise of the WoeLab in Togo. Due to the urban nature of this testbed, the range attainable by the LoRa network is reduced. Our tests show that the range needs to be limited to 1Km around the WoeLab, which allows us to reach 22 users.

6. Conclusions

In recent years, we witnessed a very fast economic growth in Western African countries. Unfortunately, this sharp development has also direct consequences in terms of environmental impact, both in rural and in metropolitan sites. In this respect, the Internet of Things is asked to alleviate the effects of these threats, hence realizing the concept of a *green* and more sustainable development. However, deploying an effective IoT platform in Western Africa countries comes with its own unique challenges. Among them, the most important is the *tolerance* to the effects of using low-cost, low-bandwidth, and intermittently connected-to-Internet devices. Moreover, widely accessible communication means, such as SMS and voice calls, need to be accommodated as services. In this paper, we proposed an open IoT platform that takes into account not only the environmental component but also the local economic and social aspects, so as to provide a truly sustainable solution for this region. We deployed our solution in three testbeds, dealing with soil monitoring in Senegal, water quality in fish ponds in Ghana, and waste management and collection in an urban area in Togo. In the first use case, we aimed at reducing the amount of water used with respect to the fertilizer type. In the second use case, we monitored key parameters such as dissolved oxygen and pH, which allow us to give recommendations. We optimized the fish growth and reduced wastes in those fisheries. Finally in Togo we implemented a network of smart bins, in order to reduce the waste thrown in the environment and increase recycling.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Additional Points

The source code of the software platform described in this study is publicly available at the following repository: https://github.com/Waziup/Platform.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement no. 687607.

References

[1] K. Ashton, "That 'internet of things' thing," *RFID Journal*, vol. 22, pp. 97–114, 2009.

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: a survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[3] M. Etelapera, M. Vecchio, and R. Giaffreda, "Improving energy efficiency in IoT with re-configurable virtual objects," in *Proceedings of the 2014 IEEE World Forum on Internet of Things, WF-IoT 2014*, pp. 520–525, Republic of Korea, March 2014.

[4] M. Vecchio, R. Giaffreda, and F. Marcelloni, "Adaptive lossless entropy compressors for tiny iot devices," *IEEE Transactions on Wireless Communications*, vol. 13, no. 2, pp. 1088–1100, 2014.

[5] F. K. Shaikh, S. Zeadally, and E. Exposito, "Enabling technologies for green internet of things," *IEEE Systems Journal*, vol. 11, no. 2, pp. 983–994, 2017.

[6] P. J. Sarkar and S. Chanagala, "A survey on IoT based digital agriculture monitoring system and their impact on optimal utilization of resources," *Journal of Electronics and Communication Engineering (IOSR-JECE)*, 2016.

[7] "Harnessing the Internet of Things for Global Development," https://www.itu.int/en/action/broadband/Documents/Harnessing-IoT-Global-Development.pdf, 2015.

[8] S. Parkin, F. Sommer, and S. Uren, "Sustainable development: understanding the concept and practical challenge," *Proceedings of the Institution of Civil Engineers - Engineering Sustainability*, vol. 156, no. 1, pp. 19–26, 2003.

[9] A. Rahim, "IoT and Data Analytics for Developing Countries from Research to Business Transformation," in *Economics of Grids, Clouds, Systems, and Services*, C. Pham, J. Altmann, and J. Banares, Eds., vol. 10537 of *Lecture Notes in Computer Science*, pp. 281–284, Springer International Publishing, 2017.

[10] African Development Bank and OECD and United Nations Development Programme, African Economic Outlook 2017: Entrepreneurship and Industrialisation. AfricanEconomicOutlook.org, 20.

[11] H. G. Miller and P. Mork, "From data to decisions: A value chain for big data," *IT Professional*, vol. 15, no. 1, pp. 57–59, 2013.

[12] S. Murugesan, "Harnessing green IT: Principles and practices," *IT Professional*, vol. 10, no. 1, pp. 24–33, 2008.

[13] G. Iazeolla and A. Pieroni, "Energy saving in data processing and communication systems," *The Scientific World Journal*, vol. 2014, Article ID 452863, 2014.

[14] C. Dupont, T. Schulze, G. Giuliani, A. Somov, and F. Hermenier, "An energy aware framework for virtual machine placement in cloud federated data centres," in *Proceedings of the the 3rd International Conference*, pp. 1–10, Madrid, Spain, May 2012.

[15] N. Xiong, A. Vandenberg, and W. Han, "Green cloud computing schemes based on networks: a survey," *IET Communications*, vol. 6, no. 18, pp. 3294–3300, 2012.

[16] N. Ahmed Malik and M. Ur-Rehman, "Green Communications: Techniques and Challenges," *EAI Endorsed Transactions on Energy Web*, vol. 4, no. 14, p. 153162, 2017.

[17] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, "Energy efficiency in the future internet: a survey of existing approaches and trends in energy-aware fixed network infrastructures," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 223–244, 2011.

[18] P. Gandotra and R. K. Jha, "A survey on green communication and security challenges in 5G wireless communication networks," *Journal of Network and Computer Applications*, vol. 96, pp. 39–61, 2017.

[19] C. Zhu, V. C. M. Leung, L. Shu, and E. C.-H. Ngai, "Green Internet of Things for Smart World," *IEEE Access*, vol. 3, pp. 2151–2162, 2015.

[20] Population Division, "World Population Prospects: the 2017 Revision," United Nations, Tech. Rep, 2017.

[21] B. A. Aubert, A. Schroeder, and J. Grimaudo, "IT as enabler of sustainable farming: An empirical analysis of farmers' adoption decision of precision agriculture technology," *Decision Support Systems*, vol. 54, no. 1, pp. 510–520, 2012.

