IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 2, 2020, accepted October 10, 2020, date of publication October 19, 2020, date of current version October 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3031974

Unsupervised Learning Clustering and Dynamic Transmission Scheduling for Efficient Dense LoRaWAN Networks

MOHAMMED ALENEZI[1], KOK KEONG CHAI[1], ATM S. ALAM[1], YUE CHEN[1], (Senior Member, IEEE), AND SHIHAB JIMAA[2]

[1]School of Electrical Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K.
[2]Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi, United Arab Emirates

Corresponding author: Mohammed Alenezi (m.alenezi@qmul.ac.uk)

This work was supported by the Queen Mary University of London.

ABSTRACT Long-Range (LoRa) communication technology is considered as a promising connectivity solutions for Internet of Things (IoT) dense applications. In particular, LoRa has drawn the interest due to its low power consumption and wide area coverage. Despite the benefits of LoRaWAN protocol, it still suffers from excessive random and simultaneous transmissions due to the adoption of ALOHA protocol. Therefore, resulting in severe packet collision rate as the network scales up. This leads to continuous retransmission attempts, which in return increase the transmission delay and energy consumption. Thus, this paper proposes a dynamic transmission Priority Scheduling Technique (*PST*) based on the unsupervised learning clustering algorithm to reduce the packet collision rate and enhance the network's transmission delay and energy consumption. Particularly, the LoRa gateway classifies the nodes into different transmission priority clusters. While the dynamic *PST* allows the gateway to configure the transmission intervals for the nodes according to the transmission priorities of the corresponding clusters. This work allows scaling up the network density while maintaining low packet collision rate and significantly enhances the transmission delay & the energy consumption. Simulation results show that the proposed work outperforms the typical LoRaWAN and recent clustering & scheduling schemes. Therefore, the proposed work is well suited for dense applications in LoRaWAN.

INDEX TERMS Unsupervised clustering, collision rate, energy consumption, IoT, LoRa, Naive Bayes classifier, packet delivery rate, priority scheduling, transmission delay.

I. INTRODUCTION

The Low-Power Wide Area Networks (LPWAN) technologies have been increasingly researched and deployed as a promising solution for serving Internet of Things (IoT) applications. Long-Range (LoRa) technology via its LoRaWAN protocol [1], has shown a very attractive platform due to its low energy consumption and wide area coverage. However, one main drawback associated with LoRaWAN is the vulnerability to high packet collision rate. This is due to the adaption of ALOHA communication protocol, where LoRa nodes initiate packet transmissions without the presence of Listen Before Talk (*LBT*) protocol [2], [3]. As a result, LoRaWAN efficiency suffers a depreciation, particularly on network's energy consumption and transmission delay. In order to compensate for the absence of LBT protocol, LoRaWAN provides different Spreading Factors (*SF*) based on the LoRa physical layer Chirp Spread Spectrum (*CSS*) technique to allow simultaneous packet transmissions. Alternating between different *SF* comes at the expense of higher transmission power and time-on-air, which can be ideal solution for small-scale networks [4]–[7]. However, adapting LoRaWAN to serve dense applications remains an open challenge.

LoRa physical layer modulations relies on the *CSS* technique [8], which quantifies how many chirps are pulsed per second. Using *CSS* technique, LoRa provides a wide area communication coverage for a range of more than 10 *km*. In addition, using *CSS* increases the robustness against noise and external interferences. The Medium Access Control (MAC) LoRaWAN protocol exploits the *CSS* by providing the *SF* feature to further boost the communication efficiency. The transmissions using different *SF*, between *SF7* and *SF12*, vary in terms of data per chirp per second [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Nabil Benamar.

This allows the receiver to distinguish between simultaneous transmissions according to the used *SF* [9]. The packet transmission delay is the duration of transmitting a packet from the sender to the receiver. In an ideal environment, the packet transmission delay is mainly effected by the *SF*, the transmission power, and the packet size [10]. Given a typical IoT packet size of 250-300 bytes, an IoT battery-powered device using LoRaWAN has an expected lifetime of up to 6 years [11], provided infrequent daily transmissions [12], [13]. This explains the wide interest of adapting LoRaWAN in IoT applications. However, as formerly mentioned, the LoRaWAN efficiency is still an open challenge especially in dense applications.

This led to the strive of a number of research bodies and industrial organisations to challenge the efficiency of LoRaWAN. For example, Rachkidy *et al.* [14], proposed a collision resolution technique that allows LoRa gateway to decode the collided and hence, corrupted received packets. While Liao *et al.* [15], introduced a multi-hop based concurrent transmission technique in order to mitigate the probability of simultaneous packets transmissions of LoRa nodes. In addition, Zhu *et al.* [16], proposed a tree based clustering algorithm to enhance LoRaWAN capacity. Particularly, their scheme exploits the variety in *SF* communication reliability by allocating different *SF* to different clusters. Based on that, clusters with less *SF* reliability off-load traffic to clusters with higher *SF* reliability via multi-hop relay. Although, the collision rate has been enhanced in the aforementioned schemes [14]–[16], this comes at the expense of compromising the transmission delay or the energy consumption. Hence, in this paper, the aim of the work is to reduce the packet collision rate in LoRaWAN while maintaining relatively low transmission delay and energy consumption.

In order to achieve an efficient implementation of LoRaWAN to serve dense IoT applications, it is necessary to define the target application. This work considers using LoRaWAN as wireless communication solution for serving an early warning weather monitoring system. Given the unlabeled data delivered by the nodes (sensors), the limited resources for the devices (battery-powered) and the random transmission behaviour of LoRaWAN due to adapting ALOHA protocol, this paper introduces the use of unsupervised learning clustering algorithm (K-Means) as a base for the dynamic transmission Priority Scheduling Technique (*PST*). Unlike other machine learning techniques, K-Means is specifically chosen for its simplicity in partitioning the unlabeled data delivered by the nodes into different clusters [17]. When applying K-Means at the gateway level, the nodes with similar data are grouped in a cluster without the need to nodes coordination in the partitioning process. This lifts the computation burdens from the nodes level to the gateway level. Hence, avoiding excessive computation overhead at the nodes level that can jeopardise the energy efficiency.

Based on K-Means clustering, the dynamic *PST* allows the LoRa gateway to configure the transmission intervals for nodes located at different clusters to prevent the vast amount of simultaneous transmissions. Hence, achieving lower packet collision rate. The configuration process of transmission intervals is based on prioritising nodes in a certain cluster according to a set of application related parameters (e.g. a packet containing weather temperature readings) received from the node to the gateway. Also, unlike the proposed work in [16], where nodes off-load traffic to neighbouring clusters, the dynamic *PST* avoids the excessive energy burden associated with multi-hopping techniques by maintaining the original LoRaWAN star topology. Simulations is carried out to evaluate the impact of reducing the total collision rate on both the Total Transmission Delay (*TTD*) and the Total Energy Consumption (*TEC*). Furthermore, different from the previously proposed static transmission scheduling in [18], the dynamic *PST* trades-off the Packet Delivery Rate (*PDR*) to achieve better *TTD* and *TEC*.

The scope of the paper is to address the feasibility of using LoRaWAN to serve a dense application. The main challenges are the excessive packet collision rate, inefficient *TTD*, *TEC* and *PDR*. In order to address these challenges, a dense network system model is designed, where up to 1000 nodes are distributed randomly within a limited area of up to 3 km^2 around one LoRa gateway. In addition, the nodes are set to communicate with the gateway via Class *A* end-device LoRaWAN[1] using *SF7*. Note that these are the most reliable parameters provided by LoRaWAN in terms of energy efficiency. An evaluation of the severe total collision rate in a typical LoRaWAN network is obtained via simulations analysis. Therefore, K-Means clustering algorithm is introduced to reduce the probability of packet collision rate caused by the randomness and excessive simultaneous transmissions from the nodes. Furthermore, to obtain the optimal number of clusters, simulations analysis is carried out to evaluate the *TTD* and *TEC* efficiency under different number of clusters. To prove the performance efficiency, simulations are carried out to evaluate the impact of the proposed work against recent clustering and scheduling techniques for LoRaWAN in terms of *TTD*, *TEC* and *PDR*.

The main contributions of this work are summarised as following:

1) Classifying the unpredicted transmissions' nature of LoRaWAN into an organised manner that allows better resource management. This is achieved by exploiting the unsupervised learning clustering algorithm. In return, this reflects the magnificent reduction to the excessive collision rate associated with typical LoRaWAN.

[1]Note that LoRaWAN provides three different classes for the end-device to join the network. First is Class *A*, which is the most energy efficient, where the nodes initiate transmissions without prior sensing to the channel status and open a temporary receive window following each transmission. Second is Class *B*, where nodes listen to periodic beacons from the gateway. Third is Class C, which is the most energy inefficient, where nodes listen continuously to the gateway. More details of LoRaWAN end-device classes are available in [19].

2) Proposing a dynamic transmission *PST* based on the unsupervised learning clustering algorithm to further enhance the network's *TTD* and *TEC* while maintaining relatively acceptable *PDR* in comparison to other techniques. This is performed in two folds:
    a) First, the gateway allocates unique transmission priority to each of the clusters in the network ranging from high to low. Based on the cluster's priority, the corresponding nodes are assigned specific transmission intervals by the gateway.
    b) Second, the dynamic *PST* provides two transmission modes; conservative (*con.*) and non-conservative *(ncon.)*. In the *con.* mode, the *PDR* is elevated at the expense of relatively higher *TTD* and *TEC*. While the *ncon.* mode provides more efficient *TTD* and *TEC* at the expense of relatively lower *PDR*. The dynamic *PST* applies the Naive Bayes classifier algorithm in order to determine the probability of assigning a specific transmission mode to each cluster.

The rest of the paper is organised as follows: Section II reviews related work carried out for the purpose of enhancing the energy efficiency and the transmission delay in LoRaWAN. Section III reveals the system model, the problem definition and formulation. Section IV exposes the proposed dynamic *PST*. Section V discusses the simulation results. This paper is then concluded in section VI.

