This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.2983873, IEEE Access

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Optimizing the Lifetime of IoT-Based Star and Mesh Networks

Hamid Al-Hamadi[1], Mohammad Saoud[2], Ing-Ray Chen[3], and Jin-Hee Cho[3]
[1]Department of Computer Science, Kuwait University, Safat 13060, Kuwait
[2]Department of Quantitative Methods and Information Systems, Kuwait University, Safat 13060, Kuwait
[3]Department of Computer Science, Virginia Tech, 7054 Haycock Road, Falls Church, Virginia, 22043, USA

Corresponding author: Hamid Al-Hamadi (e-mail: hamid@cs.ku.edu.kw).

This work was supported and funded by Kuwait University Research Grant #RQ02/18

ABSTRACT In this paper, we propose and analyze a novel optimization model to maximize the lifetime of Internet-of-Things (IoT) networks, including the Low Power Wide-Area Network (LPWAN) based Sigfox star networks and Time Slotted Channel Hopping (TSCH) mesh networks. An IoT cloud manages the IoT network adapting to sensed phenomenon changes in the deployment area retrieved from peered cloud-based environmental monitoring systems. While increasing the number of paths for IoT devices to cloud communication increases reliability, it also comes at the expense of increased energy consumption. We consider an optimization problem to determine the best redundancy level to be applied in the IoT network such that the lifetime is maximized while achieving the quality-of-service (QoS) requirements in the presence of unreliable sensing environments. Our model is generic and easily adaptable to a given IoT technology by considering the technology's devices, environmental, and protocol specifications while spanning single-hop, multi-hop, short-range, and long-range IoT technologies. We formulate the tradeoff between energy conservation vs. reliability of an IoT network as an Integer Non-Linear Programming (INLP) optimization problem. The feasibility of our approach in maximizing the lifetime of IoT networks for both the star and mesh network topologies is demonstrated using SigFox and TSCH as representative technologies, respectively. We conduct an extensive comparative performance analysis demonstrating that our model outperforms contemporary baseline models in both SigFox and TSCH IoT network technologies.

INDEX TERMS Internet of things, INLP, optimization, LPWAN, lifetime maximization, fault tolerance.

I. INTRODUCTION

Low-cost network deployments, cheaper equipment, and high flexibility make deploying a sensor network an attractive wide-area monitoring solution [1], [2]. Tinymesh [3] and VERICOM [4], [5] are examples of wireless mesh network infrastructures available in the market where digital and analog inputs can be monitored and shared with an Internet-of-Things (IoT) cloud using multi-hop wireless network protocols. The network is flexible with regards to node failures as it automatically adjusts communications to avoid disruptions. Such a network could be part of an IoT framework, providing the necessary data which would then be used for analysis and decision making in an IoT cloud. Earthquakes, volcanos, dust storms, heatwaves, winter storms, and hurricanes are a few examples of environmental phenomena that can impact the reliability of devices and their communications. Temperature and absolute humidity can have a negative correlation with wireless received signal strength indication (RSSI) and packet reception rate (PRR) [6, 7]. Heavy dust and sandstorms can cause an increase in propagation path loss [8, 9]. Furthermore, wind and weak earthquakes can cause alignment errors between point to point line of sight communications between a sender and a receiver causing fading of the received signal [10]. As a natural response, such real-time or predictive weather data can be incorporated to dynamically configure the network to improve the safety and reliability of its operation. The application domain of the IoT framework itself does not need to be related to environmental phenomena monitoring. For example, the application domain might be related to taking measurements of water, gas, or electricity from smart meters located in both urban and rural areas [1, 4]. Another example is monitoring the movement of habitats in a geographic area. In such cases, rather than adding various sensors to measure many unrelated phenomena that could affect the reliability of smart meters and monitoring devices, existing IoT platforms may already provide a backbone to integrate real-time data from clouds sharing the data of their environmental sensors and devices in the same geographic area [11]. It is expected that using clouds to provide services will increase in the future. Phenomena measurements data sharing between clouds would enable the IoT platform to make calculated decisions that would otherwise be infeasible to obtain due to the cost for procuring and maintaining the infrastructure and devices. In this work, we are concerned with a specific type of phenomena measurements called node/link failure data.

By obtaining node/link failure related data through data sharing, the expected communication failure from a monitoring device to a final gateway of the deployed IoT network can be derived. Deriving the expected communication failure in turn determines the best response to maximize lifetime of the deployed IoT network. That is, if the expected failure is high, using more devices to transmit the same message is a wise approach, since even if more energy is used (i.e., IoT devices are active as opposed to being in a low power sleep state), the probability of a message reaching an IoT gateway is increased. However, keeping all IoT devices in an active state is unnecessarily a waste of energy, thus minimizing the lifetime of the network without any gain in return. This is especially the case since most deployed IoT devices, e.g. SigFox, and Time Slotted Channel Hopping (TSCH), deployed devices have finite energy. Thus, the expected failure probability determines how many devices need to be active and transmitting over separate paths to satisfy a specified minimum message reliability requirement while minimizing energy consumption to prolong the system lifetime. This tradeoff between energy consumption vs. message reliability of the star and mesh networks forms a problem that is best described as an optimization problem for which the goal is to find the best number of paths (i.e., through imposing device status settings) to maximize the deployed IoT network lifetime.

In this paper, we propose and analyze an optimization model to maximize the lifetime of IoT networks. Using environmental data obtained from neighboring cloud systems, the IoT cloud finds the best redundancy level to be applied in its IoT network in order to maximize the lifetime of its operation while meeting Quality-of-Service (QoS) reliability requirements in the presence of unreliable sensing environments. The deployed network consists of nodes distributed over a wide geographic area and can be of mesh (such as Time Slotted Channel Hopping, or TSCH [12, 13]) or star (such as a low-power wide-area network or LPWAN of SigFox [14, 15]) topologies [16]. While the use of more sensing devices can be used to increase reliability, it also increases the consumption of the IoT network energy. Similarly, using more paths increases reliability while consuming more energy of an IoT network. We formulate the tradeoff between energy consumption and reliability of the IoT network through an Integer Non-Linear Programming (INLP) optimization problem taking into account the information obtained from the environmental cloud.
The key contributions of this paper are as follows:

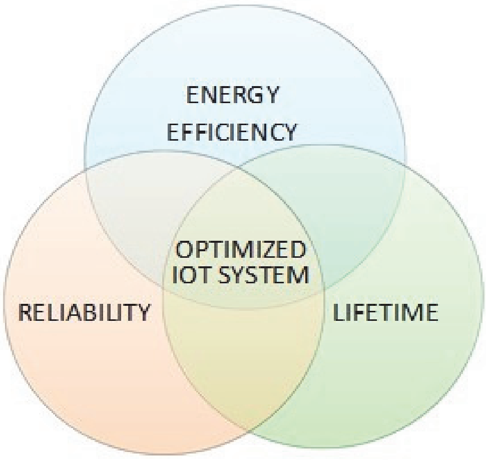


FIGURE 1. **Multiple considerations in optimizing IoT lifetime.**

1. We proposed a novel optimization model to maximize the lifetime of IoT networks and conducted its in-depth analysis. To the best of our knowledge, we are the first in considering energy and reliability of the emerging IoT networks in tandem (see Fig. 1) to provide an optimization model to maximize the lifetime while meeting operational constraints in the presence of attacks and environmental threats. The majority of existing literature regarding IoT energy modeling focuses on providing a basic platform for IoT energy analysis without considering the reliability of the operation and the lifetime of the deployed IoT network in unreliable environments [13, 16, 17]. Our work not only considers energy modeling for IoT but also provides an optimization model that considers the effects of both energy and reliability of the network to find the optimal parameters for maximizing the lifetime while satisfying QoS requirements of a given deployed IoT technology. We formulate the tradeoff between energy consumption and reliability of the IoT network through an Integer Non-Linear Programming (INLP) optimization problem.

2. Our optimization formulation is generic taking into consideration the energy model of technology, the reliability model of technology, and failure factors affecting the deployed technology. Our analytical formulation also considers single and multi-hop, short and long-range IoT technologies. Our model can be easily adapted to a given IoT technology by considering the technology's devices, environmental, and protocol specifications. To the best of our knowledge, we are the first to give an analytical solution and optimization model for IoT network lifetime, considering all of the above issues.

3. We demonstrate the feasibility of our approach, through detailed performance evaluation, using SigFox and TSCH as representative IoT technologies for LPWAN and mesh networks, respectively. We compare baseline models [13, 16] with our INLP model for both SigFox and TSCH, and show that our model outperforms these baseline models.

The rest of the paper is organized as follows. In Section II. we provide a literature survey of related work. In Section III, we discuss the system model and assumptions for both the Low Power Wide-Area Network (LPWAN) and mesh network topologies using SigFox and TSCH as representative technologies, respectively. In Section IV, we discuss the research problem to be solved and the solution methodology developed based on an optimization model that can maximize IoT network lifetime when given a set of input parameters characterizing the operational and environmental conditions. In Section V, we perform a performance analysis for both the LPWAN and mesh networks. Finally, in Section VI, we conclude the paper and outline future work.

II. RELATED WORK

Many research papers in IoT have focused on how real-time environmental sensor data can be used for decision-making. Wong and Kerkez [11] discuss how real-time environmental sensor data in an IoT cloud can be used for decision-making. They show how existing IoT platforms already provide a backbone to integrate real-time data from web-enabled environmental sensors and devices. A case study to determine water quality from Internet Protocol (IP) enabled hydrologic sensor nodes is discussed. Knowledge is updated through an API web service providing sensor node measurements to maximize the number of quality samples by updating the node's sampling frequency based on anticipated storm events. Our model leverages existing IoT clouds for gathering environmental data, but we used it to maximize the lifetime of the IoT network through calculating and updating optimal redundancy parameters. Ashok et al. [17] discuss cyber-physical security of Wide-Area Monitoring, Protection and Control (WAMPAC) in a smart grid environment. Safety and reliability of Wide-Area Monitoring (WAM) systems becomes increasingly important for advanced control applications. The authors discuss how WAM and protection is crucial in systems like NASPInet [18], which is a separate wide area network for Phasor Measurement Units (PMUs). Parameters like currents and voltages at different places are measured using sensors (e.g., PMUs), required all over the smart grid network to improve system visibility, and sent through the high-speed communication network to the Wide-Area Protection controller to implement smart and preventative control strategies. Our work also considers the importance of WAM and considers the use of WAM system information by a centralized control center for decision making. The authors in [19] propose an architecture for agricultural habitat information acquisition systems where environmental information in the farmland (e.g., light intensity, carbon dioxide content, soil temperature) is collected. Realtime warnings of abnormal parameters are sent to an online decision support system for decision-making. Our work similarly uses the information obtained from environmental sensors (via environmental clouds) for decision-making. Moreover, we consider environmental phenomena that effects network communication, with the optimal redundancy parameters derived by the control center to maximize the application lifetime. In [20] the authors propose an IoT based sensor system for sewage treatment plant monitoring. They use temperature, turbidity and pH sensors to monitor the sewage system and send real-time results to a cloud-based server. The system cloud data is stored for further analysis. While the collected data is proved to be efficient to monitor the systems status and as a decision-making tool, their application can be further enhanced by using the fault-tolerant optimization approach discussed in this paper. Furthermore, their protocol does not consider energy consumption of relaying devices, reliability of communication, nor prolonging system lifetime.

