IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 30 December 2022, accepted 25 January 2023, date of publication 27 January 2023, date of current version 2 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3240308

RESEARCH ARTICLE

Intelligent Resource Allocation in LoRaWAN Using Machine Learning Techniques

SYED USAMA MINHAJ[1,2], AAMIR MAHMOOD[3], (Senior Member, IEEE), SARDER FAKHRUL ABEDIN[3], (Member, IEEE), SYED ALI HASSAN[2], (Senior Member, IEEE), MUHAMMAD TALHA BHATTI[2], SYED HAIDER ALI[2], AND MIKAEL GIDLUND[3], (Senior Member, IEEE)

[1]Department of Electrical Engineering and Information Technology, Technical University of Munich, Munich, 80333 Bavaria, Germany
[2]School of Electrical Engineering and Computer Science (SEECS), National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan
[3]Department of Computer and Electrical Engineering, Mid Sweden University, 851 70 Sundsvall, Sweden

Corresponding author: Aamir Mahmood (aamir.mahmood@miun.se)

This work was supported in part by the Knowledge Foundation Sweden (KKS) Research Profile Next Generation Industrial IoT (NIIT) and WICOM.

ABSTRACT With the ubiquitous growth of Internet-of-things (IoT) devices, current low-power wide-area network (LPWAN) technologies will inevitably face performance degradation due to congestion and interference. The rule-based approaches to assign and adapt the device parameters are insufficient in dynamic massive IoT scenarios. For example, the adaptive data rate (ADR) algorithm in LoRaWAN has been proven inefficient and outdated for large-scale IoT networks. Meanwhile, new solutions involving machine learning (ML) and reinforcement learning (RL) techniques are shown to be very effective in solving resource allocation in dense IoT networks. In this article, we propose a new concept of using two independent learning approaches for allocating spreading factor (SF) and transmission power to the devices using a combination of a decentralized and centralized approach. SF is allocated to the devices using RL for contextual bandit problem, while transmission power is assigned centrally by treating it as a supervised ML problem. We compare our approach with existing state-of-the-art algorithms, showing a significant improvement in both network level goodput and energy consumption, especially for large and highly congested networks.

INDEX TERMS Internet-of-Things (IoT), LPWAN, LoRaWAN, machine learning, network scalability, parameter selection, reinforcement learning.

I. INTRODUCTION

The proliferation of wireless technologies, especially for the Internet-of-things (IoT) ecosystem of remotely connected devices, has been unprecedented for the past few years. By 2027, it is expected that 30.2 billion IoT devices (including short-range, cellular, and other wide-range IoT segments) will be active around the world [1]. Because of low power requirements, long range and decent data rates, LPWANs technologies have emerged to support cost-effective and power-efficient wide-area connectivity of IoT devices. Some famous LPWAN technologies include LoRaWAN, Sigfox, LTE-M, and NB-IoT. Out of these, LTE-M and NB-IoT operate in licensed bands to provide reliable communication at comparatively higher data rates but access protocols require a lot of re-transmission and energy overheads. In contrast, LoRaWAN and Sigfox operate in unlicensed ISM bands requiring sophisticated interference control but enabling the establishment of large private and public networks at a much lower cost. Therefore, channel and radio resources must be utilized efficiently in these networks to enable massive IoT access.

LoRaWAN protocol stack uses LoRa as a physical layer technology, which can provide reliable communication under harsh link conditions. The modulation scheme used in LoRa is the chirp spread spectrum (CSS), in which each LoRa signal is split into multiple information pieces, and each piece is called a chirp. The frequency of the chirp

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

TABLE 1. A summary of various methods for LoRa network performance optimization.

Method	Reference	Year	Approach (algorithm or ML)	Parameter optimization	Performance metric	
					Goodput	Energy
ADR	EXPLoRa [2]	2017	Evenly distributing SF	SF, Power	x	
	FADR [3]	2018	Fair data extraction	SF, Power	x	x
	EARN [4]	2020	Greedy ADR, CR adaptation and adaptive SNR margin	SF, Power, CR	x	x
	CA-ADR [5]	2021	Collision-aware ADR	SF, Power	x	
	Extended ADR for Mobility [6]	2021	Linear regression	SF, Power	x	x
ML	[7]	2020	Deep Q-networks	SF, Power, CF	x	x
	[8]	2020	Deep RL	SF	x	
	LoRa-MAB [9]	2019	RL (EXP3s)	SF	x	
	STEPS [10]	2021	RL	SF	x	x
	[11]	2022	RL	SF	x	x
	[12]	2022	RL	SF, Power	x	x
	[13]	2022	DL	SF, Power		x
	This study	2022	Supervised ML and RL (EXP4)	SF, Power, CR, CF	x	x

increases linearly with time and the increment step is calculated using a transmission parameter called spreading factor (SF). Besides SF, LoRa Physical layer (PHY) provides other transmission parameters, including coding rate (CR), bandwidth (BW), center frequency (CF), and transmission power. These parameters give numerous unique tuneable combinations, impacting the network performance differently, e.g., extended communication range at the cost of higher energy consumption and lower data rate or higher data rate in exchange for a shorter range. Although LoRaWAN specifications [14] and regional parameters [15] reduce the controllable parameters in a specific region, their complex mutual interdependence affects the link performance differently, making it challenging to find a suitable configuration in dynamic channel conditions.