II. RELATED WORK

This section sheds light on a set of studies that were carried out to enhance the energy efficiency and the transmission delay in low power networks. It also explores the use of various machine learning techniques, all for the purpose of enhancing LoRaWAN performance. The section concludes with identifying the technical concerns that are bridged by the proposed unsupervised learning based dynamic *PST*.

A. ENERGY EFFICIENCY

Energy efficiency has always been within the interest scope of researchers, especially with technologies designed for serving IoT applications. In this regard, Kavitha and Suseendran in [20] propose a priority based adaptive scheduling algorithm for IoT sensor systems where several performance aspects were taken into consideration. One main issue the proposed algorithm tackles is the energy consumption in wireless sensor networks. The scheduling algorithm is based on preset delay and energy requirements. Based on these requirements a given packet can only transmit when there is a free slot for transmission. In particular, the algorithm introduces a queuing procedure where packets queue before initiating transmissions. This procedure is mainly utilised to reduce the amount of transmissions and hence reduce the total energy consumption of the network. Their work is also inspired by similar techniques presented in [21]–[24].

Rubel *et al.* [25] propose a clustering based priority management scheme to reduce the overall energy consumption in a wireless sensor network. In specific the scheme classifies data received from nodes in different delay requirements. Based on each classification the scheme allows sensors to initiate communication with the base station. Their scheme trades off the quality-of-service in serving each class of nodes.

In regards to the energy efficiency of IoT devices, Ogundile *et al.* [26] investigate the energy consumption constraint in wireless sensor networks and propose a clustering based routing algorithm. Their work takes into consideration that in some scenarios the sensors tend to consume more power attempting to initiate transmissions with the base station. Therefore, they proposed a clustering algorithm where each group of nodes utilise predefined nodes (cluster heads) within their cluster to reach the base station. This multi-hop technique showed an improvement to the energy efficiency. However, such an approach can negatively impact the overall performance in the case of busy multi-hops. Another common problem associated with the rerouting approaches is that the nodes selected for multi-hoping are vulnerable to having a short lifetime. Hence, the reliability is still an open issue. Similar approaches are also adapted in [27]–[29].

B. TRANSMISSION DELAY

In line of the clustering based routing algorithms, Liu and Chang [30] propose a clustering rerouting scheme based on the assumption that nodes are scattered unevenly. Their assumption results on their clustering being performed based on an unequal number of nodes in each cluster. They utilise a probabilistic model in order to determine which node has the energy capability to perform the tasks of multi-hoping for other nodes in need within the same cluster. The node then has the ability to opt out from serving as a cluster head when the energy level reaches a specified threshold. Although the proposed scheme outperforms those in [31]–[33], still it has a negative impact on the total network delay.

As the functionality of LoRaWAN is highly dependent on resource allocation the technology uses an Adaptive Data Rate (*ADR*) technique. In practice, *ADR* aims to achieve right first time reception between end-device and the gateway through basic minimum *SF* selection. However, Cuomo *et al.* [34] recognised two sophisticated *SF* allocation algorithms, *EXPLoRa-SF* and *EXPLoRa-TA*. The algorithms show a reduction in interference between clusters of end-devices with varying *SF* through improved time-on-air. More specifically, *EXPLoRa-SF* attempts to equally assign redundantly high *SF* groups across multiple base stations that are restricted solely by their Received Signal Strength Indicator (*RSSI*). Although high *SF* provide long-range coverage, they increase interference and collisions through greater time-on-air. Hence, *EXPLoRa-TA* works by assigning different *SF* to end-device groups to ensure each group has an equal amount of time-on-air. They coined the term ''ordered water-filing''. It was observed that both algorithms prevailed over *ADR* at improving throughput in highly loaded systems of end-devices distributed 200 meters from the gateway.

LoRaWAN has three classes of communication, Class *A*, *B* and *C*, listed in descending order of energy consumption. Delobel *et al.* [35] select Class *B* to study the energy efficiency of downlink communication (performance) as it is optimised for this purpose. The downlink communication is confirmed through an acknowledgment (*ACK*) mechanism. Failure to receive *ACK* will trigger a retransmission, which accumulates delays. The expected delay time is analytically computed in their proposed Markov chain model. However, it exposes further flaws within the application of Class *B*. The limitations include; the gateway duty-cycle, conflict between Class *A* & *B*, and delay before *ACK* sub-band availability.

The first limitation of duty-cycle is apparent to Delobel *et al.* [35], where the gateway is prevented from sending *ACK* for a large number of confirmed uplinks, for which it has delays of up to 98.13s before the use of the next ping slot could be seen. Nonetheless, they assumed all data frames could be acknowledged by gateways in which all ping slots could be used.

The second limitation in LoRaWAN specifications is the conflict between Class *A* and *B*. Since Class *A* devices transmissions are random, Delobel *et al.* [35] prevented other Class *B* devices from transmissions during designated ping slots from the gateway (beacons). By adapting this approach, their scheme was using Markov chain based model increased the data-rate which results in reduced time-on-air frames. Moreover, the delay time was further improved by increasing the number of sub-bands together with increasing the ping period, which in return allow more frame transmissions and less delays.

C. MACHINE LEARNING IN LoRaWAN

LoRaWAN is meant for serving IoT applications, where low latency is not usually critical requirement [19]. Hence, the simplicity in LoRaWAN protocol, which makes it suitable for IoT serving applications given the limited resources. However, this results in a major drawback in LoRaWAN, which is the severe packet collision rates especially as the served networks scale up. In return, this results in a serious degradation to LoRaWAN performance and thus, to its reliability. The different features LoRaWAN protocol provides e.g. *SF* and Coding Rate (*CR*), enhance its flexibility and suitability for being adapted according to the needs of the served application [1]. In addition, these features encouraged number of research efforts to use them as elements to improve LoRaWAN in different aspects using various machine learning algorithms and techniques.

For example, Cui and Joe [36] have proposed an enhanced packet collision prediction scheme based on Long Short-Term Memory (*LSTM*) model. Despite the high prediction accuracy *LSTM* model provided, LoRaWAN random transmission behaviour requires an online training schemes to achieve a practical prediction process. This has motivated Cui and Joe to combine *LSTM* with a State Space Model (*SSM*) and propose an enhanced Long Short-Term Memory Extended Kalman Filter (*LSTM-EKF*) scheme. Their proposed scheme showed relatively higher prediction accuracy in comparison to the original *LSTM* model. However, the prediction process is highly dependant on the input parameters, which are chosen to include LoRaWAN protocol features such as different *SF*, *CR* and class of end-node communication. Hence, the prediction accuracy remains a function of the pre-inputs selection process leading to a very high computation overhead.

Acknowledging the randomness of LoRaWAN transmissions, Cuomo *et al.* [37] proposed nodes profiling scheme based on the unsupervised learning clustering algorithm (K-Means). Considering two gateways within the proximity of the nodes, the profiling scheme aims to predict the duplication in nodes transmissions by grouping packets that have similar transmission characteristics. Based on the duplication prediction, a traffic prediction is carried out via combining the Decision Tree (*DT*) and *LSTM* models for the purpose of enhancing the resource allocation. Although the unsupervised learning clustering algorithm is preferable machine learning classifying tool in low power networks due to its simplicity, however the number of clusters could play a vital role making it very complex to implement. Hence, analysis to the optimal number of clusters is essential to achieve the optimal clustering accuracy. This is highly dependant on the application parameters used in the clustering process.

Exploiting the variety of LoRaWAN features as parameters in machine learning tools, Sandoval *et al.* [38] proposes a configuration update scheme to the nodes based on Reinforcement Learning (*RL*) to maximise the throughput of each node individually. The configuration process relies on the fact of categorising packets received from the nodes into different importance scales. Their scheme reserves LoRaWAN parameters that ensure robust transmission for nodes classified as important source of information. These nodes receive configuration updates from the gateway to elevate their individual throughput, whereas, the gateway using the *RL*-based scheme retains from updating nodes classified as lower importance source of information. In other words, the gateway using the proposed scheme learns how to intelligently elevates the chance of allowing certain nodes to successfully transmit at the expense of other nodes, all based on prior importance classifications of the nodes.

Similarly, Aihara *et al.* [39] use *RL* by proposing *Q*-learning model combined with Carrier-Sense Multiple Access with Collision Avoidance (*CSMA/CA*) to mitigate the collisions in LoRaWAN and enhance the network *PDR*. In their proposed scheme, the number of successfully received packets form the nodes to the gateway is defined as a reward function that their scheme learns to maximise. They have defined the main cause of the collisions to be simultaneous transmissions. Hence, to mitigate the collision rate, their schemes allows the gateway to learn which nodes are prone to simultaneous transmissions. Hence, the gateway allows the target nodes to transmit over different channels using *CSMA/CA*. However despite the high energy consumption needs when using such techniques, the *PDR* is still a function of the number of available channels. Hence, this can cause negative impact especially when the network scales up.

D. TECHNICAL CONCERNS

The use of machine learning in LoRaWAN is attractive, however it can be easily accompanied with severe negative consequences especially in terms of energy consumption. These consequences are usually due to the very high computation needs which can jeopardise the whole purpose of using LoRaWAN as a low power solution for IoT applications. For example, it was noticed when reviewing the aforementioned set of studies [30]–[39] that the energy consumption was de-prioritised if not completely neglected. The fact that IoT devices are resource limited (e.g. battery-powered) makes it inefficient to use machine learning algorithms that require end nodes coordination. Taking this fact into account and based on the random LoRaWAN transmissions behaviour, this work implements the unsupervised learning clustering algorithm (K-Means) together with the dynamic *PST* at the gateway level. The sole motive behind using K-Means is to partition a given number of nodes into a set of clusters according to their associated unlabeled data, which allows the gateway to implement the dynamic *PST* to configure the nodes with different transmission priorities. These transmission configurations allow the nodes to be aware of the transmission intervals. Hence, the nodes effortlessly transmit in different time intervals decided at the gateway level. This mechanism lifts the computation burdens of finding the transmission intervals from the nodes level to the gateway level, assuming the gateway has sufficient resources. Hence, the chance of packet collisions is reduced due to reducing the number of simultaneous transmission by the nodes. In contrast, achieving higher *PDR* while keeping low *TEC* and *TTD*.