Deployed IoT devices are prone to both software and hardware faults. Moreover, the large scale of IoT requires efficient mechanisms to tolerate faults, thus many scholars have used fault-tolerance, redundancy management, and communication optimization to achieve IoT system objectives. Gautam et al. [21] propose a novel approach of fault management and restoration of network services in IoT clusters to ensure disaster readiness. Their approach depends on early detection and isolation of fault and subsequently providing alternate network paths. Our algorithm also uses multiple paths to avoid a single path failure from disrupting communication. Al-Hamadi and Chen [22] propose an analyzed a dynamic redundancy management of integrated intrusion detection and tolerance for maximizing the lifetime of clustered wireless sensor networks (WSNs). Multisource and multipath routing are used for intrusion tolerance with majority voting for intrusion detection in a redundancy management protocol design. They focused on a cluster-based WSN utilizing intrusion detection and tolerance in the presence of attackers whereas we focus on IoT cloud-based environmental monitoring and formulate a lifetime optimization problem. In [23] the authors propose a framework for data delivery in large-scale risky IoT networks, where data is relayed toward a gateway connected to the internet. Their approach is based on carefully choosing the next hop for routing while considering energy constraints, hop counts, and remaining energy levels. A comparison with energy-aware protocols is provided to show the effectiveness of the framework. Their framework however does not consider the effects of energy and reliability in tandem and fails to consider the environmental effects in the deployment area on the reliability of data delivery. In [24] the authors consider a multipoint-to-point network where multiple LoRa sensors send messages to a single LoRa gateway. The protocol aims to use redundant data measurements in order to enhance reliability of measurement delivery, where a frame includes current and past few measurements. A measurement is obtained if a single copy is able to arrive successfully. The protocol considers the effects of fading and interreference in order to determine the number of measurements to include in a single frame. This is similar to our work in that we consider the effects on the reliability of the communication in order to determine the number of devices sending data to the gateway. However, we consider energy lifetime and provide a generic protocol that can be adapted not only for star networks such as LoRa, but also for IoT mesh networks. In [25] the authors propose a protocol to enhance the QoS of wireless sensor network based IoT applications. The protocol considers lifetime, reliability, and traffic intensity when choosing the next-hop node to choose the optimal path. Their work however does not consider multiple paths nor using environmental data for finding the optimal routing configurations. The authors in [26] use the redundancy of network connections to ensure availability in IoT networks and tolerate drops in connections thus decreasing the probability of network downtime. They add a new level of abstraction to networking in order to send data over multiple networking solutions while using an optimization algorithm to determine the optimal and most reliable path for delivery over the IoT network. Their work however does not consider energy nor lifetime. The authors in [27] propose a bio-inspired particle multi-swarm optimization (PMSO) routing algorithm in a two-tiered WSNs that provides fault tolerance by selecting k-disjoint paths to route from sensor nodes to super nodes. Their objective is to minimize transmission power range and the average delay for all sensors while maintaining the k-disjoint multipaths. Their work focuses on WSN-based mesh networks only. Also, unlike our work, their work does not consider security or environmental failure conditions.

Designing for an energy-efficient IoT has become an important objective in resource-constrained environments. In [28], the authors propose a lifetime-aware resource allocation framework to maximize the network lifetime of cellular-based machine to machine (M2M) networks. They consider battery-driven smart devices deployed in remote areas to minimize energy consumption and thus maximize the network battery-lifetime as a priority. Network lifetime maximization is achieved by providing optimal scheduling decisions that consider the number of devices to be scheduled and the available resources to be allocated. Similar to our work, they consider energy and reliability in the form of signal-to-interference-plus-noise ratio to maximize network lifetime. Their model, however, only considers cellular-based M2M networks (i.e. star and not mesh) with limited failure factors affecting reliability of communication. Unlike our work, there is no security consideration. In [29], the authors propose a message disjoint security routing protocol for energy harvesting networks using solar energy. They explore the tradeoff between sleep state durations and transmission delay on information delivery, while considering energy consumption. Their scheme establishes two disjoint connected dominating sets, where one set is used to transmit data packets and the other set is used to verify data message sending at the receiver. Identification information is recorded in data packets and forwarding nodes for malicious node detection, thus providing a way for increasing delivery ratio by retransmission. Unlike our work, their method does not find the optimal number of redundant paths to achieve system lifetime maximization. Further, their method considers only malicious blocking attacks without considering other environmental failure factors in the area of deployment that can affect the required fault tolerance or reliability requirements. The authors in [30] propose an energy and reliability aware protocol for resource-constrained IoT devices where the tradeoff between reliability and energy is explored. They explore the effect of power amplification (PA) models on both the energy and reliability of IoT devices, highlighting the limited efficiency and linearity of traditional PA models. This leads to their derivation of an optimization model that optimizes certain performance metrics at the physical layer (modulation size, signal-to-noise ratio) and at the medium access control (MAC) layer (payload size and the number of retransmissions) as a function of link distance, while satisfying the IoT device hardware constraints. They show that by finetuning the model parameters, a link's lifetime can be maximized. Our work is different from [30] in that our parameter tuning is at the network and application layers, while their parameter tuning is at the physical and MAC layers. In [31], the authors propose an energy-efficient multi-objective scheduling model for monitoring-based IoT networks. Their objective is to minimize energy consumption and communication overhead of monitoring for each node while considering link faults due to energy limitations. The proposed model consists of a node subset generation phase which creates multiple vertex covers, followed by an optimized scheduling of vertex covers. They formulate the sequencing assignment between vertex covers as a multi-objective generalized assignment problem and a traveling salesman path problem. Their work however only considers IoT networks operating on a generated destination oriented directed acyclic graph (DODAG). Also, unlike our work which considers faults due to security, hardware, and energy failures, their work [31] only considers faults due to energy limitations. Rango et al. [32] analyze energy-aware communication between smart IoT monitoring devices. Energy consumption of devices with Wi-Fi and Radio Frequency (RF) interfaces is analyzed under different configurations by adopting an IoT energy framework [33]. They identify the best configuration to prolong the lifetime of IoT devices. However, their focus is on energy and they do not consider the reliability of the IoT network. In [34] the authors stress the important role of IoT in smart grids and smart metering, enabling efficient control and management of cities. They discuss the value of IoT in AMI (Advanced Metering Infrastructure), especially in providing power quality and reliability monitoring. In their survey, they highlight important issues which enhance the large-scale network deployment in the smart grid, including using optimization algorithms for network reconfiguration and reacting to events effecting power quality and reliability. Our work uses optimization to find optimal settings maximizing lifetime while considering reliability constraints for the deployed application network, which can be applied to enhance smart grid deployments. Martinez et al. [35] define a general methodology for modeling energy consumption of IoT devices. The proposed framework models key components of point-to-point communications, such as SigFox [14] and also mesh networks, such as TSCH [12] networks. The comprehensive framework is helpful in evaluating and contrasting different technologies by plugging in their empirically quantified parameters representing the platform and the operating conditions. Their work however does not consider modeling the reliability of the IoT network and does not consider optimizing its lifetime. Morin et al. [16] provide a comparison of the device lifetime in IoT networks. They develop an analyzer that computes the energy consumption of IoT devices for varying IoT technologies including short range and emerging long-range technologies (e.g., SigFox and LoRa) based on the transmission, reception, idle, and sleep states of the devices and their respective durations. They consider energy-constrained nodes and use the analyzer to derive the expected lifetime for the IoT device based on the used technology and parameters. The authors claim that the analyzer is helpful for IoT network designers to understand the effects of a type of IoT technology and associated parameters on the lifetime of the IoT device. Vilajosana et al. [13] provide a slot-based modeling of energy consumption of TSCH mesh networks. The authors model the process of the energy consumption occurring in each slot in a TSCH slot frame for both relay and leaf motes. The state of mote modules (i.e., micro-controller and radio) in each type of TSCH slot is determined while the total energy consumed per slot frame is derived accordingly. The authors show how the model can be used to evaluate energy consumption for different TSCH network configuration choices. Later in Section V, we will use the baseline models by Morin et al. [16] and Vilajosana et al. [13] for performance comparison with our INLP model in the TSCH and SigFox technologies, respectively. Daneels et al. [36] use the same model as in [13]; they extend the model by providing detailed modeling for time slot and state energy consumption for TSCH, and provide energy consumption measurements for both 868 MHz and 2.4 GHz frequency bands using dual-band OpenMote device running the OpenWSN firmware. Our work is different from [13] and [36] in that our work focuses on the MTTF or the failure probability of the IoT networks which depends on multiple failure factors including energy consumption failure, communication failure, hardware failure, node compromise failure, and technology failure, while [13] and [36] focused on only the correctness of energy consumption of IoT devices and comparing with datasheets and real world collected measurements. We adopt their energy consumption collection model which they (along with ([16, 35]) have shown to work for IoT devices (so how a single device consumes energy based on states and time spent in states). Specifically, we follow their energy consumption model that the energy consumed per state follows $t_s \times P_s$ where t_s is the time spent in the state and P_s is the power consumption based on both CPU and Radio for the given state. Our analysis is novel with respect to [13] and [36] in that other than energy failure, we deal with communication failure, hardware failure, node compromise failure, and technology failure which can contribute to IoT network failure, and unlike [13] and [36] we perform reliability modeling of the best multipath routing strategies considering all failure causes for maximizing the IoT network lifetime.

III. SYSTEM MODEL

As illustrated in Fig. 2, our system model of an IoT-based cloud platform consists of a control center cloud, an IoT star/mesh network co-located with an IoT environmental monitoring network, and a number of environmental clouds for collecting and relaying monitored data. The control center cloud is responsible for communicating and controlling the deployed IoT network. The control process that executes our proposed optimization model to maximize the IoT network lifetime is sitting in the control center cloud. The IoT environmental monitoring network consists of IoT nodes distributed over a wide geographic area in either mesh (e.g., TSCH) or star (e.g., LPWAN) topologies [16].