Two major shortcomings of a LoRaWAN network are, i) it operates on open Industrial, Scientific and Medical (ISM) band, which is shared with other technologies, and ii) it employs ALOHA protocol to support multiple access (MA) among end devices. The theoretical maximum capacity of an ALOHA network is only 18% [16]. Therefore, it is essential that an intelligent resource allocation mechanism must be employed to efficiently and dynamically allocate communication parameters. A LoRa network infrastructure can employ an adaptive data rate (ADR) mechanism to optimize the lifetime of devices, overall network capacity, and scalability [14]. To do so, ADR controls some of the transmission parameters, such as TP and SF, tooptimize transmission power and data rate while ensuring the energy-efficient and stable connectivity of the individual devices to the gateway. The parameters adaptation is not without a trade-off; using higher SF leads to a higher link budget (long-range connectivity) at the expense of higher packet airtime and lower data rate. In this respect, devices close to the gateway are preferred to use lower SF. However, increasing the device density of established LoRa networks means that there will be more devices transmitting at the same settings, leading to increased interference and potential loss of information. However, the ADR approach is shown to be sub-optimal for dense networks for its slow and conservative adaptation to the environment [17]. Therefore, it is imperative to develop a mechanism that analyzes the network and updates the parameters of end devices (EDs), considering existing traffic congestion when new nodes are added to the network. *To this end, in this research, we propose an intelligent resource allocation mechanism based on ML techniques to solve the interference and network congestion problem in LoRaWAN.*

II. PREVIOUS WORKS AND OUR CONTRIBUTIONS

The scalability analysis and enhancement of LoRaWAN networks have been an active area of research and development. The objective is to determine the impact of self-interference on network dimensioning (e.g., cell size, device density) and, consequently, to mitigate it by adopting various medium access efficient resource allocation or techniques. A few examples of medium access-related solutions are: exploiting time diversity [18] or successive interference cancellation (SIC) to minimize packet losses due to self-interference [19], or using different access mechanisms such as slotted ALOHA and collision avoidance (CCA)-based carrier sense multiple access (CSMA) [20], [21], [22]. Meanwhile, numerous techniques have been proposed to solve scalability issues by efficient resource allocation schemes, i.e., via adaptation of LoRa-PHY parameters. In this respect, the adaptive data rate (ADR) algorithm-based and ML-based methods are at the forefront, which inspired the development of many variants to improve the scalability of LoRa networks. We discuss these methods in the following subsections while selected solutions related to this study are summarized and differentiated in Table 1.

A. ADR ALGORITHM AND ITS VARIANTS

In LoRaWAN specifications [14], the ADR is referred to as a scheme used by the LoRa network infrastructure to individually adapt and optimize the data rate and transmit power of the devices. By appropriately selecting the transmission parameters, ADR helps in maximizing the lifetime of EDs and overall network capacity. ADR algorithm consists of two asynchronous routines: devise-side and network-side. The former routine is specified in LoRaWAN specifications for parameter selection if the device seems to have lost connection to a gateway. While the latter routine, which evaluates the link quality based on SNR, is not explicitly defined. Instead, its design is delegating, e.g., The Things Network (TTN) [23], implements a revised version of the algorithm recommended by Semtech [24].

However, the basic ADR's conservative policy, load distribution, and fairness restricts its scalability, and many improved versions of the algorithm have been proposed. For instance, the authors in [2] proposed a scheme to minimize collisions by evenly distributing the load across all SFs while exploiting the quasi-orthogonal property of SFs [25]. Similarly, a fair adaptive date rate (FADR) algorithm was proposed in [3] as an attempt to increase the fairness across EDs without considering their energy efficiency. Considering the problems with previous ADR, the authors in [4] developed an efficient algorithm for resource allocation, named *EARN*, which followed an approach similar to [2] for evenly distributing the load across all SFs but it also exploited CR for increasing robustness to noise. However, their solution works on the assumption of having full knowledge of all wireless devices operating on a frequency band. As LoRaWAN operates on the unlicensed ISM band, which is shared with other technologies, the presence of devices unknown to the network would be highly likely. These devices could significantly affect the performance and usefulness of the algorithm. In this respect, we show that, compared to the conservatism inherent in ADR algorithms, our intelligent resource allocation algorithm performs better than EARN in terms of energy per packet (EPP) and goodput due to the fast adaptability of the proposed learning-based algorithm to the environment. Another noteworthy work is the collision-aware ADR (CA-ADR) algorithm proposed in [5], which minimizes the probability of devices transmitting during the vulnerability period in ALOHA; however, the performance analysis metric is limited to packet success rate only.

Since a large class of IoT applications involves the mobility of devices, many research studies have also been proposed to improve the parameter allocation problems for such networks [26], [27]. In [26], a modified version of E-ADR is proposed for IoT networks with a known mobility pattern of devices. On the other hand, the proposed solution in [27] extended E-ADR for a sensor with undefined trajectories. Reference [6] Suggested to use linear regression to calculate the required SNR for communication and changing the SF and P_t accordingly. Hence, the required SNR value maximizes the PRR while minimizing the energy consumption. Yet, these approaches are not efficient for simultaneously handling both the stationary and mobile nodes, like in most IoT network applications, for their lack of adaptability in dynamic environments.

B. MACHINE LEARNING-DRIVEN TECHNIQUES

As machine learning (ML) has proven to be extremely useful for solving complex problems in wireless networks, many researchers have tried to solve the optimal resource allocation in LoRaWANs using ML techniques, including deep learning and reinforcement learning (RL) [7], [8], [9], [10], [11], [12], [28]. In RL algorithms, an agent (i.e., a gateway (GW) or base station (BS) for centralized while EDs for decentralized RL algorithms) tries to maximize the reward by choosing an appropriate action out of the action space. The study in [7] solved the problem using deep Q-networks and a mixture of a centralized and decentralized approach. However, the RL model proposed is too complex due to the large action space of ninety possible actions offered by LoRa-PHY parameters; consequently, the authors considered only thirty nodes for the analysis. In [8], the authors presented deep reinforcement learning-based solution, termed LoRaDRL, for SF allocation in dense LoRa networks. Although LoRaDRL indicated performance gains for mobile devices, it considered only one transmit power level and, therefore, was unable to exploit the achievable energy efficiency of LoRaWAN. In STEPS (Score Table-based Evaluation and Parameters Surfing) [10], the authors introduced an RL-based approach for SF optimization using a score table of probability for adapting the devices' parameters. Meanwhile, [11] extended STEPS by adopting an MDP-based approach for parameter initialization, enabling it to reduce network energy consumption. However, the solutions in [10] and [11] are limited to optimal SF allocation only. In [28], the authors used deep RL-based SF and channel assignment to minimize gateway energy consumption for networks powered by renewable energy and conventional grid. While in [12], the authors proposed a Q table-based adaptation strategy with a reward function defined in terms of SNR, SF, and goodput. Although [12] presented a novel approach for SF and power allocation, the performance gains are marginal compared to the conventional ADR algorithms.