III. SYSTEM MODEL, PROBLEM STATEMENT AND FORMULATION

The system model, the considered dense application scenario and the impact of packet collision rate are revealed in section III-A. This is followed by formulations of the Total Transmission Delay (*TTD*) and Total Energy Consumption (*TEC*) in section III-B. Where section III-C introduces the unsupervised learning clustering algorithm (K-Means) and reveals the optimal number of clusters analysis. The notations used in the rest of the paper are presented in Table 1.

TABLE 1. List of notations.

Notation	Description
A_{C_K}	Average value of A transmitted by all n_i in the corresponding C_K
A_{n_i}, B_{n_i}	A,B values transmitted by n_i
A_{nC_K}	A value transmitted by n_i in the corresponding C_K
B_{nC_K}	B value transmitted by n_i in the corresponding C_K
B_{C_K}	Average value of B transmitted by all n_i in the corresponding C_K
C_{Pr}	Cluster's transmission priority
C_{HPr}	Higher transmission priority cluster
C_{LPr}	Lower transmission priority cluster
C_K	Clusters of $K \in \{1, 2, ..., 5\}$
c_j	Cluster's center point
C_k	Clusters of $k \in \{1, 2, 3, ..., 30\}$
$D_{R_{colli}}$	Daley of n_i's retransmission of collided packets
$D_{R_{ch}}$	Daley of n_i's retransmission of lost packets
D_{I_T}	Daley of n_i's initial packet transmission
$D_{C_{HPr}}$	Delay in of higher priority clusters
$D_{C_{LPr}}$	Delay in of lower priority clusters
D_{active}	Duration of n_i in active mode
D_{idle}	Duration of n_i in idle mode
d	n_i's distance from *GW*
D_{C_K}	Total transmission delay of a cluster of *K*
d_0	Distance from *GW* (500 *meters*)
d_{max}	Maximum distance from *GW* ($3km$)
E_{active}	Energy consumption of active transmitting cluster (*Joules*)
E_{idle}	Energy consumption of idle cluster (*Joules*)
G	Rate of packet transmission attempts per node
GW	LoRa gateway
I_T	Initial packet transmission
K_{opt}	Optimal number of clusters, ($K = 5$)
$n_{C_{HPr}}$	n_i in higher priority clusters
$n_{C_{LPr}}$	n_i in lower priority clusters
$n_{i_{Pr}}$	i^{th} node transmission priority
n_{C_K}	Node corresponding to a cluster of *K*
n_i	i^{th} LoRa node (sensor)
n_{i_d}	i^{th} node distance from *GW*
Pr	Transmission priority, $Pr \in \{LP, LMP, MP, UMP, HP\}$
R_{colli}	Retransmission due to a packet collision
R_{ch}	Retransmission due to a packet loss
S_n	Transmission slot
T_n	Transmission time interval
T_m	Transmission mode
$T_{m_{con.}}$	Conservative transmission mode
$T_{m_{ncon.}}$	Non-conservative transmission mode
$Th_{con.}$	Threshold value of z for $T_{m_{con.}}$
$Th_{ncon.}$	Threshold value of z for $T_{m_{ncon.}}$
v_{ab}	the normalisation value of A & B
z_{C_K}	Average value of z in cluster of K
z_{n_i}	Difference between n_i's values *A* & *B*

A. SYSTEM MODEL

The system model shown in Fig. 1 resembles a forest scenario with one gateway (*GW*) and randomly distributed nodes $n \in \{n_1, n_2, \ldots n_i\}$, where $1 \leq i \leq 1000$. The LoRa nodes n_i are configured following LoRa SX1272 model. This is to validate n_i's performance against practical experiments carried out in [40], [41]. The nodes are stationary and communicate with the *GW* following Class A LoRaWAN protocol, while the *GW* communicates back through a temporary receive window that opens following each transmission from n_i [1].

Note that this work is based on applying a clustering algorithm, which usually incorporates the conventional solution of deploying multiple *GW* in order to provide transmission alternatives for the nodes in different clusters. However, this is not the case in this work for number of reasons. Firstly, deploying multiple *GW* introduces a set of problems, some of which are multipath propagation, interferences, and nodes transmission duplication just like the problem reported in [37]. Secondly, multiple *GW* is usually ideal in protocols where the energy efficiency is not as critical as the low-latency and ultra-reliability requirements, for example, cellular networks (*5G*). However, this is not the case in low power protocols like LoRaWAN, where the resources are

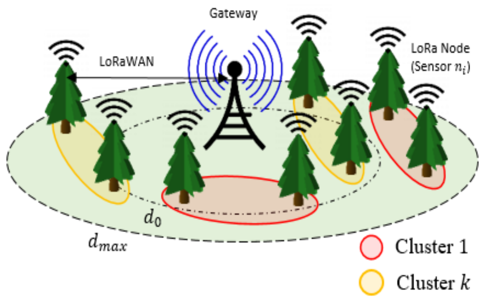

FIGURE 1. Dense application resembling a forest scenario using LoRaWAN.

limited [1]. Therefore, considering more than one *GW* could deviate the scope of this work away from evaluating the feasibility of adapting LoRaWAN for severing dense IoT applications as a worse-case scenario. Finally, thanks to the *CSS* technique in LoRa modulation, a LoRa GW is reported in several studies and experiments to be capable of serving thousands of nodes [42]–[44]. Hence, the complexity in this work lies in reducing the collision rate and therefore, enhancing the *PDR* in LoRaWAN while maintaining relatively low *TEC* and *TTD*, all using a single *GW*.

For that, this work scenario considers two sets of random values (A) and (B) that are assigned to each node n_i following random-uniform distribution.[2] The node n_i transmits these values to the gateway *GW*, where the clustering is formed. Based on these values the node is assigned to the corresponding cluster. This is for the purpose of evaluating the wildfire possibility within the covered area.

In [40], [41], two different experimental projects using LoRaWAN were carried out to evaluate the channel condition impact on the *PDR*. Both were carried out in urban environments where obstacles are highly deployed between the nodes and the gateway. Both showed that the node's distance n_{i_d} from the *GW* has a great impact on the *PDR*. Specifically, n_i located at a distance (d) more than 500 meters (d_0) away from the *GW* ($n_{i_d} \geq d_0$) experiences a bad channel condition, where the *PDR* ranges between 50% to 90%. On the other hand, the *PDR* is guaranteed more than 90% when $n_{i_d} \leq d_0$.

Since this work is inspired by a forest scenario and based on the results in [40], [41], only one fifth of the nodes are distributed within a range of d_0 from the *GW*. These nodes are assumed to have a good channel condition with a *PDR* more than 90%. The rest of the nodes are distributed at distances range from d_0 up to 3000 meters (d_{max}). These nodes have bad channel condition with *PDR* that can deplenish to 50%. In other words, for a more realistic system model, only one fifth of the nodes have good *PDR* of more than 90% while the rest are vulnerable to packet loss.

In addition, the nodes communicate with the *GW* using Class *A* of LoRaWAN protocol, spreading factor (*SF7*) and coding rate of 4/5. These parameters are particularly chosen to provide the maximum data rate, lowest transmission delays, and lowest energy consumption for the network. Although the LoRaWAN network with the most reliable *SF7* provides the best performance in terms of data rate, transmission delay and energy consumption [1], [19], the LoRaWAN network still under-performs in certain scenarios due to high collision rate, especially when the network is dense. Hence, given the system model is dense at a limited area (up to d_{max} around the *GW*), the analysis in this paper is based on using *SF7*.

The main objective of this work is to enhance the *TTD* and *TEC*. This is achieved by reducing the excessive packet collision rate associated with LoRaWAN due to the adaption of ALOHA protocol communication in Class A LoRaWAN [45]. Packet collisions happen when two packets are transmitted at the same time over the same frequency using the same *SF* [1], [19]. When a collision happens, the node keeps attempting to retransmit until an acknowledgement from the *GW* is received, which results in increasing the *TTD* and eventually the *TEC*.

Since LoRaWAN adopts ALOHA protocol for communications between the nodes and the gateway, the node transmits packets whenever there are ready to transmit data, regardless of the channel status. Hence, following Poisson distribution, the probability P of a packet collision to happen is given as in equation (1):

$$P = e^{-2G} \tag{1}$$

where G is the rate of packet transmission attempts per node. Hence, having more nodes transmitting at the same time increases the probability of a packet collision. Simulations are carried out in section V to show the proportional relationship between the number of nodes and the total collision rate.

Given the limited resources and random transmission behaviour of LoRaWAN nodes, it is essential to minimise the number of simultaneous transmission. Considering the given application scenario with the unlabeled data associated with the nodes, an effective method to reduce the simultaneous transmissions is to partition the nodes into different clusters. Assuming sufficient resources for the gateway, K-Means can be adopted to perform clustering of the nodes based on their transmitted data. Therefore, the gateway applies the dynamic *PST* to regulate the nodes transmissions without exhausting the nodes limited resources in the transmission intervals configuration process.

Since the aim is to reduce transmission delay and energy consumption, formulations of *TTD* and *TEC* are essential in order to evaluate the effect of applying K-Means to the system model. The following subsection reveals *TTD* and *TEC* as functions of K number of clusters (C_K). In addition, the formulations show that *TEC* is proportional to *TTD*.

B. PROBLEM FORMULATION

1) TOTAL TRANSMISSION DELAY (*TTD*)

The transmission delay is a function of the packet's number of bits and the bitrate [8]. It is proportional to the number of

[2]These values can be adjusted according to any application. As for the forest scenario adopted in this work, the values A and B represent atmospheric humidity and weather temperature respectively.

bits within a packet and it is given in equation (2):

$$Transmission\ Delay = \frac{Number\ of\ bits}{Bitrate} \tag{2}$$

The bitrate is given by (3):

$$Bitrate = \frac{SF \times BW}{2^{SF}} \times \frac{4}{4 + CR} \tag{3}$$

where SF is the Spreading Factor and it is fixed to $SF7$, $BW = 125\ kHz$ and CR is the Coding Rate and is set to $CR = 1$. Note that $CR \in \{4/5, 4/6, 4/7, 4/8\}$ is the ratio of the actual data bits to the redundant bits and is represented by $CR = \{1, 2, 3, 4\}$, respectively. Using these parameters insures a maximum successful transmissions in LoRaWAN given a limited area [1]. More details of the BW, SF, and CR are given in [19].