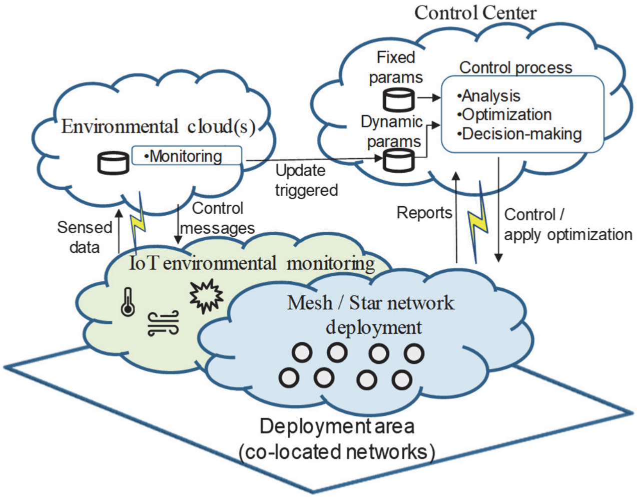


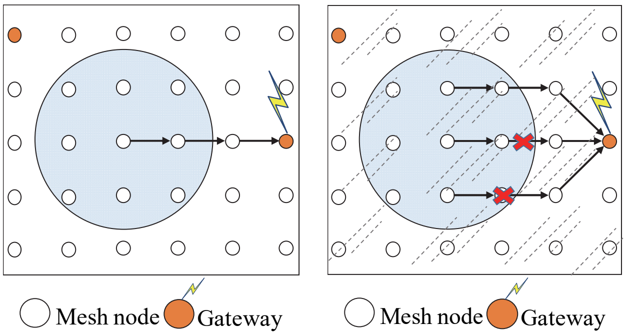
FIGURE 2. System Model of an IoT cloud platform.

Environmental monitored data are passed from IoT nodes to the environmental clouds which in turn relay data to the control process in the center cloud for analysis and decision making. In the case of a mesh network, mesh nodes communicate their sensed data through multiple hops to reach any of the multiple gateway nodes that directly communicate with the control center. Multiple paths can be created between source nodes generating data and gateway nodes. The control center dynamically configures its mesh nodes based on the data retrieved from the environmental cloud and its deployed monitoring infrastructure co-located in a same area. While the environmental monitoring could be part of the deployed network itself, this separation of concerns helps the deployed IoT network (managed by the control center) to focus on its main application (e.g., relaying smart meter information) instead of needing to invest in resources to build and maintain an environmental monitoring infrastructure (e.g., monitoring dust). The control center only needs services of the environmental cloud and does not need to own it. Real world implementations exist where the deployed network [5] uses multiple batteries operated mesh nodes with a manager; however, they do not use information from other clouds or consider finding the optimal number of paths for redundancy management to prolong lifetime. For example, environmental phenomena can be obtained via IoT devices of peered clouds running on IoT cloud platforms, such as Xively [37], ThingSpeak [38], ThingWorx [39], Google Cloud Platform [40] and many more, all of which support real-time feeds and notifications. Real-time and historical data can be further obtained by government organizations, such as United States Geological Survey (USGS) [41], which provides data regarding earthquakes, floods, and other environmental hazards. Global environmental communities consisting of shared weather stations and air quality monitor from different entities, such as Weather Underground [42], can be further used as a data source. Additionally, tampering incidents can be obtained using a geographic information system (GIS)-based crime mapping tools and applications [43, 44], some of which are readily available to the public (e.g., ArcGIS used by Hailfax Regional Police [45]).

In this paper, we consider using the obtained environmental data to optimize the redundancy level needed to satisfy reliability of the operation of the deployed network and prolong its lifetime. Unlike a mesh network, a star network is formed when the nodes directly communicate with the gateway based on a single-hop. LoRa [46] and SigFox [14] are the examples of such deployments. As environmental conditions become harsh, the probability of reliable monitoring of an area and subsequent propagation of this data to the control center decreases. Data from the environmental cloud is used by the control center to find the best redundancy configurations to apply in order to achieve the required reliability and prolong the system lifetime.

Fig. 3 illustrates a scenario showing how the control center monitors a specific geographic location using a mesh network based on current environmental conditions. When the environment conditions are clear, a single source located within the radius of the queried area relaying the information over a single path achieves the required reliability of service (i.e., QoS) by the mesh network. On the other hand, when the environment conditions are harsh, achieving the same QoS requires 3 sources to report their data over separate paths to the control center. Gateway nodes collect the data and directly communicate with the control center in the IoT cloud. In the case of LPWANs, the number of IoT devices reporting the same phenomenon can be configured based on the probability of failure. Each device would transmit a direct long-range transmission to the gateway which in turn forwards the packet to the control center. Fig. 4 depicts this scenario. We consider multiple gateways evenly distributed in the network where each gateway is responsible for relaying data sent from the LPWAN nodes to the control center cloud. We consider a design where nodes are deployed around gateways such that all nodes can reach the closest gateway over about the same distance, hence implying similar energy consumption (due to using similar power) to report to the gateway. The number of gateways required is LPWAN technology dependent, such that nodes must be able to communicate with gateways, and the gateways collectively must be able to provide the required coverage over the deployment area.

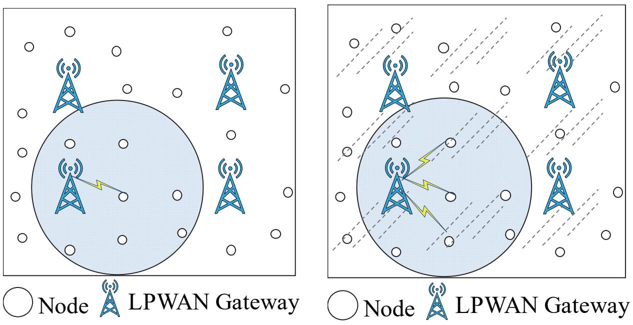
Similarly, for TSCH mesh networks, we consider a deployment where multiple gateways exist and the distance between a mesh node to its nearest gateway is about the same for every node in the system so there is a balance of energy consumption and packet transmission time delay for every node in the system. Fig. 3 and Fig 4. scenarios show how redundancy management can be applied to IoT technologies, such as TSCH for mesh networks and SigFox for LPWANs.



(a) Single path used under clear conditions.

(b) Multiple paths used under harsh conditions.

FIGURE 3. Mesh network replying to a location-based query.



(a) Long-range transmissions by a single device under clear environmental conditions.

(b) Long-range transmissions by multiple devices under harsh conditions.

FIGURE 4. LPWAN replying to a location-based query.

In our model, we consider the following failure factors:

- *Hardware failure* is caused by environmental phenomena introducing physical damage and malfunction of the deployed IoT devices.
- *Communication failure* is caused by environmental phenomena and interference that can negatively impact network link quality of the deployed IoT devices. This type of failure is technology-dependent and can be further influenced by many other factors, such as potential interference of the used spectrum by industrial applications and/or home-automation technology, e.g. interference of 868 MHz industrial, scientific and medical (ISM) band on LoRa and SigFox [46, 47].
- *Node compromise* can occur due to being compromised by an attacker performing physical tampering of the IoT device. A compromised device is unresponsive and no longer interacts with the deployed network.

Information related to environmental phenomena, tampering incidents, and/or interfering signals are all location-dependent and can be potentially obtained via peered clouds or via logged historical information when necessary, all of which can be translated to a failure probability of the IoT device. We model these location-based failure factors in Section IV-B.

IV. PROBLEM DEFINITION AND SOLUTION METHODOLOGY

A. PROBLEM DEFINITION

Our research problem is to determine the optimal number of paths for query/response message passing in an IoT network, given a set of input parameters characterizing the operational and environmental conditions, such that the deployed IoT network lifetime is maximized.

TABLE I
NOTATIONS

Symbol	Meaning	Type
A	Side length of deployment area (meter)	input
n	Number of deployed IoT nodes	input
n_j	Reachable neighbors per node	derived
r	Transmission range of IoT technology (meter)	input
E_0	Initial energy of IoT node (Joule)	input
E_{init}	Total energy of all IoT nodes (Joule)	derived
SF_{size}	TSCH Slot Frame size	input
$S_{duration}$	TSCH slot duration (ms)	input
$SF_{duration}$	TSCH Slot Frame duration (ms)	derived
N_{msg}	SigFox maximum number of messages allowable per device per day	input
t_{msg}	SigFox message transmission duration (ms)	input
N_{trans}	Number of repetitive transmissions per message	input
t_{Δ}	Inter-arrival time between messages	derived
hw_j^x	Probability of hardware failure of node j at location x	input
$comm_{jk}^x$	Probability of communication failure of node j at location x	input (dynamic)
$comp_j^x$	Probability of node j compromise at location x due to attacker tampering	input (dynamic)
$Pf_{hop,jk}^x$	1-hop failure prob. for IoT node j located at x	derived
N_{hops}	Number of hops to reach gateway	derived
E_s	Energy consumption per node in state s (Joule)	derived
P_s	Power consumption per node in state s (W)	derived
E_{LN}	Energy consumption of TSCH leaf node (Joule)	derived
E_{RN}	Energy consump. of TSCH relay node (Joule)	derived
E_{SN}	Energy consump. of TSCH sleep node (Joule)	derived
E_{rd}	Redundancy related energy consumption for the i'th query (Joule)	derived
E_{nrd}	Non-redundancy related energy consumption for the i'th query (Joule)	derived
M_p	# paths used to transmit a report to the gateway	design
R_q	Reliability of a report	derived
R_{req}	Minimum acceptable reliability of a report	input
N_q	Expected number of queries/reports that can be handled by the system before energy depletion	derived
$MTTF$	Lifetime of the IoT cloud	output

This research aims to find the best redundancy level in terms of the number of paths for query/response message passing so it can best balance the tradeoff between energy consumption (leading to a shorter lifetime) vs. message reliability (leading to a longer lifetime). Table I lists the parameters along with their physical meanings and types. A parameter is labelled as input, derived, design, or output based on its type. Specifically, our design parameter is M_p (i.e., the number of paths used to transmit a report to the gateway), which serves as the decision variable to maximize the mean time to failure (MTTF) of the IoT network as the only output parameter. Input parameters serve to characterize operational and environmental conditions. Therefore, given a set of input parameter values specifying the operational and environmental conditions as input, our optimization model would decide the optimal M_p value that maximizes the MTTF of the IoT network. Finally, derived parameters are those deriving their values from input parameters and are hidden from the user who uses our optimization model to decide the optimal M_p maximizing the MTTF of the IoT network.