While all the studies mentioned above all ALOHA-based access, for the sake of completeness, it is worth mentioning that resource allocation problem using Q-learning has also been studied for CSMA/CA-based access in LoRaWANs (e.g., see [29], [30]).

Meanwhile, the distributed allocation of radio resources has motivated to explore RL-based Multi-Arm Bandit (MAB) techniques in LoRaWAN, wherein each node act as an intelligent agent to select the best parameters to maximize its reward. In this direction, LoRa-MAB [9] utilized the popular EXP3 algorithm to solve the scalability problem inherited in centralized RL algorithms. In LoRa-MAB, EDs (acting as agents) aimed to maximize reward (i.e., packet reception ratio—PRR) by choosing the most suitable action from the action space and learning from it. The algorithm provided excellent results for SF allocation. However, by focusing only on actions suitable for PRR maximization, it fails to take into account the energy consumption of those actions. Furthermore, EXP3 requires an exceedingly large convergence time of around 200 k-hours of training, making it tedious and resource-consuming. In [31], the problem of long convergence time is addressed by modifying the algorithm selection pattern via switching of the parameters on the run. However, the convergence time still remained significantly long whereas it also decreased the overall throughput of the network since it used buffering.

C. OUR CONTRIBUTIONS

In this article, we propose a new concept of finding optimal SF and transmission power for EDs in a LoRa cell using two independent approaches to solve the dual objective of minimizing energy consumption while maximizing the PRR, which to the best of the authors' knowledge have not been explored before. In this respect, the main contributions of this article are as follows.

- We propose an algorithm to solve the energy per packet (EPP) minimization problem of a LoRaWAN network, which divides the problem into two independent problems of energy consumption minimization and PRR maximization.
- The energy consumption problem is solved using a centralized supervised ML-based approach for transmission power allocation. Our approach reduces the average energy consumption of a device by up to 370%, while making a minimal difference in computational requirements, with required computations being performed at the BS.
- SF allocation is treated as a contextual multi-arm bandit problem, which we solve for PRR maximization using a decentralized EXP4 RL algorithm. By using expert advice, the RL algorithm can converge quickly in just tens of hours as compared to thousands of hours of previous RL algorithms [9], [32].
- Goodput and energy per packet (EPP) are compared with a wide range of well-known algorithms from the literature for different network parameters to prove the effectiveness of our algorithm.
- Some extended versions of EXP4 are given by utilizing CR and CF allocations in special scenarios. Furthermore, a modified version of the algorithm for non-stationary nodes in an IoT network is also presented and evaluated.

III. SYSTEM MODEL

We consider a single-cell LoRa network, consisting of a fixed number of LoRa end devices (EDs) and a single half-duplex gateway (GW). The EDs are Class A devices, static (unless stated otherwise), and uniformly distributed around the GW. The EDs sleep most of the time to conserve battery, waking up to perform uplink transmissions only at new packet arrival instants. Also, during the network training phase, each ED receives a downlink acknowledgment (ACK) from the GW upon successful uplink transmission. To avoid interference between uplink transmissions and downlink ACKs, we assume that the GW transmits the ACK on a channel different than the uplink channel. For example, TTN allows the use of separate channels for downlinks with a 10% duty cycle [33]. Although LoRaWAN specifies two receive windows, RX1 and RX2, in which a device listens for confirmed traffic, we consider that EDs wait for an ACK only in the second receive window, RX2, to conserve energy and channel resources. In this respect, we assume that the gateway transmits ACKs at a fixed SF9 while it compensates for the possible link asymmetry for higher SFs by adopting the transmit power, $p_t = 27$ dBm.

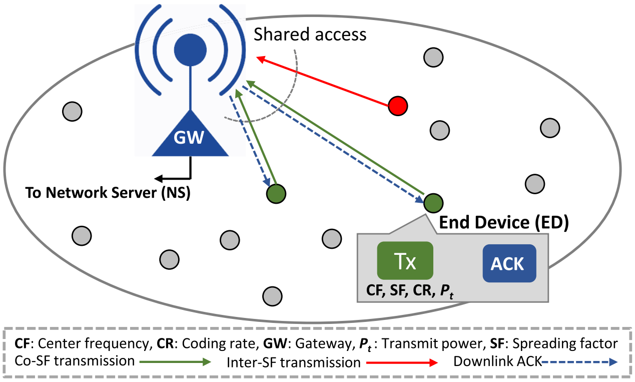


FIGURE 1. System model of a LoRa network.

For the uplink, to capture the most realistic radio environment and LoRa-modulation-specific details, we consider all the possible factors that result in packet loss during the uplink transmission. Such factors include wireless channel attenuation and fading, bit error rate (BER) [34], time collision during the critical window of packet transmission [35], and inter-SF, and co-SF capture effect [25].

A. PATH LOSS MODEL AND CHANNEL FADING

We consider the log-exponential path loss model along with Nakagami-m fading with shape parameter m for the wireless channel model as

$$P_L(d)[\text{dB}] = P_L(d_0) + 10n \log_{10}\left(\frac{d}{d_0}\right) + X_\sigma \qquad (1)$$

where P_L is the path loss at distance d, n is the path loss exponent, X_σ is the fading parameter, while $P_L(d_0)$ and d_0 are reference path loss and distances.