In the proposed dynamic PST (section IV), the GW assigns different transmission priorities Pr to K number of clusters C_K, where $K = \{1, 2, 3, \ldots, k\}$. The nodes in a lower Pr cluster wait until transmissions from nodes in higher Pr clusters are satisfied. This introduces waiting times in lower transmission priority clusters. Hence, TTD can be given as in equation (4):

$$TTD(K) = \sum_{j=1}^{K} (D_{C_1}, D_{C_2}, \ldots, D_{C_K}), \tag{4}$$

where K is the number of clusters in the network, D_{C_K} is the total transmission delay of n in a cluster of C_K and is given as in equation (5):

$$D_{C_K} = \sum_{i=1}^{n_{C_K}} X(i), \tag{5}$$

where n_{C_K} is the total number of all n_i in the corresponding C_K; $X(i) = (D_{I_T} + D_{R_{colli}} + D_{R_{ch}})$; D_{I_T} is the delay of the initial transmission I_T of each n_i; $D_{R_{colli}}$ is the delay of the retransmission caused by n_i's collided packet (R_{colli}); $D_{R_{ch}}$ is the delay of the retransmission caused by n_i's lost packet due to bad channel condition (R_{ch}). Note that the GW is assumed to be able to distinguish between I_T, R_{colli} and R_{ch}.

Since the transmissions from n_i in the lower Pr clusters C_{LPr} wait until transmissions from n_i in the higher Pr clusters C_{HPr} are satisfied, the delay of $D_{C_{LPr}}$ is given as in equation (6):

$$D_{C_{LPr}} = D_{C_{HPr}} + \sum_{i=1, i \notin C_{HPr}}^{n_{C_{LPr}}} X(i), \tag{6}$$

where $D_{C_{HPr}}$ is the delay of all n_i in higher Pr clusters and $n_{C_{LPr}}$ is n_i in the corresponding C_{LPr} clusters.

2) ENERGY CONSUMPTION

The GW using the proposed dynamic PST (detailed in section IV) regulates the transmissions from n_i in different clusters of C_K to the GW based on the corresponding transmission Pr. Hence, each of C_K is either at active or idle transmission status. When a cluster of C_K is at an active status, the corresponding n_{C_K} are allowed transmissions. Otherwise, the cluster is at an idle transmission status, and no transmissions from the corresponding n_{C_k}. Note that only one cluster of C_K can be active at a time. Hence, TEC as a function of the number of clusters *(K)* can be given as in equation (7):

$$TEC(K) = E_{active} + \sum_{j=1, E_{active} \notin j}^{K} E_{idle}^{(j)} \tag{7}$$

where E_{active} and E_{idle} are the energy consumption in *Joules* of all n_{C_K} in the corresponding active and idle clusters of C_K, respectively. E_{active} and E_{idle} are given in equations (8) and (9), respectively.

$$E_{active} = \sum_{i=1}^{n_{C_k}} (P_T \times D_{active}), \tag{8}$$

$$E_{idle} = \sum_{i=1}^{n_{C_k}} (P_{idle} \times D_{idle}), \tag{9}$$

where P_T and D_{active} are the transmission's power and duration of n_i in an active cluster of C_K. While P_{idle} and D_{idle} are the power consumption and the duration of standby n_i in the other idle clusters of C_K.

From III-B1 and III-B2, the TTD and TEC are proportionally impacted by the number of n_i's initial transmissions and retransmissions of collided or lost packets. In other words, minimising TTD eventually results in minimising TEC. Hence, from equations (4) and (7), the objective function of obtaining the minimum value of TTD at a given number of clusters K can then be represented as in equation (10), subject to a number of constraints:

$$\min_{k} TTD(K) \tag{10}$$

$$S.T.$$

$$\min TEC(K) \tag{11}$$

$$Pr = K \tag{12}$$

$$0 \leq R_{colli} \leq 1 \tag{13}$$

$$R_{ch} = \begin{cases} 1, & d_0 < n_{i_d} < d_{max} \\ 0, & \text{else} \end{cases} \tag{14}$$

where constraint (11) denotes the proportionality of TEC to TTD at a given number of clusters K. Pr in constraint (12) is the transmission priority assigned to each cluster of K. The process of assigning Pr to each cluster is revealed in section IV. R_{colli} in constraint (13) is the retransmission of collided packets and it is limited to one retransmission per node. This is to retain the practicality of simulations given the considered high number of nodes. R_{ch} in constraint (14), is the retransmission of lost packets due to bad channel condition for nodes n_i located at distances further than d_0 from the GW.

C. UNSUPERVISED LEARNING CLUSTERING ALGORITHM (K-MEANS)

Clustering algorithms are well-known machine learning approaches for having more control over wireless networks resource allocation. LoRaWAN adopts star topology where the nodes communicate directly to the gateway. In a dense application, the probability of collisions to happen increases vastly affecting the network overall performance. Hence, for the purpose of reducing the packet collision rate within LoRaWAN; the unsupervised learning K-Means clustering algorithm is adopted in our system model to achieve less collision rates via reducing the unnecessary number of simultaneously transmitting nodes.

In our previous work [18], the impact of adopting the unsupervised clustering algorithm K-Means on the total collision rate was evaluated. In fact it was noticed that the *TTD* is a decreasing function of the number of clusters. However, in K-Means, the number of clusters k is a predefined value. Hence, in this work the optimal number of clusters is obtained according to the most efficient performance of *TTD* and *TEC* against the number of clusters k.

The partition of the nodes takes place by minimising the within-cluster sum of square (WCSS) of the given data set $z_n = \{z_{n_1}, z_{n_2}, \ldots z_{n_i}\}$, where z_{n_i} is the difference between the values A and B, which are transmitted by n_i as explained in the system model. Since the values A and B can be measured in different units, a normalisation is needed to obtain the value of z. In other words, nodes with almost similar values of z are grouped together forming one cluster. Note that the clustering is based on the values of z_n. This means that nodes at different locations from the *GW* can belong to the same cluster, see Fig. 1. The objective partitioning function can then be represented as in equation (15):

$$\arg\min_{c_j} \sum_{j=1}^{k} \sum_{z_{n_i} \in C_k} \| z_{n_i} - c_j \|^2 \tag{15}$$

where c_j is an initial value of z fixed to form the center point of the corresponding cluster C_k. Note that the clustering is formed at the *GW* level. The *GW* is assumed to be aware of z_n from previous successful transmissions. z_{n_i} is then updated at the *GW* upon each successful transmission from the corresponding n_i.

K-Means is a suitable unsupervised machine learning clustering tool for networks with limited resources. This is due to the simplicity of performing the clustering process provided unlabeled data [17]. However, since the number of clusters play a vital role in the clustering process, implementing K-Means can be very complex in the case of excessively diversified data. Hence, it is very important to define objectives that can be used to evaluate the optimal number of clusters.

In order to obtain the optimal number of clusters, extensive simulations were carried out to evaluate *TTD* and *TEC* performance at a different number of clusters k, where $(0 \leq k \leq 30)$. Based on equations (4) and (7), Fig. 2 shows that *TTD* and therefore *TEC* are generally decreasing functions of k. On one hand, it is noticed that *TTD* and *TEC* sharply decrease until $(k = 5)$. This is mainly due to the reduction of the number of nodes within each cluster and

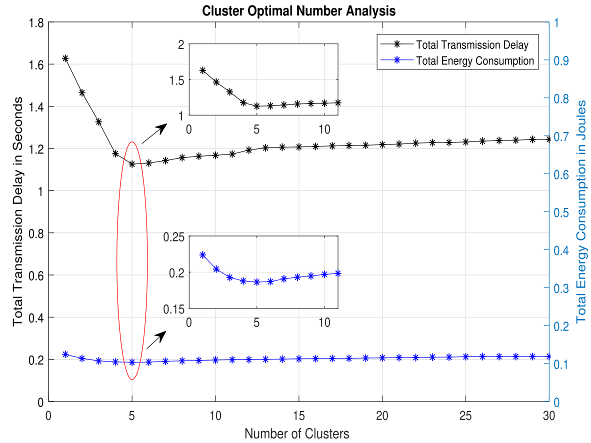

FIGURE 2. Simulation analysis of the optimal number of clusters.

hence the reduction of the collision rate caused by nodes transmitting at the same time. Note that collided packets get retransmission attempts, which impact both *TTD* and *TEC* in an almost symmetrical manner. On the other hand, at $(5 \leq k)$, *TTD* and *TEC* start to regain their values forming convex curves. This is due to the gradually fading impact of the retransmissions caused by packet collisions R_{colli}, and the increasing impact of the retransmissions caused by packet loss due to bad channel conditions R_{ch}. Thus, from Fig. 2, the optimal number of clusters K_{opt} in our scenario is at $(k = 5)$, where *TTD* and *TEC* at the bottom points of the convex curves forming the lowest values.

IV. PROPOSED DYNAMIC PRIORITY SCHEDULING TECHNIQUE

This section reveals the proposed dynamic Priority Scheduling Technique (*PST*), where the *GW* schedules transmissions from the nodes according to different transmission priorities assigned to the different clusters that are obtained by K-Means. Initially, a set of values z_{n_i} transmitted to the *GW* from each n_i is processed to partition n_i to different clusters. Following the clustering formation, each n_i is assigned to a cluster of C_K. Let z_{C_K} denote the average value of all z_{n_i} within the same cluster of C_K. Based on z_{C_K}, the *GW* using the dynamic *PST* designates different transmission priorities (Pr) to the different C_K. Since the optimal number of clusters is $K_{opt} = 5$, the transmission priorities range between lowest, lower-middle, middle, upper-middle and highest, where $Pr = K_{opt}$ and $Pr \in \{LP, LMP, MP, UMP, HP\}$, respectively.