[22] P. P. Jayaraman, A. Yavari, D. Georgakopoulos, A. Morshed, and A. Zaslavsky, "Internet of things platform for smart farming: Experiences and lessons learnt," *Sensors*, vol. 16, no. 11, 2016.

[23] P. P. Jayaraman, D. Palmer, A. Zaslavsky, and D. Georgakopoulos, "Do-it-Yourself Digital Agriculture applications with semantically enhanced IoT platform," in *Proceedings of the 10th IEEE International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP '15)*, pp. 1–6, Singapore, April 2015.

[24] A. Ilapakurti and C. Vuppalapati, "Building an IoT framework for connected dairy," in *Proceedings of the 1st IEEE International Conference on Big Data Computing Service and Applications, BigDataService 2015*, pp. 275–285, USA, April 2015.

[25] D. Liu, X. Cao, C. Huang, and L. Ji, "Intelligent agriculture greenhouse environment monitoring system based on IOT technology," in *Proceedings of the International Conference on Intelligent Transportation, Big Data and Smart City, ICITBS 2015*, pp. 487–490, Viet Nam, December 2015.

[26] Z. Nakutis, V. Deksnys, I. Jaruevicius et al., "Remote Agriculture Automation Using Wireless Link and IoT Gateway Infrastructure," in *Proceedings of the 26th International Workshop on Database and Expert Systems Applications, DEXA 2015*, pp. 99–103, Spain, September 2015.

[27] R. Ranjan, P. P. Jayaraman, M. Villari, and D. Georgakopoulos, "Advances in Orchestrating Sustainable Smart Cities (Part 1)," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 4, pp. 317–319, 2017.

[28] A. Pieroni, N. Scarpato, L. Di Nunzio, F. Fallucchi, and M. Raso, "Smarter City: Smart Energy Grid based on Blockchain Technology," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 1, pp. 298–306, 2018.

[29] N. Onyalo, H. Kandie, and J. Njuki, "The Internet of Things, progress report for Africa: A survey," *International Journal of Computer Science and Software Engineering*, 2015.

[30] M. Masinde, "IoT applications that work for the African continent: Innovation or adoption?" in *Proceedings of the 12th IEEE International Conference on Industrial Informatics, INDIN 2014*, pp. 633–638, Brazil, July 2014.

[31] R. Heeks, "I-development not e-development: Special issue on ICTs and development," *Journal of International Development*, vol. 14, no. 1, pp. 1–11, 2002.

[32] N. Dlodlo and J. Kalezhi, "The internet of things in agriculture for sustainable rural development," in *Proceedings of the International Conference on Emerging Trends in Networks and Computer Communications, ETNCC 2015*, pp. 13–18, Namibia, May 2015.

[33] K. Fleming, P. Waweru, M. Wambua, E. Ondula, and L. Samuel, "Toward Quantified Small-Scale Farms in Africa," *IEEE Internet Computing*, vol. 20, no. 3, pp. 63–67, 2016.

[34] M. Mafuta, M. Zennaro, A. Bagula, G. Ault, H. Gombachika, and T. Chadza, "Successful deployment of a wireless sensor network for precision agriculture in Malawi," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 150703, 13 pages, 2013.

[35] N. Dlodlo, "The internet of things in transport management in South Africa," in *Proceedings of the International Conference on Emerging Trends in Networks and Computer Communications, ETNCC 2015*, pp. 19–26, Namibia, May 2015.

[36] A. H. Cloete, M. J. Booysen, R. C. Sandell, and A. B. van der Merwe, "Smart electric water heaters: A system architecture proposal for scalable iot," in *Proceedings of the ACM International Conference*, March 2017.

[37] W. Musakwa and T. Gumbo, "Impact of urban policy on public transportation in gauteng, South Africa: Smart or dumb city systems is the question," *Green Energy and Technology*, pp. 339–356, 2017.

[38] J. Ouoba, T. F. Bissyandé, and C. Béré, "Towards smart city implementations in Sub-Saharan Africa: The case of public transportation in Ouagadougou (Burkina Faso)," *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 166, pp. 78–90, 2016.

[39] M. Zennaro and A. Bagula, "Iot for development (IoT4D)," https://iot.ieee.org/newsletter/july-2015/iot-for-development-iot4d.html.

[40] M. N. S. Miazi, Z. Erasmus, M. A. Razzaque, M. Zennaro, and A. Bagula, "Enabling the Internet of Things in developing countries: Opportunities and challenges," in *Proceedings of the 5th International Conference on Informatics, Electronics and Vision, ICIEV 2016*, pp. 564–569, Bangladesh, May 2016.

[41] C. Pham, A. Rahim, and P. Cousin, "WAZIUP: A Low-Cost Infrastructure for Deploying IoT in Developing Countries," in *e-Infrastructure and e-Services for Developing Countries*, T. Bissyande and O. Sie, Eds., vol. 208 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 135–144, Springer International Publishing, 2018.

[42] J. McKeown, "LoRa: A Communications Solution for emerging LPWAN," *LPHAN and Industrial Sensing IoT applications , Semtech, Tech. Rep*, 2016.

[43] Y. Z. Ibrahim, H. Balzter, J. Kaduk, and C. J. Tucker, "Land degradation assessment using residual trend analysis of GIMMS NDVI3g, soil moisture and rainfall in Sub-Saharan West Africa from 1982 to 2012," *Remote Sensing*, vol. 7, no. 5, pp. 5471–5494, 2015.

[44] Asia-Pacific Fishery Commission, "Regional overview of aquaculture trends in the Asia-Pacific Region 2014," Food And Agriculture Organization Of The United Nations, Tech. Rep, 2014.