In earlier works on optimal parameter selection in LoRaWAN, mostly either fading or shadowing are ignored. However, this makes the channel much more predictive with fewer variations, and algorithms like ADR [36] and LoRa-MAB [9] can easily converge to optimal solutions. To make the assumptions more realistic, we consider Nakagami-m fading, which is generic enough to model a wide range of wireless channel conditions, including Rayleigh and Rician fading.

B. PERFORMANCE METRICS

The performance of an IoT network can be described by three primary factors, including PRR, energy consumption, and data rate. Out of these three, PRR and energy consumption are of primary importance for LPWANs such as LoRaWAN, as they are battery-powered networks and do not have high data rate requirements.

1) PACKET RECEPTION RATIO (PRR)

To determine whether a packet has been received, we use the analytical model of bit error rate (BER) of LoRa modulation [34]

$$\mathrm{BER} = Q\left(\frac{\log_{12}(\mathrm{SF})}{\sqrt{2}} \cdot \Gamma_b\right), \tag{2}$$

where $Q(\cdot)$ is the Gaussian Q-function and $\Gamma_b = E_b/N_o$ is the ratio between the energy per bit to noise power spectral density

$$\Gamma_b = \mathrm{SNR}(p_t) - 10 \cdot \log_{10}\left(\frac{R_s \cdot \mathrm{SF} \cdot \mathrm{CR}}{\mathrm{BW}}\right), \tag{3}$$

where SNR is a function of p_t and R_s is the symbol rate defined as

$$R_s = \mathrm{SF} \cdot \left(\frac{4}{4+\mathrm{CR}} \Big/ \frac{2^{\mathrm{SF}}}{\mathrm{BW}}\right) \cdot 1000. \tag{4}$$

Using the results from [4], this model can be extended to different CRs; that is, from (2), the probability of successful reception in the presence of noise, P_{ne}, for a k-bit packet can be determined as

$$P_{\mathrm{ne}} = \begin{cases} \left(1 - \mathrm{BER}(\Gamma_b)\right)^k, & \text{if CR} = 1, 2 \\ \left(1 - \mathrm{BER}(\Gamma_b)\right)^k + k & \\ \mathrm{BER}(\Gamma_b) \cdot \left(1 - \mathrm{BER}(\Gamma_b)\right)^{k-1}, & \text{if CR} = 3, 4. \end{cases} \tag{5}$$

On the other hand, in the presence of self-interference (i.e., collision), a packet can be considered successful if LoRa-specific co-SF and inter-SF capture threshold is satisfied

$$P_{\mathrm{ne+}} = \begin{cases} P_{\mathrm{ne}}, & \text{if inter-SF \& co-SF threshold is satisfied} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

i.e., $P_{\mathrm{ne+}}$ is zero if the collision happens and the packet is lost due to interference, and P_{ne} otherwise.

Let P_{nc} be the probability of no collision in the presence of co-SF and inter-SF interference; then the PRR can be approximated as [37]

$$\mathrm{PRR}(p_t) \approx P_{\mathrm{nc}} \cdot P_{\mathrm{ne+}}. \tag{7}$$

2) ENERGY CONSUMPTION

The energy consumption (EC) of a device for packet transmission and the corresponding ACK reception can be defined as

$$\begin{aligned} \mathcal{E}(\mathrm{SF}, \mathrm{CR}) &= \mathcal{E}_{\mathrm{PKT}} + \mathcal{E}_{\mathrm{ACK}} \\ &= p_t \cdot \mathrm{ToA}\,(\mathrm{SF}, \mathrm{CR}) + p_{\mathrm{rec}} \cdot \mathrm{ToA}\,(\mathrm{SF}(9), \mathrm{CR}(4/5))\,, \end{aligned} \tag{8}$$

where, in the first term, $p_t = V_{\mathrm{tx}} \cdot I_{\mathrm{tx}}$ is the transmit power of an uplink packet defined in terms of the supply voltage (V_{tx}) and p_t-dependent supply current (I_{tx}) of a typical LoRa transceiver (c.f. Table 2). Whereas, in the second term, $p_{\mathrm{rec}} = I_{\mathrm{rec}} \cdot V_{\mathrm{tx}}$ is the power consumed during an ACK receive window, with I_{rec} as the corresponding current supply, and SF and CR of ACKs are fixed at 9 and 4/5, respectively.

Moreover, the ToA of a packet in terms of LoRa symbol time, T_{sym}, can be defined as

$$\mathrm{ToA}(\mathrm{SF}, \mathrm{CR}) = \left(n_{\mathrm{pre}} + \tilde{n}(\mathrm{SF}, \mathrm{CR})\right) \cdot T_{\mathrm{sym}}, \tag{9}$$

where $T_{\mathrm{sym}} = 2^{\mathrm{SF}}/\mathrm{BW}$, which implies that a higher SF has a higher T_{sym} or ToA, yielding higher collision probability.[1] Also, n_{pre} and $\tilde{n}$ are the number of symbols in the preamble and payload of the frame, respectively, with $\tilde{n}$ defined as [38]

$$\tilde{n}(\mathrm{SF}, \mathrm{CR}) = 8 + \max\left(\left\lceil \frac{8\mathrm{PL} - 4\mathrm{SF} + 28 + 16C}{4(\mathrm{SF} - 2DE)} \right\rceil (\mathrm{cr}+4), 0\right),$$

where $\lceil \cdot \rceil$ is the ceil function, PL is the number of bytes in frame payload, C the indicator function for payload CRC, DE is the data rate optimization indicator (enabled only for SF11 and SF12), and lowercase cr the CR index for coding rate from 4/5 to 4/8.