Following the transmission priority designation process to each cluster of C_K, the dynamic *PST* provides two transmission modes to trade-off *TTD* and *TEC* for further *PDR* gain according to each cluster transmission priority C_{Pr}. Given the density in the network, the *GW* using the Naive Bayes classifier determines the probability of each n_i to transmit using a certain transmission mode.

A. TRANSMISSION PRIORITY SCHEDULING

For better elaboration, it is necessary to explain the details of the considered scenario in this work. The *GW* assigns n_i

that has highest value of z_{n_i} to a highest transmission Pr cluster. To reiterate, z is the difference between the values A & B, where A_{n_i} & B_{n_i} represent the atmospheric humidity and weather temperature values transmitted by n_i, respectively. Since A & B can be measured in different units, a normalisation is needed to obtain the value of z. There are a number of normalisation methods [46]–[49], which vary in terms of the considered values. Considering the scenario adapted in this work, A & B are given as upward and downward attributes.[3] Hence, the enhanced max-min normalisation method is adapted for adjusting the normalisation value $v_{ab} = f(A, B)$ and is given as in equation (16):

$$v_{ab} = \begin{cases} \text{for upward attributes:} \\ 1 - \dfrac{|A_{n_i} - max(A_{nC_K})|}{(max(A_{nC_K}) - min(A_{nC_K}))} \\ \text{for downward attributes:} \\ 1 - \dfrac{|B_{n_i} - min(B_{nC_K})|}{(max(B_{nC_K}) - min(B_{nC_K}))} \end{cases} \tag{16}$$

where A_{nC_K} and B_{nC_K} represent A and B values reported by n_i in the corresponding cluster C_K.

The designation process of the Pr level to each cluster of C_K follows equation (17):

$$\max(Pr) = \max(z_{C_K}) \tag{17}$$

where z_{C_K} denotes the average value of z in the corresponding cluster C_K. Note that $z_{C_K} = A_{C_K} - B_{C_K}$, where A_{C_K} and B_{C_K} denote the average values of A and B in the corresponding cluster C_K, respectively.

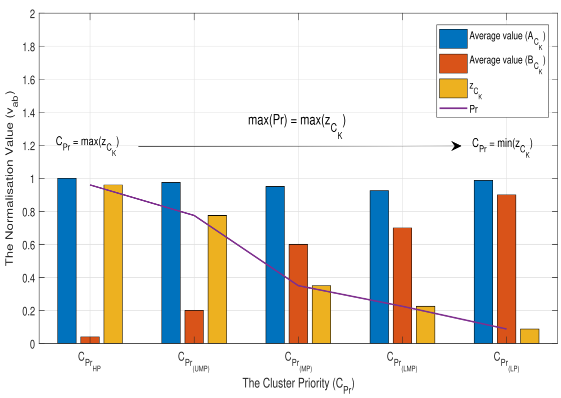

FIGURE 3. Transmission priority designation to the clusters of C_K based on the corresponding value z_{C_K}.

For further illustration, the transmission cycle used in the simulations of this work is depicted in Fig. 3 to show the Pr designation process. From Fig. 3, the Pr is proportional to z_{C_K}. This means the higher value of z_{C_K} is assigned higher Pr. Based on the cluster transmission priority C_{Pr}, the GW configures transmissions from the corresponding nodes accordingly. In other words, the GW allows transmissions from n_i in higher Pr clusters $n_{C_{HPr}}$, where it blocks transmissions from n_i in lower Pr clusters $n_{C_{LPr}}$.

This strict condition introduces a network underperformance for some cases. For example, given a dense application, the transmissions from C_{LPr} can be blocked due to the presence of excessive and unnecessary transmissions from C_{HPr} that may not be desirable. In order to achieve further performance gain, the proposed dynamic PST is performed under two transmission modes (T_m): conservative ($con.$) and non-conservative ($ncon.$).

Algorithm 1 Transmission Priority Scheduling Process

At the GW **level**
Initialize: TTD, TEC, n_i, z_{n_i}, $n_{i_{Pr}}$, $Pr \in \{LP, LMP, MP, UMP, HP\}$,
To Achieve $min\, TTD$ & $min\, TEC$;
for n_i **do**
  if $n_i \in HP$ **then**
  $n_i := n_{i_{HP}}$ and $n_{i_{HP}}$ transmission = 1;
  else if $n_i \in UMP$ and $(n_{i_{HP}}) = 0$ **then**
  $n_i := n_{i_{UMP}}$ and $n_{i_{UMP}}$ transmission = 1;
  else if $n_i \in MP$ and $(n_{i_{HP}}, n_{i_{UMP}}) = 0$ **then**
  $n_i := n_{i_{MP}}$ and $n_{i_{MP}}$ transmission = 1;
  else if $n_i \in LMP$ and $(n_{i_{HP}}, n_{i_{UMP}}, n_{i_{MP}}) = 0$ **then**
  $n_i := n_{i_{LMP}}$ and $n_{i_{LMP}}$ transmission = 1;
  else if $n_i \in LP$ and $(n_{i_{HP}}, n_{i_{UMP}}, n_{i_{MP}}, n_{i_{LMP}}) = 0$ **then**
  $n_i := n_{i_{LP}}$ and $n_{i_{LP}}$ transmission = 1;
  else
  $n_{i_{Pr}}$ transmission = 0;
  end if
end for

Based on the following assumptions, Algorithm (1) shows the process of the transmission initiations from each n_i according to its corresponding cluster.

Assumptions:

- GW already has z_{n_i} for all the nodes from previous successful transmissions
- z_{n_i} at the GW are updated upon each successful transmission
- Each n_i transmit one packet an hour to the GW unless configured otherwise
- Each n_i is allowed only one retransmission in the case of a collision R_{colli}
- Each n_{i_d} at $d_0 \leq d \leq d_{max}$, is allowed one retransmission in the case of a packet loss R_{ch} due to bad channel condition

[3] In a forest scenario, it is less likely for a wildfire to happen when the atmospheric humidity (A) is high, while the wildfire possibility increases with lower values of A. Hence, the value A is considered as an upward attribute. Vice versa, it is less likely for a wildfire to happen when the weather temperature (B) is low and the possibility increases with higher values of B. Hence, the value B is considered as a downward attribute.

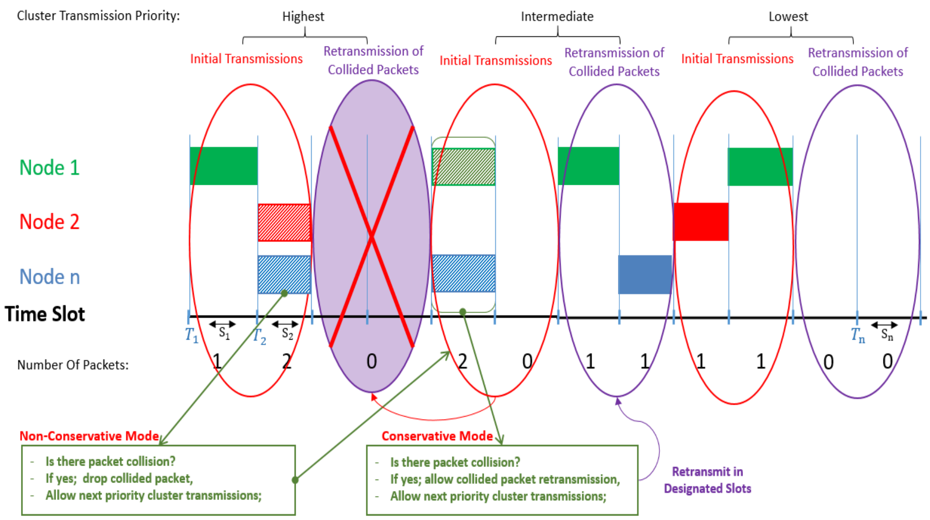

FIGURE 4. Transmission modes control.

- The environment is idle, where there are no inter communications exist and the channel duty-cycle constraint is neglected

B. TRANSMISSION MODES OPTIONS

The two transmission modes T_m (*con.* and *ncon.*) are provided by the dynamic *PST* to trade-off *PDR* with *TTD* and *TEC*. T_m is defined based on whether or not retransmissions of collided packets R_{colli} in each C_K are permitted. In other words, the purpose is to allow the *GW* to assess whether there is a need for R_{colli}, hence control the *PDR* accordingly. The *GW* decides which mode to operate for each C_K based on threshold values (*Th*) assumed to be provided by a third party (e.g local authority). These *Th* are $Th_{con.}$ for *con.* mode and $Th_{ncon.}$ for *ncon.* mode. Furthermore, $Th_{con.}$ and $Th_{ncon.}$ contain a set of values of z that act as limits. The *GW* uses these limits in order to determine the probability of using one of the two T_m by each C_K.

In *con.* mode, the *GW* allows retransmissions of collided packets (R_{colli}). Note that for simulation practicality the n_i with collided packet is allowed one retransmission attempt. In the case of any further collisions, the collided packets will be dropped. Operating the *con.* mode elevates *PDR* at the expense of higher *TTD* and *TEC*. On the other hand, R_{colli} in *ncon.* mode is not allowed. Operating the *ncon.* mode minimises *TTD* and *TEC* at the expense of lower *PDR*. Simulations results in section V show that both transmission modes T_m maintain acceptable *PDR* in comparison to other techniques. Fig. 4 illustrates the difference between both *con.* and *ncon.* transmission modes. While Algorithm 2 shows the dynamic *PST* process of alternating between the two transmission modes according to *Th* values.