As seen in Table I, there are 13 input parameters whose values are given as input to our optimization model to be executed by a control process sitting in the control center cloud. Among the 13 input parameters, 11 parameters are set locally by the control process based on design configuration settings and protocol specifications and have their values predetermined at the system deployment time, namely, A, n, r, E_0, SF_{size}, $S_{duration}$, N_{msg}, t_{msg}, N_{trans}, R_{req}, and hw_j^x (described below in IV.A.3 through IV.A.13). The remaining 2 parameters namely, $comm_{jk}^x$ and $comp_j^x$, have dynamic values and are related to the location-based failure factors discussed in the previous section. These 2 dynamic parameters can be supplied from the environmental cloud and are labeled as dynamic inputs in Table I.

In case the environmental cloud cannot provide the two dynamic parameters due to communication failure, the control process predicts their values so that the optimization method can continue to run in the control center cloud for service continuity without delay. We explain below how the two dynamical parameters are assigned values at runtime with an algorithm description and two illustrations.

Algorithm 1 Obtain dynamic parameter y

```
Ask for data regarding y from environmental cloud
if (received)
    derive y using received data
else
    predict y at the control center
```

The scenario that the environmental cloud can provide the values of the two dynamic parameters is shown below in Fig. 5; it follows the path 1-2-3 in the algorithm description.

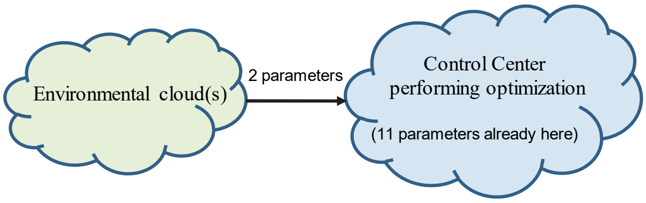


FIGURE 5. Control center successfully receiving dynamic parameters.

The scenario that the environmental cloud cannot provide the values of the two dynamic parameters due to communication failure is shown below in Fig. 6; it follows the path 1-2-4-5 in the algorithm description.

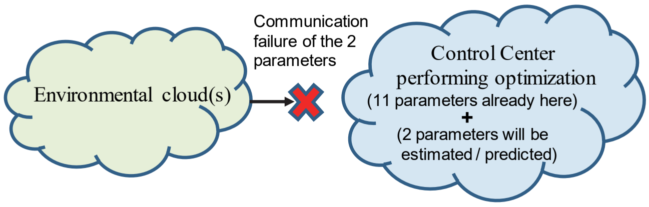


FIGURE 6. Control center predicting the dynamic parameters due to failure of communicating with environmental cloud.

Below we explain how the IoT cloud platform can measure and parameterize (i.e., give values to) these 13 input parameters, starting with the 2 dynamic parameters:

1. $comm_{jk}^{x}$: This parameter reflects the probability of communication failure of node j at location x when communicating with the next hop neighbor k. We parameterize it by 1 – *LQE* where *LQE* (standing for link quality estimator) is an industrial standard measurement [48] indicating the quality of a radio link in a location. The environmental cloud can measure environmental phenomena that have a high correlation with the link quality, such as Received Signal Strength Indicator (RSSI), Packet Reception Ratio (PRR), signal-to-noise ratio (SNR), and signal-to-interference-plus-noise-ratio (SINR), and provide location-specific *LQE* measures, i.e., $LQE(x)$ for location x, to the control center cloud of the cloud platform which can compute $comm_{jk}^{x}$ as $1 - LQE\ (x)$. If the environmental cloud cannot provide $comm_{jk}^{x}$ due to communication failure, the control center predicts *LQE(x)* by using a *LQE(x)* prediction method [37][38] which relates *LQE(x)* to two probability factors: background noise P_n and received signal strength P_r at location x. The background noise P_n can be modeled by the alpha-stable distribution with four parameters which can be parameterized (given values) by the control process based on the environment information at location x of the deployed IoT network. The received signal strength P_r can be modeled by a log-distance path loss model with three parameters which can be parameterized by the control process based on the environment information at location x and the distance separating two communicating nodes reside in the IoT network. This distance separating two communicating IoT devices in the IoT network is known to the control process because the star/mesh topology structure information of the deployed IoT network is known before deployment. The control process therefore can model both background noise P_n and received signal strength P_r at location x as random variables following certain distributions as described above. In effect, *LQE(x)* can be modeled as a random variable whose PDF (probability density function) is obtained by the convolution of PDFs of P_n and P_r. With the PDF of *LQE(x)* in hand, the controller can simply predict $comm_{jk}^{x}$ at runtime as $1 - E[LQE(x)]$ where $E[LQE(x)]$ is the expected value of the $LQE(x)$ random variable.
2. $comp_{j}^{x}$: This parameter reflects the security failure probability of node j at location x due to capture attacks (theft or device tampering so a device is compromised and does not perform the intended functions). It can be measured by the environmental cloud by leveraging Geographic Information System (GIS)-based crime mapping tools and applications [31, 32] which provide the number of incidents logged in a location x over a time period with which the security failure rate at a location can be derived. With the information of security failure rate for node j at location x known, the environmental cloud can parameterize $comp_{j}^{x}$ as the insecurity of the sensor. That is, $comp_{j}^{x} = 1 -\ e^{-CA_{j}^{x}t}$ where CA_{j}^{x} is the security failure rate of node j at location x in the IoT network measured by the environment clouds and t is the elapsed time since deployment. If the environmental cloud cannot provide $comp_{j}^{x}$ due to communication failure, the control process predicts it by modeling the security failure time of node j as a random variable following the exponential distribution with rate CA_{j}^{x}, such that $comp_{j}^{x}$ at current time t' is predicted as $F(t') = 1\ -\ e^{CA_{j}^{x} * t\prime}$ where F is the cumulative distribution function of failure time and CA_{j}^{x} is the estimated security failure rate of node j based on the environment information at location x prior to deployment time.
3. hw_{j}^{x}: The parameter reflects the probability of hardware failure of node j at location x. An IoT device equipped with sensors such as a Bloomsky, RainWise, or Ambient Weather 1002 (all of which are manufactured personal weather stations [42] for measuring temperature, humidity, wind speed, rainfall, and solar radiation) at the time of market release come with a manufacture documented "hardware failure rate" which is an industrial standard measurement [48], allowing a buyer to have some idea about how long a sensor will last. Furthermore, since there are many different types of sensors being manufactured with different sensing functions in mind, each sensor was thoroughly tested in a specific environment in which the sensor will likely be deployed. The specific sensor testing environment published maps

to the location of a deployed sensor. Thus, with the hardware failure rate and the manufacturer's published sensor testing environment , one can parameterize hw_j^x as the hardware unreliability of the sensor. That is, $hw_j^x = 1 - e^{-H_j^x t}$ following the exponential failure law [48] where H_j^x is the published hardware failure rate of node j with the sensor testing environment matching the sensor's location x in the IoT network and t is the elapsed time since deployment.

4. A: The perimeter of the operating area of the IoT network. We parameterize it to 1 km for TSCH according to [36] and to 10 km for SigFox according to [14, 15] .
5. n: The parameter defines the number of IoT devices deployed in the IoT network. We parameterize it to 50 for TSCH according to [5] and to 10 for SigFox [14, 15] .
6. r: The parameter reflects the radio transmission range. We parameterize it to be in the range of $[70m, 200m]$ reflecting the TSCH technology [36].
7. E_0: The initial energy of an IoT device. We parameterize it to 13.5 kilojoules (KJ) which corresponds to an IoT device with an energy of two AAA batteries as adopted in [16].
8. SF_{size}: The parameter defines the TSCH frame size. We parameterize it to 100 according to [36].
9. $S_{duration}$: The parameter defines the TSCH slot duration. We parameterize it to 15 ms according to [36].
10. N_{msg}: The parameter defines the maximum SigFox messages allowable per device per day. We parameterize it to 140 according to the SigFox standard [15, 16].
11. t_{msg}: The SigFox message transmission duration. We parameterize it to 2.08 sec according to [15, 16].
12. N_{trans} : The number of repetitive transmissions per SigFox message. We parameterize it to 3 according to [15, 16].
13. R_{req}: The minimum reliability of a report below which the IoT network service is not sustainable because critical, timely reports cannot be delivered reliably. It is a system requirement. We parameterize it to [0.9, 0.9999999] to model a varying range of reliability requirement [48].

At system deployment time, the control center estimates the values of $comm_{jk}^x$ for all pairs (j,k) and $comp_j^x$ for all j's at location x for the IoT network (following parameterization techniques discussed in IV.A.1 and IV.A.2 above respectively). Once the control process receives an updated value for these parameters, it replaces old stored values with the new values (see Fig. 2) reactively, based on environmental cloud value update messages. However, $comm_{jk}^x$ and $comp_j^x$ do not need to be updated frequently because communication link and node security failures are expected to happen infrequently compared to the deployed IoT star/mesh message transmission requests. Therefore, the control process can apply existing stored values as inputs into the optimization model for a large number of message transmission requests before an update to these dynamic parameters is received.

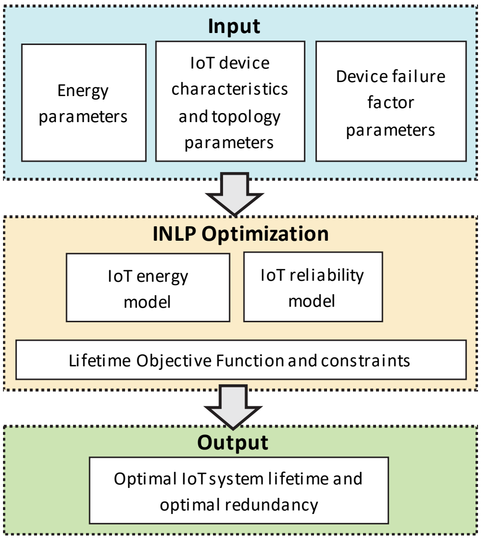

Input
Energy parameters
IoT device characteristics and topology parameters
Device failure factor parameters
INLP Optimization
IoT energy model
IoT reliability model
Lifetime Objective Function and constraints
Output
Optimal IoT system lifetime and optimal redundancy

FIGURE 7. Optimization model for IoT lifetime.