3) ENERGY PER PACKET (EPP)

From (8) and (7), a combined performance metric energy per packet (EPP) can be derived as

$$\mathrm{EPP} = \frac{\mathcal{E}(\mathrm{SF}, \mathrm{CR})}{\mathrm{PRR}(p_t)}, \tag{10}$$

which defines the energy consumed for successful packet transmission to the gateway, as in [4].

C. PROBLEM FORMULATION

When maximizing the overall network performance, EPP in (10) nicely captures the tradeoff between PRR and energy consumption. Therefore, with an objective of maximizing network-wise PRR while minimizing energy consumption based on the appropriate selection of transmission parameters, we can formulate our problem as

$$\min_{a'} \ \frac{1}{N} \sum_{j=1}^{N} \mathrm{EPP}, \tag{11}$$

where a' is the set of the decision variables, $\{\mathrm{CR}, \mathrm{SF}, p_t, \mathrm{CF}\}$, where the range of these decision variables is according to the LoRa specifications [38]. However, the optimization problem (11) is a combinatorial and mixed-integer problem, which is known to be NP-hard. In addition, the decision variables CR, SF, and CF are discrete while p_t is continuous. Therefore, it is difficult to achieve optimal results in polynomial time. To simplify the problem, by assuming a uniform deployment of devices on CFs to ensure that each orthogonal

[1] For different SFs, ToA of a LoRa packet with 19-byte payload is given in Table 3.

channel has an equal number of nodes to minimize interference, we define a two-stage problem: First, we minimize the average energy consumption of a device with the appropriate selection of p_t for given SF and CR as

$$\begin{aligned} \min_{p_t} \; & \frac{1}{N}\sum_{j=1}^{N} \mathcal{E}(\mathrm{SF}, \mathrm{CR}) \\ \text{s.t. } \; & P_t \leq P_{\max} \\ & \mathrm{PRR} \geq 0.95, \end{aligned} \tag{12}$$

where $P_t \leq P_{\max}$ sets the constraint on maximum to transmit power, and constraint PRR ≥ 0.95 ensures that the optimal p_t must achieve a minimum PRR of 95% without interference.

Second, using optimal p_t (i.e., p_t^*), we maximize the network PRR by appropriate configuration of SF and CR, i.e.

$$\begin{aligned} \max_{a} \; & \frac{1}{N}\sum_{j=1}^{N} \mathrm{PRR}(p_t^*) \\ \text{s.t. } \; & \mathrm{CR} \in [5, 7] \\ & \mathrm{SF} \in [7 \cdot\cdot 12], \end{aligned} \tag{13}$$

where decision variable a is the subset of a' with $a = \{\mathrm{SF}, \mathrm{CR}\}$, whereas p_t^* is the output of the problem (12).

In the next section, we develop an algorithm to find the optimal parameter configuration of a LoRa network, where we solve problem (12) using supervised ML and problem (13) using contextual multi-armed bandit RL technique.

IV. ALGORITHM DESIGN AND DESCRIPTION

Note that, to solve problem (11), we could create an RL environment consisting of realizable states equal to the number of possible transmission parameters like those considered in [9] and [39]. However, it makes the action space exceptionally large, requiring a lot of time and computations to converge to an optimal solution [9]. Furthermore, choosing a suitable reward for such an RL algorithm is difficult as it has two conflicting goals (energy minimization and PRR maximization).

Therefore, we design an algorithm that minimizes the EPP by using two-stage optimization, i.e., supervised ML for minimizing energy consumption and RL for maximizing PRR, with the following blueprint.

- For CF, we consider a uniform deployment of devices on each CF to minimize interference (line 1 in Alg. 1).
- The energy consumption minimization is converted into a supervised machine learning problem, which outputs an optimal p_t matrix for each SF and CR.
- Lastly, SF and CR are allocated using the RL algorithm for the non-stochastic bandit problem EXP4 utilizing expert opinion-based actions. Each expert has a trust coefficient, which is updated based on the reward from following the expert's advice, with two experts; the first expert depends on ToA while the second exploits the packet history to learn the environment using EXP3s, as in [9].

Simply put, the ML algorithm ensures that packet loss due to low SNR is minimized, while the RL algorithm reduces packet losses due to interference or low signal-to-interference ratio (SIR). These two approaches are described in the following subsections, while the full algorithm is given in Alg. 1. The detailed evaluation and insights on the effectiveness of the selected approach are given in Sec. V.

Algorithm 1 Proposed Algorithm for Optimal Parameter Selection in a LoRa Network

```
Input: Distance (d) and Nakagami-m parameter of EDs
Output: Optimal SF, CR, CF, and $p_t$ for all EDs
1:  Assign CF using node-id
2:  Find $p_t$ matrix using $d$ & $m$
3:  Initialization:
4:  $w_{\text{exp3s}}(0) = 1$, $w_{\text{exp4}}(0) = 1$, $\mathcal{P}_{\text{exp3s}} = \frac{1}{K}$ as uniform
    distribution, and $\mathcal{P}_{\text{ToA}}$ from (16).
5:  EXP4 Training
6:  for $t = 1$ to $T$ do
7:      for each end device $j$ do
8:          if Transmit then
9:              Compute reward, $r_s^j(t) \leftarrow$ (18)
10:             Compute EXP3s weight, $w^j_{\text{exp3s}}(t+1) \leftarrow$ (17)
11:             Compute EXP3s prob., $\mathcal{P}^j_{\text{exp3s}_s}(t+1) \leftarrow$ (19)
12:             Compute expert advice matrix, $\mathcal{P}_\epsilon(t) \leftarrow$ (20)
13:             Compute EXP4 weights, $w_{\text{exp4}}(t+1) \leftarrow$ (21)
14:             Compute EXP4 prob., $\mathcal{P}_{\text{exp4}}(t) \leftarrow$ (23)
15:         end if
16:     end for
17: end for
```

A. SUPERVISED ML FOR POWER ALLOCATION

In the following, we give an overview of the power optimization problem and the different ways it can be solved. p_t is the most important factor in the energy consumption of an ED. Previously, RL algorithms [7], [9] have been used to find optimal p_t. However, while the ED remains stationary, a dynamic allocation method (e.g., RL-based) of p_t is unnecessary, and a single p_t value can be considered optimal at a specific SF, CR, and distance from the BS. Further, even if p_t can be considered to be discretized to six power levels in commercially available LoRa transceiver [38], the size of the action space increases six times, and correspondingly increasing the computations by the same factor as well as the convergence time. If the channel conditions remain the same, every time recalculating the p_t would be a waste of computational resources and less effective as the RL algorithm may try to explore values of p_t that are not optimal.