C. NAIVE BAYES CLASSIFIER ALGORITHM

Considering a dense application with massive amount of transmissions from n_{C_K} to *GW*, the process of classifying C_K to a certain T_m can be time inefficient. For this reason, the *GW* applies the Naive Bayes classifying algorithm to efficiently determine the probability of n_{C_K} transmissions using either $T_{m_{con.}}$ or $T_{m_{ncon.}}$. Where $T_{m_{con.}}$ and $T_{m_{ncon.}}$ denote *con.* and *ncon.* transmission modes, respectively. According to *Th* values, the *GW* classifies each cluster of C_K to a certain T_m based on the average value of z_{C_K} following equation (18):

$$P(T_{m_{con.}}|z_{C_K}) \geq P(Th_{m_{con.}}|z_{C_K}) \tag{18}$$

Algorithm 2 Dynamic *PST* Transmission Modes

At the *GW* **level**

Initialize: z_{C_K}, $T_{m_{con.}}$, $T_{m_{ncon.}}$, I_T, R_{colli}, R_{ch}, n_{i_d}, d_0, $Pr \in \{LP, LMP, MP, UMP, HP\}$

for $T_{m_{con.}}(HP, UMP)$ **do**
  $I_T = 1;$
  $R_{colli} = 1;$
  if $n_{i_d} > d_0$ **then**
  $R_{ch} = 1;$
  else
  $I_T = 0;$
  $R_{colli} = 0;$
  $R_{ch} = 0;$
  end if
end for
for $T_{m_{ncon.}}(MP, LMP, LP)$ **do**
  $I_T = 1;$
  $R_{colli} = 0;$
  if $n_{i_d} > d_0$ **then**
  $R_{ch} = 1;$
  else
  $I_T = 0;$
  $R_{colli} = 0;$
  $R_{ch} = 0;$
  end if
end for

where $P(T_{m_{con.}}|z_{C_K})$ is the posterior probability of a cluster of C_K to transmit using $T_{m_{con.}}$ and is given as in equation (19):

$$P(T_{m_{con.}}|z_{C_K}) = \frac{P(z_{C_K}|T_{m_{con.}})P(T_{m_{con.}})}{P(z_{C_K})} \tag{19}$$

where $P(z_{C_K}|T_{m_{con.}})$ denote the posterior probability of z_{C_K} conditioned on $T_{m_{con.}}$; $P(T_{m_{con.}})$ is the prior probability of $T_{m_{con.}}$; and $P(z_{C_K})$ is the prior probability of z_{C_K}. In a similar approach, equation (19) is applied to obtain the posterior probability of $P(Th_{m_{ncon.}}|z_{C_K})$.

z_{C_K} is an independent value that varies in each cluster of K, which can result in a high computation complexity when obtaining $P(T_{m_{con.}}|z_{C_K})$. In order to reduce the computation complexity, the posterior probability $P(z_{C_K}|T_{m_{con.}})$ can be calculated is in equation (20):

$$\prod_{K=1}^{K_{opt}} P(z_{C_K}|T_{m_{con.}}) = P(T_{m_{con.}}) \times P(z_{C_K}) \tag{20}$$

TABLE 2. Likelihood occurrence pattern table.

C_K	$T_{m_{con.}}\ (+\alpha)$	$T_{m_{ncon.}}\ (+\alpha)$
C_1	1 (2)	0 (1)
C_2	1 (2)	0 (1)
C_3	0 (1)	1 (2)
C_4	0 (1)	1 (2)
C_5	0 (1)	1 (2)
$P(T_m)$	$P(T_{m_{con.}}) = \frac{2}{2+3} = 0.4$	$P(T_{m_{ncon.}}) = \frac{3}{3+2} = 0.6$

Since there are two transmission modes $T_{m_{con.}}$ and $T_{m_{ncon.}}$, a cluster of C_K communicates with GW using one of them. Hence, the corresponding T_m is denoted by 1 when used by a cluster and 0 when not in use. The out of use T_m denoted by 0 can cause inaccurate probability results when using the product function in equation (20). Hence, α is added to avoid such confusion in the Naive Bayes classifying process, where $\alpha = 1$.

using equation (20), the communications data set in Fig. 3 is utilised as a training set to construct the likelihood occurrence pattern given in Table 2. This is to determine the probability of classifying a cluster of C_K to a certain T_m. In particular, there are K number of clusters, where $K = \{1, 2, \ldots, K_{opt}\}$. Lets assume a threshold value for the *con.* mode, $Th_{con.} \geq 0.5$, where the value 0.5 represent z_{C_K}. Hence, all clusters with $z_{C_K} \geq 0.5$ are considered as higher priority clusters that more likely need to communicate with the *GW* using $T_{m_{con.}}$. The rest of the clusters are more likely to communicate with the *GW* using $T_{m_{ncon.}}$. Thus, based on the Bayesian theorem, the probability of having a cluster assigned to using $T_{m_{con.}}$ is given as in $P(T_{m_{con.}})$ and $P(T_{m_{ncon.}})$. Note that the classifying process is performed by the *GW* upon each new transmission cycle, where the T_m probability is determined according to the new values of z_{C_K} reported by each n_i. Fig. 5 illustrates the dynamic *PST* using Naive Bayes classifier.

V. DISCUSSION AND SIMULATION RESULTS

The impact of the proposed dynamic *PST* on the network performance is evaluated via simulations following the parameters in Table 3.

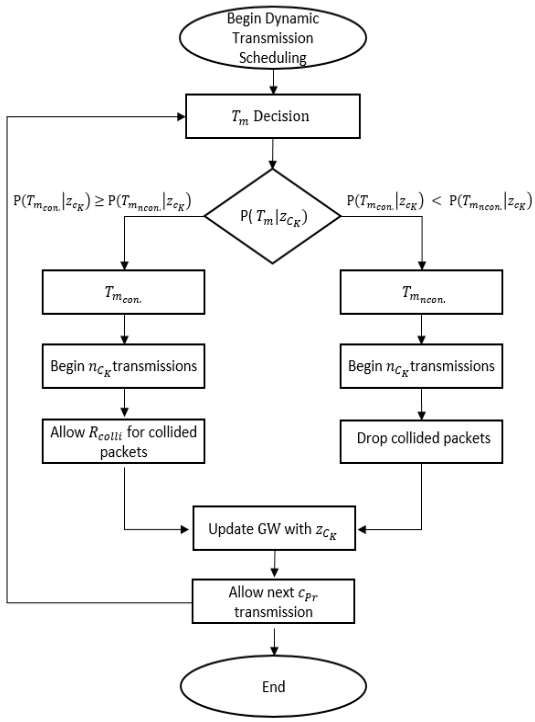

FIGURE 5. Naive bayes classifier in the dynamic priority scheduling technique (*PST*).

TABLE 3. Simulation parameters.

Parameter	Value
Protocol	LoRaWAN (v1.1)
Number of nodes n_i	1000
Payload size	25 $Bytes$
SF	7
CR	4/5
BW	125 kHz
Channel frequency	915 MHz
Power consumption per active n_i	0.1 W
Power consumption per idle n_i	0.072 W
Transmission duration	0.036 s
Optimal number of clusters	$K_{opt} = 5$

It is shown that LoRaWAN is vulnerable to severe collision rate especially when serving a high number of nodes. This is due to the fact that LoRa nodes adapt ALOHA style communication in its LoRaWAN protocol of Class *A* [1], [19]. As shown in Fig. 6, the total collision rate in typical LoRaWAN network serving up to 1000 n_i is up to 91.3%. This motivated our previous work [18] to adopt the unsupervised learning K-Means clustering algorithm for the aim of reducing the collision rate. Having introduced the optimal number of clusters K_{opt}, the total collision rate when $k = 5$ is vastly reduced to an average of 38%. Note that the more clusters introduced to the system result in less nodes simultaneously transmitting, which in return reflect in less packet collisions. However, this comes at the expense of inefficient *TTD* and *TEC* (as discussed in section III-C, Fig. 2).

In regards to the *TTD*, *TEC* and *PDR*, a comparison of the proposed dynamic *PST* is carried out against typical

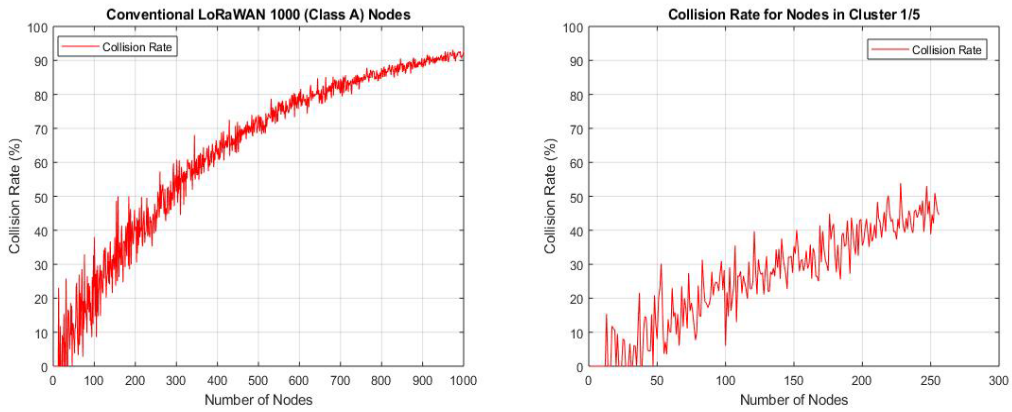

FIGURE 6. Collision rate in conventional LoRaWAN vs. collision rate in one cluster of K_{opt}.

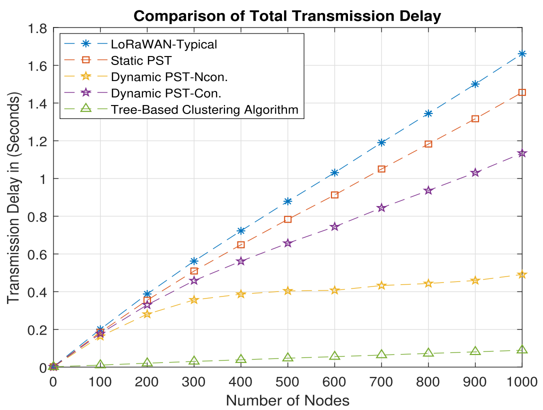

FIGURE 7. Total transmission delay.

LoRaWAN network, the static *PST* proposed in [18], and the tree-based clustering algorithm scheme proposed in [16].