B. SOLUTION METHODOLOGY: BUILDING AN OPTIMIZATION MODEL TO MAXIMIZE IOT NETWORK LIFETIME

Our solution methodology lies in formulating an optimization model that can determine the optimal number of paths for query/response message passing to maximize the IoT network lifetime when given a set of input parameters characterizing the operational and environment conditions, as described in Fig. 7. We formulate the optimization problem using an INLP technique based on TSCH and SigFox IoT technologies for the mesh network and LPWAN star network, respectively. Below we describe how we model location-based failure, query reliability, general energy consumption, TSCH energy consumption, SigFox energy consumption, and the formulation of the optimization model.

1) LOCATION-BASED FAILURE

The probability of an IoT node to communicate successfully with its peer is dependent upon its location. We consider hardware failure, communication failure, and node compromise as location-based factors that determine the failure probability of of a node j at a location x to relay a report towards the control center.

The clouds can measure environmental phenomena that have a high correlation with link quality estimators, such as RSSI, PRR, SNR, SINR, in order to derive a link quality estimate, LQE [49]. The LQE is technology-dependent and can be further influenced by many factors, such as interference [46]. Using the LQE of location x obtained from peered clouds, the control center can then derive the probability of failure of an IoT node j at location x to send a report to node k due to communication failure, which can be expressed as:

$$comm_{jk}^x = 1 - LQE_{jk}^x \qquad (1)$$

where LQE_{jk}^x is the measured quality link estimator obtained from the environment cloud. In the case where

LQE_{jk}^{x} cannot be obtained then the controller simply predicts it using $E[LQE(x)]$ (as discussed in Section IV.A). The hardware failure of an IoT node j at location x, hw_j^x, is also dependent upon its physical surroundings, including environmental effects. Finally, an IoT node is susceptible to tampering, compromise and theft which can be derived from statistics regarding the geographic area of deployment and the probability of being compromised due the susceptibility is denoted by $comp_j^x$. Thus, we derive a 1-hop failure probability for an IoT node located at x based on the location-based failure factors as:

$$Pf_{hop,jk}^{x} = 1 - \left(1 - hw_j^x\right)\left(1 - comm_{jk}^{x}\right)\left(1 - comp_j^x\right) \quad (2)$$

2) QUERY RELIABILITY

Given that an IoT node can forward its report via any of its n_j neighbors, we derive the probability of success to send to at least one next hop neighbor:

$$\theta_j = 1 - \prod_{k=1}^{n_j} Pf_{hop,jk} \quad (3)$$

A report is sent over a path consisting of multiple hops until it is finally delivered to a gateway node which is in contact with the control center. The success probability of sending over a single path i to the control center is:

$$\theta_{path,i} = \left(\prod_{j=1}^{N_{hops}-1} \theta_j\right) \times \left(1 - Pf_{hop,jk}\right) \quad (4)$$

where $\left(\prod_{j=1}^{N_{hops}-1} \theta_j\right)$ is the probability of success to relay the message over $N_{hops} - 1$, and $\left(1 - Pf_{hop,jk}\right)$ is the success to send to the final node over the final hop (i.e. to the gateway node). The number of neighbors n_j is based on deployment density $\lambda = (\frac{nodes}{area})$. Since the message is being relayed towards a direction (e.g. quadrant $f = \frac{1}{4}$), then n_j approximately equals $f\lambda\pi r^2$, where r is the transmission range of the nodes, which is technology dependent. For TSCH mesh networks, we consider a deployment where multiple gateways exist and the distance between a mesh node to its nearest gateway is about the same for every node in the system so there is a balance of energy consumption and packet transmission time delay for every node in the system. Hence, the average number of hops, N_{hops}, between a mesh node and a gateway node is roughly equal to this distance divided by the radio range. For SigFox or LoRa LPWAN star topology network, each node uses a single-hop long-range wireless transmission to communicate with its nearest gateway in which case $N_{hops} = 1$.

To increase the probability of a report from location x being delivered to the control center at a given reporting interval, many IoT nodes deployed in location x may send their individual reports over separate paths to reach the gateway node. The failure probability of all sources to deliver the data to the control center is derived by:

$$Q_f = \prod_{i=1}^{M_p} (1 - \theta_{path,i}) \quad (5)$$

where M_p is the set of paths for multipath routing from an IoT device to its nearest gateway; it is a decision variable to be determined from our INLP network lifetime optimization model. It follows that the reliability is:

$$R_q = 1 - Q_f \quad (6)$$

In the case where single-hop long-range wireless transmission is used in a star topology (e.g., SigFox [14] and LoRa [46] LPWAN technologies), we can use multiple source node devices located in the same region to transmit directly to the control center k over a single path, giving a total of M_p paths. Thus, the reliability can be rewritten by:

$$R_q = 1 - \prod_{i=1}^{M_p} Pf_{hop,ik} \quad (7)$$

where $Pf_{hop,ik}$ is the failure probability to transmit directly to the control center k over a single path i (where each path consists of a single long range hop).

3) GENERAL ENERGY CONSUMPTION

Energy consumption is dependent upon a chosen IoT wireless technology and associated topology which in turn is based on the application requirements. We follow the generic approach used in [16, 35] to consider the energy consumption by an IoT device, which is estimated based on the time spent in each of the $Tx, Rx, Sleep,$ or $Idle$ states. In these models the energy consumed per state is based on the form $t_s \times P_s$ where t_s is the time spent in state s and P_s is the power consumption when in state s. We normalize the energy consumption over N_q application reports. Denote the total energy of the system at deployment by E_{init}, obtained by:

$$E_{init} = \sum_{i=1}^{N_q} E_{q,i} \quad (8)$$

where $E_{q,i}$ is the total energy spent by all the deployed nodes between time of query i and time of query $i + 1$. We consider that this time between sending successive queries is the same, and is denoted by the query interval (t_Δ).

Let $E_{rd,i}$ be the redundancy-related energy consumption by n nodes, and $E_{nrd,i}$ is the non-redundancy related energy consumption for the i'th query. $E_{q,i}$, $E_{rd,i}$, and $E_{nrd,i}$ are given by:

$$E_{q,i} = E_{rd,i} + E_{nrd,i} \quad (9)$$

$$E_{rd,i} = M_p \times N_{hops} \times (E_{Tx,i} + E_{Rx,i}) \quad (10)$$

$$E_{nrd,i} = n \times \sum_s E_{s,i} \ \ , where\ s \in \{Tx, Rx, idle, sleep\} \quad (11)$$

Eq. (10) accounts for the energy consumption required for reporting a single report utilizing M_p paths and an average of N_{hops} to reach a gateway node. $E_{i,Tx}$ and $E_{i,Rx}$ are the total energy spent for the transmission and reception of a single data packet over one hop in the i'th reporting interval, respectively. Eq. (11) explains the ongoing general energy consumption required by all IoT nodes in the system for the control and management of the network, in addition to the energy required for reporting. We measure this energy consumption per node attributed to transmission, reception, idle and sleep states denoted by $E_{T_x,i}$, $E_{R_x,i}$, $E_{idle,i}$, and $E_{sleep,i}$, respectively. The amount of energy per state and its applicability is dependent upon the multi-hop technology. For single-hop long-range IoT transmission technologies, we set $N_{hops} = 1$, and omit the receiving energy consumption of the receiving device at the control center since it is electricity powered (as opposed to battery). Thus, $E_{rd,i}$, the redundant energy consumed by node i is estimated by:

$$E_{rd,i} = M_p \times E_{T_x,i} \qquad (12)$$

4) TSCH ENERGY CONSUMPTION

To analyze the energy consumption of a deployed TSCH network, we distinguish each node based on its role in the propagation of a query to the gateway node in a mesh network. Fig. 8 illustrates the example of a TSCH network in operation where leaf nodes sense environment characteristics and propagate messages to relay nodes. Relay nodes relay the messages over multiple hops to finally reach a gateway node. Sleep nodes do not participate in the message passing of the query; but we consider the energy consumption during the sleep time. Our model considers a TSCH schedule where n nodes alternate the roles between leaf, relay, and sleep in order to fully cover the deployment area [35, 36].

Each leaf, relay, or sleep node consumes energy of E_{LN}, E_{RN}, or E_{SN}, respectively. To this end, we adjust Eq. (10) and Eq. (11) by:

$$E_{rd} = M_p \times (E_{LN} + (N_{hops} - 1) \times E_{RN}) \qquad (13)$$

$$E_{nrd} = [n - (M_p \times N_{hops})] \times E_{SN} \qquad (14)$$

The IEEE 802.15.4e TSCH mode has 7 types of time slots [12] which occupy the slot frame ϵ $\{TxDataRxAck$, $RxDataTxAck$, $TxData$, $RxData$, $RxIdle$, $Sleep$, $TxDataRxNoAck\}$. Each type of node consumes energy based on the allocated combination of these slot types within its slot frame schedule [36] as:

$$E_{LN} = E_{RxIdle} + E_{TxDataRxAck} + E_{Sleep} \times (SF_{size} - 2) \qquad (15)$$

$$E_{RN} = E_{RxDataTxAck} + E_{TxDataRxAck} + E_{Sleep} \times (SF_{size} - 2) \qquad (16)$$

$$E_{SN} = E_{RxIdle} + E_{Sleep} \times (SF_{size} - 1) \qquad (17)$$

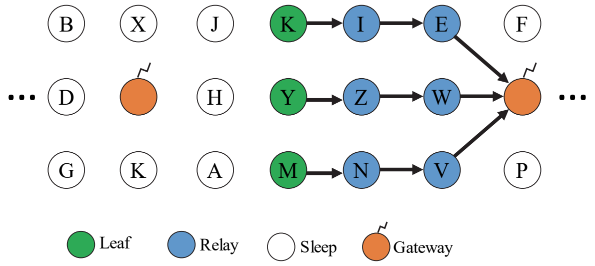


FIGURE 8. An example TSCH network in operation: relay nodes propagating messages from leaf nodes to a gateway.

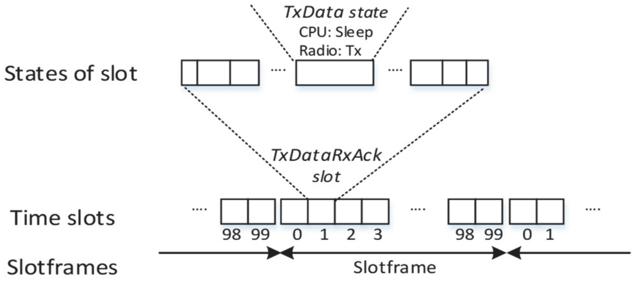


FIGURE 9. TSCH composition of time slots within a slot frame of size SF_{size}=100.