As shown in [40], path loss models often do not provide a fair estimate of path loss of the environment; consequently, they are unable to estimate the required minimum power to maintain an adequate PRR. The curve fitting techniques studied in [40] are shown to perform better in terms of reducing root mean square error (RMSE) between the actual and predicted values. For instance, using linear regression, we can get an RMSE error of 11.2% on validation data (with a 70:30 train/test split) (c.f., Sec. V-B). It demonstrates that the supervised ML algorithms can be better utilized in predicting the appropriate minimum p_t required for successful transmission. Our approach to analyzing different supervised ML models to get the desired result is described Sec. V-B.

Therefore, the power optimization problem is solved by treating it as a supervised ML problem. The ML algorithm is trained on the previous data generated from packet transmissions. The algorithm assumes that the location of nodes around the base stations is known and using which the distance from end-nodes to BS is calculated. The other parameter required is the m-parameter to get an estimate of the environment condition between the node and BS. It can either be previously known or calculated using the SNR of a few of the previous packets (around twenty for a fine estimate but a rough estimate of the channel is also adequate as it does not have much effect on the calculations). The algorithm first estimates the optimal power based on its environment condition and distance from BS to get optimal powers for each set of SF and CR. So every combination of SF and CR pairs has one optimal transmission power associated with it (line 1 in Alg. 1). A glimpse of power prediction for an ED located at $d = 2180$ m is given below.

$$\begin{array}{c c} & \begin{array}{cccccc} \mathrm{SF}_7 & \mathrm{SF}_8 & \mathrm{SF}_9 & \mathrm{SF}_{10} & \mathrm{SF}_{11} & \mathrm{SF}_{12} \end{array} \\ \begin{array}{c} \mathrm{CR}_5 \\ \mathrm{CR}_7 \end{array} & \begin{bmatrix} 17 & 14 & 11 & 8 & 5 & 5 \\ 14 & 11 & 11 & 8 & 5 & 5 \end{bmatrix} \end{array}$$

That is, for instance, if an ED at a distance d m chooses to transmit at SF_7 and CR_5, the transmission power should be 17 dBm for optimal transmission. The main benefit of this approach is that it reduces the action space of the RL algorithm by six times, as now the algorithm does not have to explore the optimal transmission power.

B. RL ALGORITHM FOR SF ALLOCATION

Although Supervised ML can be and has also been used to allocate SF in an IoT network [13], the model becomes highly dependent on the traffic load and becomes invalid just as the traffic load changes. Therefore, to tackle the SF allocation problem, the contextual multi-armed bandit problems facet of RL is used and explored, which requires no prior statistical assumptions regarding the channel. To facilitate RL, we create an RL environment consisting of realizable states equal to the number of possible transmission parameters and fully observable state space ($\mathcal{X}$). The EDs act as independent agents in a distributed, non-cooperative manner, unaware of each other, to select the best actions to maximize their reward. The reward of the action is based on the successful reception of the packet at BS, equal to one in case of success, while zero otherwise. Each device j selects an action, $s_t \in \mathcal{S}_{x(t)} \subset \mathcal{S}$, where $\mathcal{S} = \{s_1, \ldots, s_j\} = \{\mathcal{P}_{\mathrm{CR}_l, \mathrm{SF}_m}\}_{\mathcal{P}_{\mathrm{ToA}}^{|\mathrm{CR}| \times |\mathrm{SF}|}}$ is the selection of the parameters CR and SF according to [38].

Exploring such a large action space to maximize reward using the RL algorithm takes a lot of computation in every step and a significant time to reach a suitable state, as in [9]. To tackle this problem, we have reduced the action space to only twelve possible states. It is done by allowing the RL algorithm to choose between only SFs (six possible values) and CRs (two possible values), whereas p_t and the center frequency are determined as in the previous section. This approach has proven to be much better in terms of PRR and EPP, as shown in the results (see Sec. V).

The agents traverse the environment with the help of an Exponential-weight algorithm for Exploration and Exploitation using Expert Advice (EXP4). The algorithm EXP4 chooses the best strategy from the pool of Expert advice. Our approach consists of $N = 2$ external experts, described as follows.

1) EXPERT-1

The first expert is the time-on-air (ToA), as defined in (9), which allocates SF based on the ToA of data frames. For simplicity, using the fact that a higher SF has a higher T_{sym} and consequently higher collision probability, we allocate probabilities inversely proportional to symbol time, instead of actual ToA, to prioritize actions with lower collision probability, i.e.

$$\mathcal{P}_{\mathrm{ToA}} = \begin{array}{c} \begin{array}{cccccc} \mathrm{SF}_7 & \mathrm{SF}_8 & \mathrm{SF}_9 & \mathrm{SF}_{10} & \mathrm{SF}_{11} & \mathrm{SF}_{12} \end{array} \\ \begin{bmatrix} \frac{1}{2^7} & \frac{1}{2^8} & \frac{1}{2^9} & \frac{1}{2^{10}} & \frac{1}{2^{11}} & \frac{1}{2^{12}} \end{bmatrix} \end{array} \tag{14}$$

As a result, for instance, the likelihood of choosing SF7 is twice that of SF8 by Expert-1.