On one hand the typical LoRaWAN shows the least efficient *TTD*. This is due to the adaption of star topology accompanied with the ALOHA protocol communication in LoRaWAN. Where the nodes initiate transmissions to the *GW* regardless any other transmission occupying the channel. For this reason the collision rate is excessively high especially when scaling up the network. As a result, retransmission attempts are much higher, which in return increase the *TTD* of the network. Note that due to the high number of nodes considered in this scenario and for simulation practicality, the retransmissions of collided packets are limited to one per node. The static *PST* comes second after typical LoRaWAN with slight improvement to *TTD*. This improvement is mainly due to the clustering where the number of nodes in each cluster is reduced. This results in lower number of transmission attempts and according to equation (1), the probability of the packet collision to happen is significantly impacted by the transmission attempts which is proportionally related to the number of nodes transmitting at the same time. Although the number of nodes in each cluster is significantly decreased, there are still collisions that happens where each collision result in another transmission attempt regardless the necessity of the retransmission.

Therefore, the proposed dynamic *PST* outperformed both typical LoRaWAN and static *PST*. This is due to the ability to alternate between *con*. and *ncon*. transmission modes. Note that in the simulation results, the dynamic *PST* is represented as *con*. when the majority of clusters are transmitting using $T_{m_{con.}}$, while it is represented as *ncon*. when the majority of clusters are transmitting using $T_{m_{ncon.}}$. It can be noticed that the *TTD* at *con*. is high in comparison to that of *ncon*. mode. This is because the nodes in a cluster that is assigned a $T_{m_{con.}}$, has the chance to initiate a retransmission for each collided packet. Vice versa, the *TTD* at *ncon*. is relatively low due to the strict retransmission condition which results in each collided packet to be dropped.

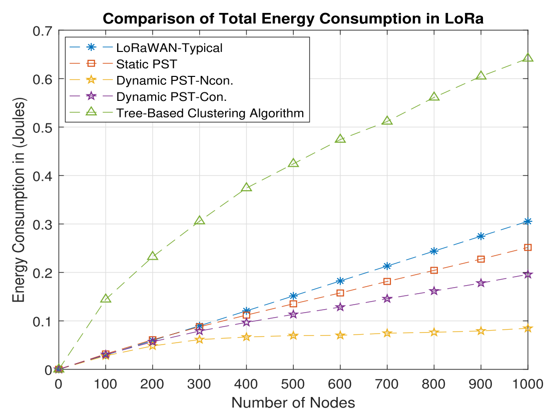

FIGURE 8. Total energy consumption.

The tree-based clustering algorithm proposed in [16] shows the best *TTD* amongst all approaches. This is due to the rerouting approach, which results in avoiding a packet collision by utilising multi-hop technique that relay the packet to the *GW* through other routes using neighbouring nodes. However, this comes at the expense of much higher *TEC* as shown in Fig. 8, which defeats the whole purpose of using LoRaWAN as a low power technology.

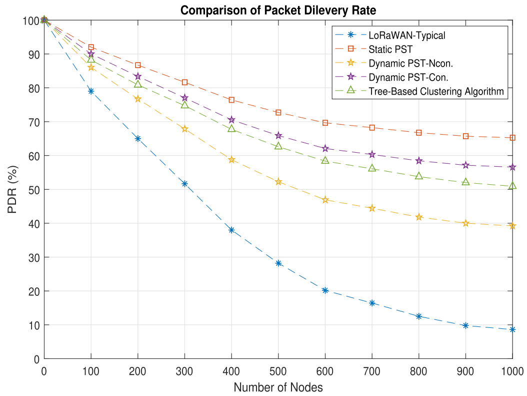

FIGURE 9. Packet delivery rate.

Referring back to section III-B2, *TEC* is directly effected by the number of collisions and hence the number of retransmissions as a consequence. For that, it can be noticed that *TEC* is generally proportional to *TTD*. In Fig. 8, *con*. mode shows more energy consumption when comparing to *ncon*. mode, however despite it consumes more *TEC*, adapting *con*. mode provides better *PDR* in comparison to *ncon*. mode as shown in Fig. 9. This comes as a result of allowing nodes in clusters that are transmitting using *con*. to initiate retransmissions of collided packets. Hence, it can be noticed that when *PDR* in *con*. is high, the *TEC* and therefore *TTD* are high, whereas the opposite in *ncon*. mode. From Fig. 9, the static *PST* outperforms the proposed dynamic *PST*, however this comes at the expense of more *TTD* and *TEC*. While the dynamic *PST* at *con*. mode shows better *PDR* in comparison to the tree based clustering algorithm. Hence, given the trade-off between *TTD*, *TEC* and *PDR*, the proposed dynamic *PST* shows more suitability for being adapted in LoRaWAN to serve dense IoT applications.

VI. CONCLUSION

The use of machine learning techniques can lead to inefficient energy consumption when applied to low power networks with limited resources. This is because machine learning techniques usually require coordination between the end-nodes and the gateway. However, the use of the unsupervised clustering algorithm K-Means in LoRaWAN network has shown a great impact in reducing the collision rate and therefore, higher *PDR*, while maintaining low energy consumption and transmission delay. This is mainly due to partitioning the nodes into different clusters, which in return reduces simultaneous transmissions as a result of using the dynamic *PST* to configure the nodes with different transmission intervals based on the clusters transmission priorities. Given the same network density, the static *PST* reduced *TTD* and *TEC* by 11.9% and 16.6% from the typical LoRaWAN, respectively. Despite the slight improvement to *TTD* and *TEC*, the static *PST* significantly enhanced the *PDR* when compared to typical LoRaWAN. Although the tree-based clustering approach sharply reduced *TTD* by almost 94%, this comes at expense of an extravagant increase to *TEC* by more than 116% in comparison to typical LoRaWAN. Such an increase to the *TEC* may defeat the core purpose of using LoRaWAN as a low power technology. Hence, the fair trade-off between *TTD*, *TEC* and *PDR* provided by the proposed dynamic *PST* due to its ability to alternate between two transmission modes, makes it the most convenient amongst the other considered approaches especially when serving dense IoT applications.

REFERENCES

[1] LoRa Alliance. *LoRaWAN Specification v1.1*. Accessed: Jul. 12, 2020. [Online]. Available: https://lora-alliance.org/resource-hub/lorawantm-specification-v11
[2] J. Ortin, M. Cesana, and A. Redondi, "Augmenting LoRaWAN performance with listen before talk," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3113–3128, Jun. 2019.
[3] Q. L. Hoang, W.-S. Jung, T. Yoon, D. Yoo, and H. Oh, "A real-time LoRa protocol for industrial monitoring and control systems," *IEEE Access*, vol. 8, pp. 44727–44738, 2020.
[4] L. P. Fraile, S. Tsampas, G. Mylonas, and D. Amaxilatis, "A comparative study of LoRa and IEEE 802.15.4-based IoT deployments inside school buildings," *IEEE Access*, vol. 8, pp. 160957–160981, 2020.
[5] E. Sisinni, P. Ferrari, D. Fernandes Carvalho, S. Rinaldi, P. Marco, A. Flammini, and A. Depari, "LoRaWAN range extender for industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5607–5616, Aug. 2020.
[6] Q. Zhou, K. Zheng, L. Hou, J. Xing, and R. Xu, "Design and implementation of open LoRa for IoT," *IEEE Access*, vol. 7, pp. 100649–100657, 2019.
[7] H. H. R. Sherazi, L. A. Grieco, M. A. Imran, and G. Boggia, "Energy-efficient LoRaWAN for industry 4.0 applications," *IEEE Trans. Ind. Informat.*, early access, Apr. 2, 2020, doi: 10.1109/TII.2020.2984549.
[8] *Lora Modulation Basics*, Semtech, Camarillo, CA, USA, 2015.
[9] J. P. Shanmuga Sundaram, W. Du, and Z. Zhao, "A survey on LoRa networking: Research problems, current solutions, and open issues," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 371–388, 1st Quart., 2020.
[10] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of LoRaWAN," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 34–40, Sep. 2017.
[11] L. Casals, B. Mir, R. Vidal, and C. Gomez, "Modeling the energy performance of LoRaWAN," *Sensors*, vol. 17, no. 10, p. 2364, Oct. 2017.
[12] Ericsson. *Massive IoT in the City*. Accessed: May 4, 2020. [Online]. Available: https://www.ericsson.com/en/mobility-report/massive-iot-in-the-city
[13] J. Xu, J. Yao, L. Wang, Z. Ming, K. Wu, and L. Chen, "Narrowband Internet of Things: Evolutions, technologies, and open issues," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1449–1462, Jun. 2018.
[14] N. El Rachkidy, A. Guitton, and M. Kaneko, "Collision resolution protocol for delay and energy efficient LoRa networks," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 2, pp. 535–551, Jun. 2019.
[15] C.-H. Liao, G. Zhu, D. Kuwabara, M. Suzuki, and H. Morikawa, "Multi-hop LoRa networks enabled by concurrent transmission," *IEEE Access*, vol. 5, pp. 21430–21446, 2017.
[16] G. Zhu, C.-H. Liao, T. Sakdejayont, I.-W. Lai, Y. Narusue, and H. Morikawa, "Improving the capacity of a mesh LoRa network by spreading-factor-based network clustering," *IEEE Access*, vol. 7, pp. 21584–21596, 2019.
[17] J. Wang, C. Jiang, H. Zhang, Y. Ren, K.-C. Chen, and L. Hanzo, "Thirty years of machine learning: The road to Pareto-optimal wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1472–1514, 3rd Quart., 2020.
[18] M. Alenezi, K. K. Chai, S. Jimaa, and Y. Chen, "Use of unsupervised learning clustering algorithm to reduce collisions and delay within LoRa system for dense applications," in *Proc. Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2019, pp. 1–5.
[19] M. Alenezi, K. K. Chai, Y. Chen, and S. Jimaa, "Ultra-dense LoRaWAN: Reviews and challenges," *IET Commun.*, vol. 14, no. 9, pp. 1361–1371, Jun. 2020.
[20] K. Kavitha and G. Suseendran, "Priority based adaptive scheduling algorithm for IoT sensor systems," in *Proc. Int. Conf. Autom., Comput. Technol. Manage. (ICACTM)*, Apr. 2019, pp. 361–366.