A single time slot has a fixed duration and is composed of several states each with varying CPU and Radio states, and varying durations, all of which determine the overall energy consumption for a time slot. We illustrate this in Fig. 9 where in the example schedule, the second time slot is of type $TxDataRxAck$ composed of many states which run in succession, one of which is $TxData$ responsible for data transfer which runs with CPU and Radio in Sleep and Tx states, respectively. Thus, the energy consumed per state follows $t_s \times P_s$ where t_s is the time spent in the state and P_s is the power consumption based on both CPU and Radio for the given state [35]. A slot frame has a bounded size of time slots denoted by SF_{size} where each time slot has a fixed duration of $S_{duration}$, thus resulting in the slot frame duration $SF_{duration}$ being equal to $SF_{size} \times S_{duration}$. The full composition of states per time slot is beyond the scope of this paper and can be found in [12, 36].

5) SIGFOX ENERGY CONSUMPTION

In SigFox [14], a message transmission is repeated 3 times (N_{trans}=3) to increase the probability of delivery at the receiver where each message is transmitted with a duration of t_{msg} and consuming P_{Tx} power needed for SigFox transmission [35]. The energy needed to transmit a Sigfox message is captured by:

$$E_{rd} = M_p \times t_{trans} \times P_{Tx} \qquad (18)$$

where $t_{trans} = N_{trans} \times t_{msg}$ and is the total transmission duration per message. The time interval between sending SigFox messages, t_Δ, is derived from the maximum number of messages allowable per device which is 140-message per day and is imposed by regulation [14, 16]. We can derive the remaining time in which reporting devices are in sleep mode in addition to the energy consumed by non-reporting devices which were set to sleep mode by the control center as follows:

$$E_{nrd} = n_{rep} \times (t_\Delta - t_{trans}) \times P_{Sleep} + n_{nrep} \times (t_\Delta) \times P_{Sleep} \qquad (19)$$

where the total number of nodes $n = n_{rep} + n_{nrep}$. We assume that the control center can alternate nodes between active (reporting) and sleep (non-reporting) states such that fair energy consumption is achieved among the Sigfox devices.

6) FORMULATION OF OPTIMIZATION MODEL

Equation 20 below formulates the MTTF objective function of our optimization model. This objective function depends on two parameters R_q and N_q which are derived by the reliability and energy models respectively. R_q is in Eq. 7 following the derivation of Eqs. 1-6. $N_q = \frac{E_{Tot}}{E_q}$ depends on the E_q parameter which is given in Eq. 9 following the derivation in Eqs. 10-19. Our optimization model formulation is also shown at a high level in Fig. 7. Essentially, our INLP optimization model aims to optimize the MTTF objective function, which in turn depends on how R_q and N_q are derived in Eqs. 1-19.

Based on the notations introduced in Section IV, our objective is formulated by:

$$Maximize \qquad MTTF = \frac{R_q\left(1-R_q^{N_q}\right)}{1-R_q} \tag{20}$$

where *MTTF* refers to Mean Time To Failure, where reliability (derived from Eq. (6) or (7)) is estimated by $R_q = 1 - \prod_{p\in M_p} Q_f(p)$, and $N_q = \frac{E_{Tot}}{E_q}$ is the expected number of queries that can be handled by the system given the total energy E_{Tot} and average energy consumption per query $E_q = E_{rd} + E_{nrd}$ (see Eq. (9)). Equation (20) is derived from an equivalent form of:

$$MTTF = \sum_{i=1}^{N_q-1} i * \left({R_q}^i\right) * \left(1-R_q\right) + N_q R_q^{N_q}$$

The term $i * \left({R_q}^i\right) * \left(1-R_q\right)$ in the above equivalent MTTF formulation accounts for the probability of the system being able to successfully execute i consecutive queries but failing the $(i+1)^{th}$ query. The second term $N_q R_q^{N_q}$ is for the best case in which all queries are processed successfully without experiencing any failure for which the system will have the longest lifetime span.

The constraints are given by:

$$R_q \geq R_{req} \tag{21}$$

where R_{req} is a constant representing the minimum acceptable reliability level for the system.

$$|M_p| = \sum_{p\in P} m(p) \geq 1 \tag{22}$$

$$m(p) = 0 \; or \; 1 \tag{23}$$

where M_p is the set of active paths of the system, and $m(p)$ is a binary decision variable such that if $m(p_i) = 1$ then $p_i \in M_p$. The constraint in Eq. (22) ensures that one path is always and at least selected, to avoid division by zero when calculating N_q.

Thus the system lifetime $MTTF$ is dependent on the expected number of queries N_q and the reliability provided for the queries R_q based on the number of M_p paths chosen, where N_q and R_q are dependent on the energy model and reliability model respectively.

V. PERFORMANCE EVALUATION

In this section, we present numerical results to evaluate the INLP optimization model for both TSCH and SigFox using the mathematical programming tool, GAMS [50]. Table II and Table III show the main parameter values when running the model for TSCH and SigFox, respectively. For both TSCH and SigFox, we consider nodes with 13500 Joules which correspond to two AAA batteries. For a TSCH, other state parameters, including the time slot durations and timing constants, are used as in the OpenWSN MAC layer implementation of IEEE 802.15.4e TSCH [12, 36]. We consider the consumption values of using a CC2538 2.4 GHz radio for TSCH nodes as adopted and validated in [36]. The SigFox parameters are derived from the protocol specification as found in [14, 15].

TABLE II
PARAMETERS AND THEIR VALUES FOR THE TSCH PERFORMANCE EVALUATION

Name	Value	Name	Value
$Area$	1km × 1km	MTU	127 Bytes
n	50	SF_{size}	100
R_{req}	[0.9, 0.9999999]	$S_{duration}$	15 ms
Pf_{hop}	[0.1, 0.00001]	$SF_{duration}$	1500 ms
E_0	13.5 KJ	t_{Δ}	1500 ms
r	[70m, 200m]	n_j	$[1, r \times \frac{n}{Area}]$

TABLE III
PARAMETERS AND THEIR VALUES FOR THE SIGFOX PERFORMANCE EVALUATION

Name	Value	Name	Value
$Area$	10km × 10km	MTU	12 Bytes
n	10	N_{msg}	140 / day
R_{req}	[0.9, 0.9999999]	t_{msg}	2.08 sec
Pf_{hop}	[0.1, 0.00001]	N_{trans}	3
E_0	13.5 KJ	t_{Δ}	$(3600 \text{ sec} \times 24)/N_{msg}$

Figs. 10 and 11 show the optimal number of paths M_p required to maximize $MTTF$ while satisfying the reliability constraints of the query message by running the model using the TSCH and SigFox parameters and their values in Tables II and III, respectively. In both Figs. 10 and 11, we first observe that for a given single hop failure probability, $Pf_{hop,}$ there exists an optimal M_p.

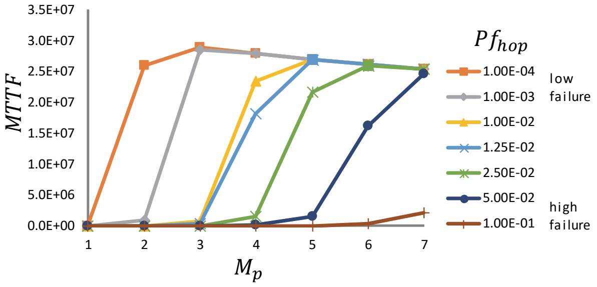


FIGURE 10. MTTF vs. M_p with varying single hop failure for TSCH.

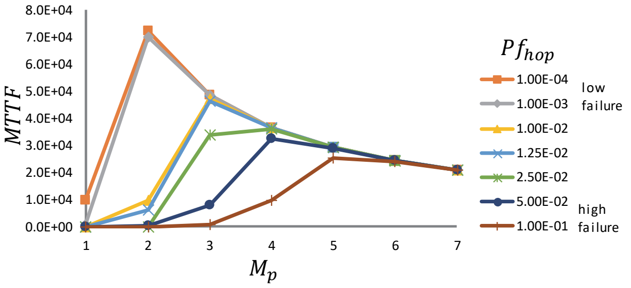


FIGURE 11. MTTF vs. M_p with varying single hop failure for SigFox

We also find that as Pf_{hop} increases, the optimal M_p increases. If $M_p{}'$ is chosen such that $M_p{}'$ < optimal M_p, then the resulting $MTTF$ using $M_p{}'$ denoted by $MTTF(M_p')$ is < $MTTF(\text{optimal}\, M_p)$ because using a lower M_p than the optimal M_p results in lower $MTTF$ caused by lowering the reliability of the query. Furthermore, using $M_p{}'$ > optimal M_p similarly leads to $MTTF(M_p{}'\ ) < MTTF(\text{optimal}\, M_p)$ because of the wasted energy of the TSCH and SigFox networks that use more redundant paths.

Fig. 12 shows the results of running the model for a SigFox using the parameters in Table III. We find that the optimal M_p(denoted by M_p^{opt}) increases as the single hop long-range failure probability Pf_{hop} increases. This is to cope with environmental effects, compromise of attackers, and hardware failure as sensed and shared by the IoT overlay. A similar trend can be observed for TSCH although we omit the discussion for brevity. Fig. 13 shows the effect of changing the reliability constraint R_{req} on the optimal decision variable M_p^{opt} and the resulting $MTTF$. Notice that R_{req} starts to affect the optimal M_p only after it is greater than a certain reliability value. This is because the reliability constraint is a non-binding constraint for lower values. Once it becomes binding, it results in lowering the $MTTF$ value while M_p is non-decreasing in R_{req}. This further can be more clearly shown in Table IV. We observe that higher Pf_{hop} values result in the constraint taking effect and changing M_p^{opt} at lower R_{req} values.

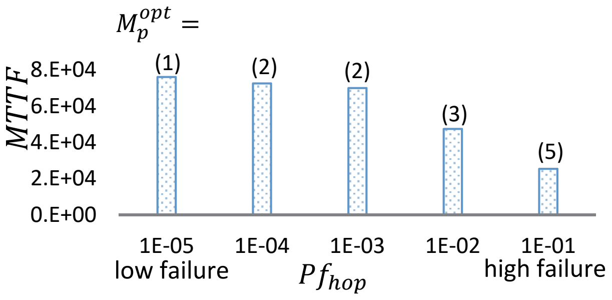


FIGURE 12. Optimal decision variable M_p^{opt} for varying the single hop long-range failure Pf_{hop} for SigFox.