It is a fixed expert, so its probability does not change with time. The probability vector in (14) only depends on the SF that can be simply repeated for the possible number of actions for the RL algorithm; that is, if for two values of CR, the size of the probability vector changes from 6 × 1 to 6 × 2 as

$$\mathcal{P}_{\mathrm{ToA}} = \begin{array}{c} \begin{array}{cccccc} \mathrm{SF}_7 & \mathrm{SF}_8 & \mathrm{SF}_9 & \mathrm{SF}_{10} & \mathrm{SF}_{11} & \mathrm{SF}_{12} \end{array} \\ \begin{bmatrix} \frac{1}{2^7} & \frac{1}{2^8} & \frac{1}{2^9} & \frac{1}{2^{10}} & \frac{1}{2^{11}} & \frac{1}{2^{12}} \\ \frac{1}{2^7} & \frac{1}{2^8} & \frac{1}{2^9} & \frac{1}{2^{10}} & \frac{1}{2^{11}} & \frac{1}{2^{12}} \end{bmatrix} \end{array} \tag{15}$$

We normalized $\mathcal{P}_{\mathrm{ToA}}$ to produce a probability distribution that adds up to one as

$$\mathcal{P}_{\mathrm{ToA}} = \frac{\mathcal{P}_{\mathrm{ToA}}}{\sum_{i=1}^{6} P_{\mathrm{ToA}}}. \tag{16}$$

2) EXPERT-2

It is a Uniform expert that uses the EXP3s algorithm for SF allocation. The uniform expert starts with equal probability $\mathcal{P}_{\mathrm{exp3s}}(0) = 1/6$ for each SF and is updated at each time step depending on the packet success or failure, similar to as described in [9]. The weights of each action are first initialized to one (line 4), and once the algorithm starts, they are updated based on the reward received from the previous action as (line 10)

$$w^j_{\mathrm{exp3s}_s}(t+1) = w^j_{\mathrm{exp3s}_s}(t) \exp\left(\frac{\gamma_{\mathrm{exp3s}_s} r^j_s(t)}{K \mathcal{P}^j_{\mathrm{exp3s}_s}(t)}\right) + \frac{e\alpha}{K} \sum_{s=1}^{K} w^j_{\mathrm{exp3s}_s}(t), \tag{17}$$

where γ_{exp3s} is the learning rate of EXP3s algorithm and $K = |\mathcal{P}_{\mathrm{ToA}}|$ is the number of possible actions of selecting SF and

CR, which in our case can be 6 or 12 based on (14) or (15), respectively.

The reward $r_s^j(t)$ depends on successful reception of the packet (i.e., PRR) as

$$r_s^j(t) = \begin{cases} \dfrac{\mathrm{PRR}^j(t)}{\mathcal{P}^j_{exp3s_s}(t)} & \text{if action } s_t \text{ is chosen} \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Therefore, the reward is only one for an action if it is chosen and the chosen action also results in successful transmission. This reward type is called bandit feedback when the algorithm observes the reward for the chosen action only.

Based on these weights, the new probability is determined using (line 11)

$$\mathcal{P}_{\mathrm{exp3s}}(t+1) = (1-\gamma_{\mathrm{exp3s}})\frac{w^j_{\mathrm{exp3s}}(t+1)}{\sum_{i=1}^{K} K w^j_{\mathrm{exp3s}}(t+1)} + \frac{\gamma_{\mathrm{exp3s}}}{K}. \tag{19}$$

3) EXP4

Both experts provide an SF allocation probability vector as described above, which is used by EXP4 to calculate optimal probabilities by updating weights and rewards associated with each SF during the entire time horizon. The probability vectors are concatenated to form an Expert Matrix $\mathcal{P}_\epsilon(t)$ as (line 12)

$$\mathcal{P}_\epsilon(t) = \begin{bmatrix} \mathcal{P}_{ToA} \\ \mathcal{P}_{\mathrm{exp3s}}(t) \end{bmatrix} \tag{20}$$

The weights of EXP4 are first initialized to one and updated based on reward as (line 13)

$$w^j_{\mathrm{exp4}}(t+1) = w^j_{\mathrm{exp4}}(t)\exp\left(\frac{\gamma_{\mathrm{exp4}}\sum_{i=1}^{2}\hat{y}_i(t)}{K}\right), \tag{21}$$

where γ_{exp4} is the learning rate of EXP4 algorithm and its optimal value is found to be 0.05 after experimenting with different values. Here, $\hat{y}(t)$ is calculated using matrix multiplication of reward matrix and expert advice matrix $\mathcal{P}_\epsilon(t)$ as

$$\hat{y}(t) = \widehat{\mathrm{reward}}_{\mathrm{exp4}}(t) \cdot \mathcal{P}_\epsilon(t), \tag{22}$$

where $\widehat{\mathrm{reward}}_{\mathrm{exp4}}(t)$ is the reward matrix found by combining the reward of each action (whether chosen or not).

Similar to exp3s, after getting the weights of each action, EXP4 uses these weights to calculate the final probability vector to select the next action as (line 14)

$$\mathcal{P}_{\mathrm{exp4}}(t) = \left(1-\gamma_{\mathrm{exp4}}\right)\frac{\sum_{i=1}^{2} w^j_{\mathrm{exp4}_i}(t)\mathcal{P}_\epsilon(t)}{\mathrm{W}_{\mathrm{exp4}}} + \frac{\gamma_{\mathrm{exp4}}}{K}. \tag{23}$$

C. COMPUTATIONAL REQUIREMENTS

A prime concern in deploying a specific algorithm for an IoT network is its computational requirements. IoT devices are usually constrained both in computational capabilities and available battery, while BS stations are assumed to have adequate power to support complex computations. Many previously proposed approaches used centralized RL based on deep RL and deep Q-Networks, which have very high computational requirements and could not scale with the increasing number of network nodes. However, decentralized algorithms do not have a scalability problem, as each node is independent of all other nodes. Therefore, these algorithms work effectively for massive networks without additional resource requirements. Moreover, EXP4 only requires a few multiplications and exponentiation calculations which are easy to implement in a small micro-controller in an end node without additional hardware requirements.