[21] V. Kakkar, "Scheduling techniques for operating systems for medical and IoT devices: A review," *Global J. Comput. Sci. Technol., Hardw. Comput.*, vol. 17, no. 1, 2017. Accessed: Jun. 26, 2020. [Online]. Available: https://computerresearch.org/index.php/computer/article/view/1604
[22] J. S. Kumar, M. A. Zaveri, and M. Choksi, "Task based resource scheduling in IoT environment for disaster management," *Procedia Comput. Sci.*, vol. 115, pp. 846–852, Jan. 2017.
[23] B. Afzal, S. A. Alvi, G. A. Shah, and W. Mahmood, "Energy efficient context aware traffic scheduling for IoT applications," *Ad Hoc Netw.*, vol. 62, pp. 101–115, Jul. 2017.
[24] T. Kim, D. Qiao, and W. Choi, "Energy-efficient scheduling of Internet of Things devices for environment monitoring applications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.
[25] M. S. I. Rubel, N. Kandil, and N. Hakem, "Priority management with clustering approach in wireless sensor network (WSN)," in *Proc. 6th Int. Conf. Digit. Inf., Netw., Wireless Commun. (DINWC)*, Apr. 2018, pp. 7–11.
[26] O. O. Ogundile, M. B. Balogun, O. E. Ijiga, and E. O. Falayi, "Energy-balanced and energy-efficient clustering routing protocol for wireless sensor networks," *IET Commun.*, vol. 13, no. 10, pp. 1449–1457, Jun. 2019.
[27] H. P. Tran, W.-S. Jung, T. Yoon, D.-S. Yoo, and H. Oh, "A two-hop real-time LoRa protocol for industrial monitoring and control systems," *IEEE Access*, vol. 8, pp. 126239–126252, 2020.
[28] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 551–591, 2nd Quart., 2013.
[29] K. Suwandhada and K. Panyim, "ALEACH-plus: An energy efficient cluster head based routing protocol for wireless sensor network," in *Proc. 7th Int. Electr. Eng. Congr. (iEECON)*, Mar. 2019, pp. 1–4.
[30] F. Liu and Y. Chang, "An energy aware adaptive kernel density estimation approach to unequal clustering in wireless sensor networks," *IEEE Access*, vol. 7, pp. 40569–40580, 2019.
[31] R. Kumar and D. Kumar, "Hybrid swarm intelligence energy efficient clustered routing algorithm for wireless sensor networks," *J. Sensors*, vol. 2016, pp. 1–19, Jan. 2016.
[32] S. Ghosh, S. Mondal, and U. Biswas, "A dominating set based modified LEACH using ant colony optimization for data gathering in WSN," in *Proc. 2nd Int. Conf. Adv. Electr., Electron., Inf., Commun. Bio-Inform. (AEEICB)*, Feb. 2016, pp. 390–396.
[33] H. Teng, X. Liu, A. Liu, H. Shen, C. Huang, and T. Wang, "Adaptive transmission power control for reliable data forwarding in sensor based networks," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–22, Mar. 2018.
[34] F. Cuomo, M. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. Pisani, "EXPLoRa: Extending the performance of LoRa by suitable spreading factor allocations," in *Proc. IEEE 13th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2017, pp. 1–8.
[35] F. Delobel, N. El Rachkidy, and A. Guitton, "Analysis of the delay of confirmed downlink frames in class b of LoRaWAN," in *Proc. IEEE 85th Veh. Technol. Conf. (VTC Spring)*, Jun. 2017, pp. 1–6.
[36] S. Cui and I. Joe, "Collision prediction for a low power wide area network using deep learning methods," *J. Commun. Netw.*, vol. 22, no. 3, pp. 205–214, Jun. 2020.
[37] F. Cuomo, D. Garlisi, A. Martino, and A. Martino, "Predicting LoRaWAN behavior: How machine learning can help," *Computers*, vol. 9, no. 3, p. 60, Jul. 2020.
[38] R. M. Sandoval, A.-J. Garcia-Sanchez, and J. Garcia-Haro, "Optimizing and updating LoRa communication parameters: A machine learning approach," *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 3, pp. 884–895, Sep. 2019.
[39] N. Aihara, K. Adachi, O. Takyu, M. Ohta, and T. Fujii, "Q-learning aided resource allocation and environment recognition in LoRaWAN with CSMA/CA," *IEEE Access*, vol. 7, pp. 152126–152137, 2019.
[40] D.-H. Kim, E.-K. Lee, and J. Kim, "Experiencing LoRa network establishment on a smart energy campus testbed," *Sustainability*, vol. 11, no. 7, p. 1917, Mar. 2019.
[41] A. Augustin, J. Yi, T. Clausen, and W. Townsley, "A study of LoRa: Long range & low power networks for the Internet of Things," *Sensors*, vol. 16, no. 9, p. 1466, Sep. 2016.
[42] J. C. Liando, A. Gamage, A. W. Tengourtius, and M. Li, "Known and unknown facts of LoRa: Experiences from a large-scale measurement study," *ACM Trans. Sensor Netw.*, vol. 15, no. 2, pp. 1–35, Apr. 2019.
[43] R. Pueyo Centelles, R. Meseguer, F. Freitag, L. Navarro, S. F. Ochoa, and R. M. Santos, "LoRaMoto: A communication system to provide safety awareness among civilians after an earthquake," *Future Gener. Comput. Syst.*, vol. 115, pp. 150–170, Feb. 2021.
[44] N. Chungsawat and P. Siripongwutikorn, "Predicting application performance in LoRa IoT networks," in *Proc. 11th Int. Conf. Adv. Inf. Technol.*, Jul. 2020, pp. 1–7.
[45] J. Duncombe, "Infrared navigation—Part I: An assessment of feasibility," *IEEE Trans. Electron Devices*, vol. ED-11, no. 1, pp. 34–39, Jan. 1959.
[46] A. S. Alam, Y.-F. Hu, P. Pillai, K. Xu, and J. Baddoo, "Optimal datalink selection for future aeronautical telecommunication networks," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 5, pp. 2502–2515, Oct. 2017.
[47] L. Wang and D. Binet, "MADM-based network selection in heterogeneous wireless networks: A simulation study," in *Proc. 1st Int. Conf. Wireless Commun., Veh. Technol., Inf. Theory Aerosp. Electron. Syst. Technol.*, May 2009, pp. 559–564.
[48] L. Wang and G.-S.-G. S. Kuo, "Mathematical modeling for network selection in heterogeneous wireless networks—A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 271–292, 1st Quart., 2013.
[49] L. Wang and D. Binet, "TRUST: A trigger-based automatic subjective weighting method for network selection," in *Proc. 5th Adv. Int. Conf. Telecommun.*, May 2009, pp. 362–368.

MOHAMMED ALENEZI received the B.Eng. and M.Sc. degrees (Hons.) in electronic engineering from the University of Hertfordshire, Hatfield, U.K. He is currently pursuing the Ph.D. degree with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. His research interests include wireless communications for the IoT applications, resource allocation optimization, and LPWAN technologies. He is a member of the Communication Systems Research Group, QMUL.

KOK KEONG CHAI (MICHAEL) received the B.Eng. (Hons), M.Sc., and Ph.D. degrees, in 1998, 1999, and 2007, respectively. He joined the School of Electronic Engineering and Computer Science (EECS), Queen Mary University of London (QMUL), in August 2008. He is currently a Reader of the Internet of Things and the Queen Mary Co-Director of Joint Programme with the Beijing University of Posts and Telecommunications (BUPT), and a member of the Communication Systems Research Group, QMUL. He has authored more than 65 technical journal and conference papers in the areas of machine to machine communications, wireless communications, the Internet of Things, and smart grids. His current research interests include sensing and prediction in distributed smart grid networks, smart energy charging schemes, applied blockchain technologies, dynamic resource management wireless communications, and medium access control (MAC) for machine to machine communications and networks.

ATM S. ALAM received the B.Sc. degree (Hons.) in information and communication engineering from the University of Rajshahi, Bangladesh, the M.Sc. degree (Hons.) in telecommunications and computer networks engineering from London South Bank University, U.K., and the Ph.D. degree in wireless communications from The Open University, Milton Keynes, U.K. He worked on several European and U.K., funded projects as a Research Fellow for the 5G Innovation Centre (5GIC), University of Surrey, U.K., and the University of Bradford, U.K. He is currently an Assistant Professor with the Queen Mary University of London, U.K. His research interests include the areas of intelligent wireless communications and networks, in particular, smart energy and spectrum-efficient radio resource management for 5G and beyond, the Internet of Things, and the emerging applications of machine learning in wireless communications for verticals, such as smart grids, intelligent transport systems, smart cities/homes, and industrial automation. He is also an Associate Fellow of Higher Education Academy.

YUE CHEN (Senior Member, IEEE) received the Ph.D. degree from the Queen Mary University of London (QM), London, U.K., in 2003. She is currently a Professor of Telecommunications Engineering with the School of Electronic Engineering and Computer Science, QM. Her current research interests include intelligent radio resource management for wireless networks, MAC and network-layer protocol design, cognitive and co-operative wireless networking, HetNets, smart energy systems, and the Internet of Things.

SHIHAB JIMAA received the M.Sc. and Ph.D. degrees in digital communications from Loughborough University, England, U.K., in 1986, and 1990, respectively. He also holds a Post-Graduate Certificate in Higher Education (PGCE) from the University of Hertfordshire, U.K., in 1998. He is currently an Associate Professor with the Khalifa University of Science and Technology, Abu Dhabi, United Arab Emirates. He has authored three book chapters, one patent, and about 100 papers in various refereed international conferences and journals. His research interests include wireless communication systems and networks, channel estimation and equalization, and signal processing for digital communications/biomedical engineering. He is a U.K. Chartered Electrical Engineer, a Senior Member of the IEEE Communications and Education societies, a member of IET, and a Fellow of the Higher Education Academy (HEA)/U.K. He is a Guest Editor of the Special Issue on Energy Efficient Wireless Communication Networks with QoS, *International Journal of Communication Systems* (John Wiley, 2017).