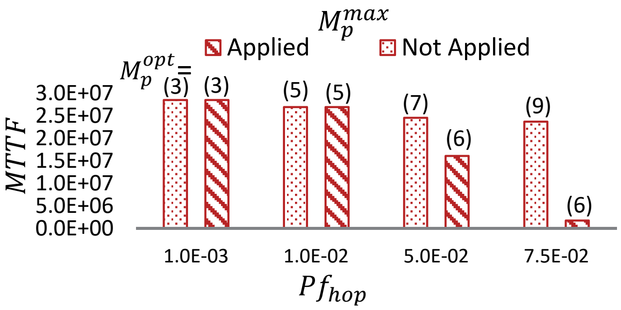


FIGURE 13. Effect of R_{req} on the optimal decision variable M_p^{opt} and resulting MTTF for SigFox.

We note here that Pf_{hop} in SigFox represents the failure probability of the total single hop link between the device and the gateway, whereas Pf_{hop} in the TSCH represents the failure probability of a single hop along the multihop path between the device and the gateway.

TABLE IV
EFFECT OF R_{req} ON M_p^{opt} FOR A SIGFOX

		R_{req}				
		0.999	0.9999	0.99999	0.999999	0.9999999
Pf_{hop}	0.01	3	3	3	3	4
	0.1	5	5	5	6	7
	0.2	7	7	8	9	10

Unlike SigFox, the TSCH relies on a TSCH schedule to transmit between a single device to one of the n_j neighbors (see Eq. (3)), and thus the resulting optimal redundancy is also a factor of the available neighboring nodes based on the density of the deployment, $\frac{n}{Area}$, and assigned schedule all of which effect R_q in the objective function. There are cases in which limiting the chosen M_p to a maximum number is required by the used technology or by the application requirements. In order to restrict the number of reporting nodes to maintain the accuracy of reporting about x, a constraint could be added. Given a deployment density of λ and a geo-casted circular area of $A = \pi(r_g)^2$ where r_g is the radius, we set $M_p^{max} = \lambda \times A$ to determine the maximum allowable number of reporting sources, and then the constraint is:

$$|M_p| = \sum_{p\in P} m(p) \leq M_p^{max} \quad (24)$$

This can also be enforced if the mesh network, for example, can handle at most M_p^{max} paths due to protocol, density, or connectivity constraints.

Fig. 14 compares the optimal decision variable M_p^{opt} for a TSCH in the original INLP vs. in a max-path enforced INLP with varying Pf_{hop} values and M_p^{max}= 6. As the single hop failure probability Pf_{hop} increases, M_p^{opt} increases while $MTTF$ decreases for the original INLP. However, for the max-path enforced INLP, for higher Pf_{hop} values, the constraint in

Eq. (24) becomes a binding constraint and $M_p^{opt} = M_p^{max}$= 6. As a result, *MTTF* decreases rapidly, compared to the original case. Therefore, under limited path choices for an area, and for relatively high single hop failure circumstances, *MTTF* could severely be compromised for a given R_{req} level. To ascertain the correctness of the results of our INLP model, we compare the resulting optimal MTTF (under M_p^{opt})with the maximum MTTF from an exhaustive search (ES) computational procedure enumerating over all possible decision variable values and logging the resulting MTTF. Figs. 15 and 16 show this comparison for the TSCH and SigFox, respectively.

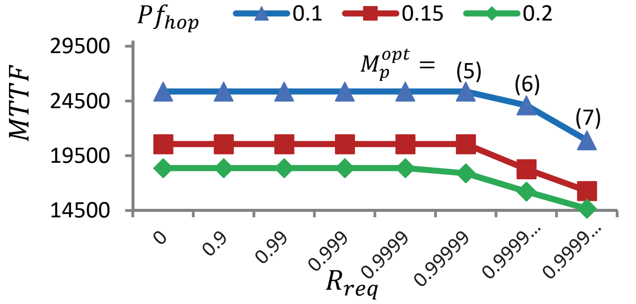


FIGURE 14. MTTF and optimal decision variable $M_p^{opt}$for varying the single hop failure probability Pf_{hop} for a TSCH in the original INLP vs. in a max-path enforced INLP when adding the constraint in Eq. (24) to the INLP model, assuming M_p^{max}= 6.

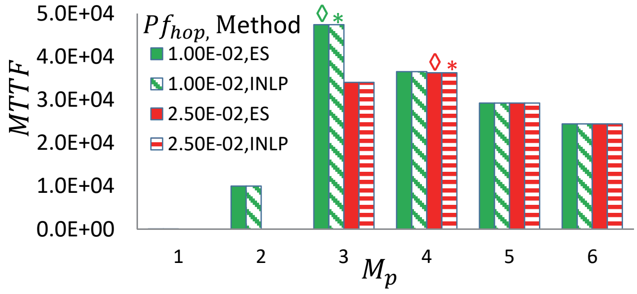


FIGURE 15. MTTF vs. M_p for TSCH under exhaustive search (ES) vs. INLP with varying Pf_{hop}. The optimal MTTF found by INLP through $M_p^{opt}$and the maximum MTTF found by ES through exhaustive search are marked by symbols * and ◊, respectively.

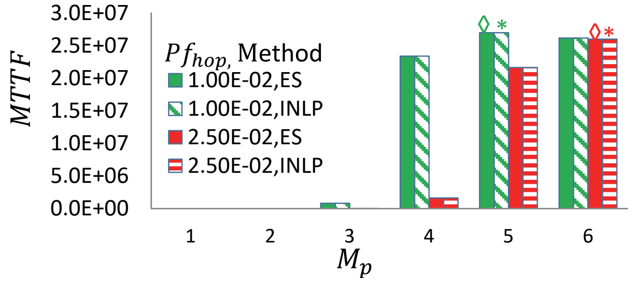


FIGURE 16. MTTF vs. M_p for SigFox under exhaustive search (ES) vs. INLP with varying Pf_{hop}. The optimal MTTF found by INLP through $M_p^{opt}$and the maximum MTTF found by ES through exhaustive search are marked by symbols * and ◊, respectively.

The result of running the INLP model identifies the optimal MTTF through the optimal decision variable M_p^{opt} whereas all other MTTF values (non-optimal) are obtained from ES for the analysis and comparison purposes. Notice that the INLP model finds M_p^{opt} (columns denoted by * in Figs. 15 and 16), which matches the best M_p value found by ES, showing the maximum MTTF (columns denoted by ◊ in Figs. 15 and 16).

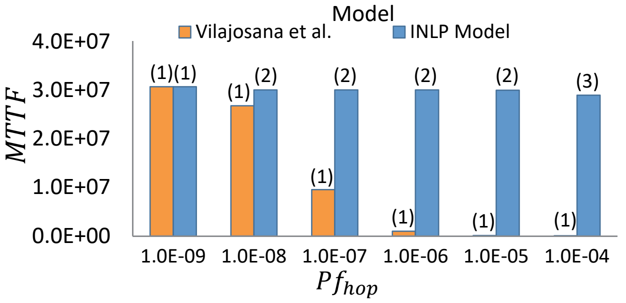


FIGURE 17. Comparing MTTF in a baseline TSCH model, Vilajosana et al., using fixed single path M_p= 1 with INLP model using the optimal decision variable $M_p^{opt}$for varying single hop failure Pf_{hop}.

For TSCH, Fig. 17 shows the performance comparison results in terms of the IoT network lifetime obtained under our INLP model with that under a baseline energy model introduced by Vilajosana et al. [13] which does not consider finding the best redundancy level based on the reliability of the deployed environment. We run both models using the same set of parameter values listed in Table II with both models consuming energy based on the same CC2538 2.4 GHz radio hardware [36]. Since both models follow IEEE 802.15.4e TSCH, initially they behave the same under identical consumption specifications given the same TSCH composition of time slots.

When the single hop failure probability $Pf_{hop}$is low (e.g. 1.0E-09), our INLP model chooses a low $M_p$value to reduce energy consumption (M_p^{opt}=1) which happens to be the default value chosen by the baseline model as it does not consider the single hop failure probability in the deployed environment (always sets $M_p = 1$). As the single hop failure increases, our INLP model increases M_p^{opt} to maximize the message reliability to the gateway, while the baseline model continues to use the default value of 1 for M_p^{opt}, resulting in a lower lifetime. We conclude that our INLP model outperforms the baseline energy model introduced by Vilajosana et al. [13]. We attribute the superior performance of our model to its ability of identifying M_p^{opt} that maximizes the message reliability in response to changes in the single hop failure probability, Pf_{hop}.

For SigFox, Fig. 18 shows the performance comparison results of our INLP model with that under a baseline energy model by Morin et al. [16]. Like their model, we only focus on Tx and $Sleep$ energy consumption. We run both models using the same set of parameter values of Table III. For fair comparison, we disregard the energy leakage of 5% per year in their model as well as the cutoff voltage when the residual energy reaches 10% of E_0 [16]. We observe identical energy consumption of a single SigFox message due to similar configuration. More importantly, the SigFox comparison results exhibit the same trend. That is, our INLP SigFox model produces a higher system lifetime than [16] as the Pf_{hop} increases, due to its ability to identify M_p^{opt} that maximizes the system lifetime in response to changes in Pf_{hop}, in the deployed area at runtime.

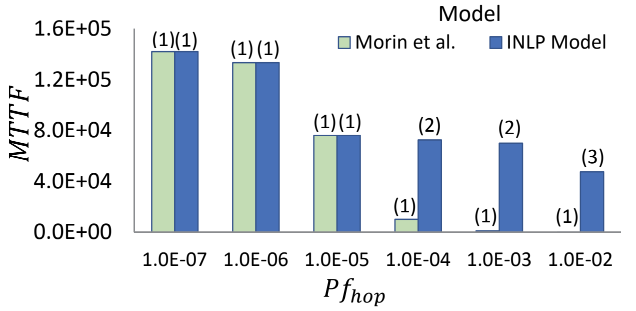


FIGURE 18. **Comparing MTTF in a baseline SigFox model derived from Morin et al. using fixed single path M_p= 1 with INLP model using the optimal decision variable $M_p^{opt}$for varying Pf_{hop}.**

VI. CONCLUSION

In this paper, we proposed and analyzed an INLP optimization model to maximize the lifetime of IoT-Based LPWAN and mesh networks in unreliable environments. We demonstrated the feasibility of our approach in optimizing lifetime while satisfying protocol and application constraints. SigFox and TSCH were used as representative IoT technologies for LPWAN and mesh networks, respectively. As our future research, we plan to expand this model to other IoT technologies, such as LoRa [46] and cellular-based networks [28], and further expand the model to consider different topology, device, and gateway settings. Like [17], we further plan to explore more sophisticated attacks.

REFERENCES

[1] Y. Kabalci, "A survey on smart metering and smart grid communication," *Renewable and Sustainable Energy Reviews,* vol. 57, pp. 302-318, 2016.

[2] S. Ali *et al.*, "Wide area smart grid architectural model and control: A survey," *Renewable and Sustainable Energy Reviews,* vol. 64, pp. 311-328, 2016.

[3] T. M. AS. *M2M Multi Hop Wireless Mesh Network that is Easier and Smarter*. [Online]. Available: https://tiny-mesh.com/wireless-mesh-network. Accessed Dec. 31, 2017.

[4] VERICOM. [Online]. Available: https://www.vericomsolutions.com. Accessed July 27, 2018.

[5] T. Watteyne, L. Doherty, J. Simon, and K. Pister, "Technical overview of SmartMesh IP," in *IEEE Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, 2013, pp. 547-551.

[6] K. Bannister, G. Giorgetti, and S. Gupta, "Wireless sensor networking for hot applications: Effects of temperature on signal strength, data collection and localization," in *Proceedings of the 5th Workshop on Embedded Networked Sensors (HotEmNets' 08)*, 2008: Citeseer.

[7] H. Wennerström, "Meteorological impact and transmission errors in outdoor wireless sensor networks," Ph.D dissertation, Department of Information Technology, Uppsala University, 2013.

[8] P. Wang, Z. Sun, M. C. Vuran, M. Al-Rodhaan, A. Al-Dhelaan, and I. F. Akyildiz, "Topology analysis of wireless sensor networks for sandstorm monitoring," in *2011 IEEE International Conference on Communications (ICC)*, 2011, pp. 1-5.

[9] H. M. Mujlid, "Real-time monitoring of sand and dust storm winds using wireless sensor technology," Ph.D dissertation, Department of Electical and Computer Engineering, Florida Institute of Technology, 2016.

[10] S. Arnon, "Effects of atmospheric turbulence and building sway on optical wireless-communication systems," *Optics letters,* vol. 28, no. 2, pp. 129-131, 2003.

[11] B. P. Wong and B. Kerkez, "Real-time environmental sensor data: An application to water quality using web services," *Environmental Modelling & Software,* vol. 84, pp. 505-517, 2016.

[12] T. Watteyne, M. Palattella, and L. Grieco, "Using IEEE 802.15. 4e time-slotted channel hopping (TSCH) in the internet of things (IoT): Problem statement," in " No. RFC 7554," No. RFC 7554. 2070-1721, 2015.

[13] X. Vilajosana, Q. Wang, F. Chraim, T. Watteyne, T. Chang, and K. S. Pister, "A realistic energy consumption model for TSCH networks," *IEEE Sensors Journal,* vol. 14, no. 2, pp. 482-489, 2014.

[14] J. C. Zuniga and B. Ponsard, "Sigfox system description," *LPWAN@ IETF97, Nov. 14th,* 2016.

[15] SIGFOX. [Online]. Available: http://www.sigfox.com/en/. Accessed July 18, 2018.

[16] E. Morin, M. Maman, R. Guizzetti, and A. Duda, "Comparison of the device lifetime in wireless networks for the internet of things," *IEEE Access,* vol. 5, pp. 7097-7114, 2017.

[17] A. Ashok, A. Hahn, and M. Govindarasu, "Cyber-physical security of Wide-Area Monitoring, Protection and Control in a smart grid environment," *Journal of advanced research,* vol. 5, no. 4, pp. 481-489, 2014.

[18] P. T. Myrda and K. Koellner, "Naspinet-the internet for synchrophasors," in *IEEE 43rd Hawaii International Conference on System Sciences (HICSS)*, 2010, pp. 1-6.

[19] Z. R. Ma, "An Agricultural Habitat Information Acquisition and Remote Intelligent Decision System Based on the Internet of Things," in *Computer and Computing Technologies in Agriculture XI: 11th IFIP WG 5.14 International Conference, CCTA 2017, Jilin, China, August 12-15, 2017, Proceedings*, 2019, vol. 546, p. 75: Springer.

[20] S. Rezwan *et al.*, "A Minimalist Model of IoT based Sensor System for Sewage Treatment Plant Monitoring," in *IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2019, pp. 0939-0945: IEEE.

[21] B. P. Gautam, K. Wasaki, and N. Sharma, "A Novel Approach of Fault Management and Restoration of Network Services in IoT Cluster to Ensure Disaster Readiness," in *IEEE International Conference on Networking and Network Applications (NaNA)*, 2016, pp. 423-428.

[22] H. Al-Hamadi and I. R. Chen, "Redundancy Management of Multipath Routing for Intrusion Tolerance in Heterogeneous Wireless Sensor Networks," *IEEE Transactions on Network and Service Management,* vol. 10, no. 2, pp. 189-203, 2013.

[23] F. Al-Turjman, "Cognitive routing protocol for disaster-inspired internet of things," *Future Generation Computer Systems,* vol. 92, pp. 1103-1115, 2019.

[24] S. Borkotoky, C. Bettstetter, U. Schilcher, and C. Raffelsberger, "Allocation of Repetition Redundancy in LoRa," *arXiv preprint arXiv:1904.06072,* 2019.

[25] K. Jaiswal and V. Anand, "EOMR: An Energy-Efficient Optimal Multi-path Routing Protocol to Improve QoS in Wireless Sensor Network for IoT Applications," *Wireless Personal Communications,* pp. 1-23, 2019.

[26] M. Ortiz, Y. Sun, G. S. Young, and Q. Sun, "An Redundant Networking Channel to Support Reliable Communications in the Internet of Things Applications," in *International Conference on Machine Learning and Intelligent Communications*, 2016, pp. 283-292: Springer.

[27] M. Z. Hasan and F. Al-Turjman, "Optimizing multipath routing with guaranteed fault tolerance in Internet of Things," *IEEE Sensors Journal,* vol. 17, no. 19, pp. 6463-6473, 2017.

[28] A. Azari and G. Miao, "Network lifetime maximization for cellular-based M2M networks," *IEEE Access,* vol. 5, pp. 18927-18940, 2017.

[29] X. Liu *et al.*, "Adaptive data and verified message disjoint security routing for gathering big data in energy harvesting networks," *Journal of Parallel and Distributed Computing,* vol. 135, pp. 140-155, 2020.

[30] A. Mahmood, M. A. Hossain, C. Cavdar, and M. Gidlund, "Energy-Reliability Aware Link Optimization for Battery-Powered IoT Devices With Nonideal Power Amplifiers," *IEEE Internet of Things Journal,* vol. 6, no. 3, pp. 5058-5067, 2019.

[31] B. Mostafa, A. Benslimane, M. Saleh, S. Kassem, and M. Molnar, "An energy-efficient multiobjective scheduling model for monitoring in internet of things," *IEEE Internet of Things Journal,* vol. 5, no. 3, pp. 1727-1738, 2018.

[32] F. De Rango, D. Barletta, and A. Imbrogno, "Energy aware communication between smart IoT monitoring devices," in *IEEE International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2016, pp. 1-7.

[33] A. F. Santamaria, F. De Rango, D. Falbo, and D. Barletta, "SmartHome: a domotic framework based on smart sensing and actuator network to reduce energy wastes," in *Wireless Sensing, Localization, and Processing IX*, 2014, vol. 9103, p. 910308: International Society for Optics and Photonics.

[34] F. Al-Turjman and M. Abujubbeh, "IoT-enabled smart grid via SM: An overview," *Future Generation Computer Systems,* vol. 96, pp. 579-590, 2019.

[35] B. Martinez, M. Monton, I. Vilajosana, and J. D. Prades, "The power of models: Modeling power consumption for IoT devices," *IEEE Sensors Journal,* vol. 15, no. 10, pp. 5777-5789, 2015.

[36] G. Daneels *et al.*, "Accurate Energy Consumption Modeling of IEEE 802.15. 4e TSCH Using Dual-Band OpenMote Hardware," *Sensors,* vol. 18, no. 2, p. 437, 2018.

[37] Xively. *IoT Platform for Connected Devices – Xively*. [Online]. Available: https://xively.com. Accessed Jan. 6, 2019.

[38] TheMathWorksInc. *Learn More - ThingSpeak IoT*. [Online]. Available: https://thingspeak.com. Accessed Jan. 6, 2019.

[39] ThingWorx. *ThingWorx Industrial IoT*. [Online]. Available: https://www.ptc.com/en/products/iot/thingworx-platform. Accessed Jan. 6, 2019.

[40] GoogleCloudPlatform. [Online]. Available: https://cloud.google.com/solutions/iot. Accessed Jan. 6, 2019.

[41] USGS. *Data and Tools*. [Online]. Available: https://www.usgs.gov/products/data-and-tools/real-time-data/. Accessed Jan. 6, 2019.

[42] WeatherUnderground. *250,000+ Weather Stations*. [Online]. Available: https://www.wunderground.com/weatherstation/overview.asp. Accessed Jan. 6, 2019.

[43] J. A. W. Maghanoy, "Crime mapping report mobile application using GIS," in *2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP)* 2017, pp. 247-251: IEEE.

[44] N. Mahmud, K. I. Zinnah, Y. A. Rahman, and N. Ahmed, "Crimecast: A crime prediction and strategy direction service," in *2016 19th International Conference on Computer and Information Technology (ICCIT)*, 2016, pp. 414-418: IEEE.

[45] HalifaxRegionalMunicipality. *Halifax crime mapping*. [Online]. Available: https://www.halifax.ca/fire-police/police/crime-mapping. Accessed Jan. 6, 2019.

[46] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: Long range & low power networks for the internet of things," *Sensors,* vol. 16, no. 9, p. 1466, 2016.

[47] M. Lauridsen, B. Vejlgaard, I. Z. Kovacs, H. Nguyen, and P. Mogensen, "Interference measurements in the European 868 MHz ISM band with focus on LoRa and SigFox," in *IEEE Wireless Communications and Networking Conference (WCNC) 2017*, pp. 1-6.

[48] R. A. Sahner, K. Trivedi, and A. Puliafito, *Performance and reliability analysis of computer systems: an example-based approach using the SHARPE software package*. Springer Science & Business Media, 2012.

[49] N. Baccour *et al.*, "Radio link quality estimation in wireless sensor networks: A survey," *ACM Transactions on Sensor Networks (TOSN),* vol. 8, no. 4, p. 34, 2012.

[50] GAMS. [Online]. Available: http://www.gams.com. Accessed July 27, 2018.