Mathematically, the upper bound on the regret of EXP4 algorithm is $\mathcal{O}(\sqrt{TM\log M})$ when $M \leq A$, where M is the number of experts, A is the number of arms (actions), and T is the number of steps in the time horizon [41]. When $M \leq A$ ($M = 2$ and $A = 12$ in our case), the regret bound of EXP4 is always better than Exp3. Concerning computational complexity, the most computationally expensive step used in the algorithm is exponentiation. The complexity of exponentiation is found using Taylor Series $\mathcal{O}(M(n)n^{1/2})$, where $M(n)$ is the complexity of the multiplication algorithm. For faster multiplication in embedded systems, Karatsuba Algorithm can be used [42], having a complexity of $\mathcal{O}(n^{1.585})$. Therefore, the final complexity becomes $\mathcal{O}(n^{1.585}.n^{1/2}) = \mathcal{O}(n^{2.085})$.

Thus, these properties make our algorithm easy and efficient to implement in highly resource-constrained IoT networks.

V. EXPERIMENTAL SETUP, ANALYSIS, AND RESULTS

A. EXPERIMENTAL SETUP

To establish the experimental setup according to the system model in Sec. III, we modified LoRaSim [43]—a discrete-event python simulator for LoRaWAN IoT networks—and incorporated the proposed ML-based parameter allocation approach. The salient features of the interference model of LoRaSim are: a) it considers both co-SF and inter-SF interference b) the message is received correctly if it satisfies minimum co-SF, inter-SF, and SNR threshold. c) a message is lost only if the overlap of packets is in the time-critical region of the considered packet [35]. The other changes to the simulator include the introduction of a) packet success model according to (5)-(7), b) Nakagami-m fading model for better realization of the real-world environment, c) optimal power allocation using supervised ML, and d) uniform CF allocation in each frequency channel and e) implementation of EXP4 to calculate optimal SF and CR. The LoRa- and system-specific parameters used in the simulation are given in Table 3 and Table 2. The results presented of the proposed algorithm are after twenty-four hours of training the reinforcement learning model in the network unless specified otherwise.

B. SUPERVISED ML FOR POWER ALLOCATION

For training our ML model, we generate the experimental data using the simulator as described earlier. In the

TABLE 2. LoRa- and simulation-specific parameters.

Parameter	Symbol	Value
LoRa-specific parameters [38]		
Transmit power	p_t	[5, 8, 11, 14, 17, 20] dBm
Spreading factor	SF	[7, 8, 9, 10, 11, 12]
Center frequency	CF	[868.1, 868.3, 868.5] MHz
Coding rate	CR	[5, 7]
Supply voltage	V_{tx}	3.3 V
Supply current in Rx	I_{rec}	10.5 mA
Supply current w.r.t. p_t	I_{tx}	[16.3 18.5, 23, 31.7, 90, 125] mA
Simulation-specific parameters		
Cell radius	R	4.5 km
Bandwidth	BW	125 kHz
PHY payload	PL	50 bytes
ACK payload	pl	8 bytes
# of devices	N	500
Traffic load	λ	0.83

TABLE 3. LoRaWAN uplink characteristics for a packet size of 19 Bytes (13-Byte header, 6-Byte payload) [38].

SF i	ToA t_i (ms)	Bitrate Rb_i (kpbs)	Rx Sensitivity S_i (dBm)	SNR threshold ψ_i (dB)
7	51.46	5.46	-123	-6
8	102.91	3.12	-126	-9
9	185.34	1.75	-129	-12
10	329.73	0.97	-132	-15
11	741.38	0.53	-134.5	-17.5
12	1318.91	0.29	-137	-20

simulations, the minimum transmits power, which ensures at least 95% PRR, is categorized as the optimal power. Meanwhile, a low network activity rate of $\lambda = 0.28$ is chosen to ensure that packet losses due to interference are negligible. The motivation was to capture the effect of the change in received SNR while changing the distance and the Nakagami-m parameter adequately. The data is then used to train the ML models with optimal power as the output and SF, CR, Nakagami-m parameter, and distance as the input.

For this purpose, we evaluated different ML algorithms to solve the problem of power optimization. The most obvious solution is to treat it as a regression problem with optimal power as the predicted label. Earlier studies in the literature have also shown that curve-fitting algorithms are much better at predicting path loss (and eventually optimal p_t) [40]. Since LoRa-PHY provides only six discrete possible transmit power values (see Table 2), a continuous output from the regression model is not entirely useful. Therefore, we treated the problem as a classification problem, with the output predicting either one of the possible power values or returning -1 in case there is no optimal power (i.e., indicating transmission is unfeasible for the given SF and CR).

To solve the classification problem of finding suitable p_t among seven available choices, we consider six popular classification algorithms, including random forest (RF), logistic regression, Gaussian naive Bayes, support vector machines (SVMs), linear discriminant analysis (LDA) and K-nearest neighbors classifier (KNNs). The algorithms are chosen considering the ease of deployment in an IoT network and low computational requirements, unlike neural networks.

TABLE 4. Classification accuracy of algorithms for power allocation problem.

Algorithm	Train Accuracy (%)	Test Accuracy (%)
Random Forest (RF)	96.22	92.96
Support Vector Machines (SVM)	90.37	90.10
Logistic Regression	81.36	81.12
K-nearest Neighbors Classifier (KNN)	93.58	91.58
Linear Discriminant Analysis (LDA)	80.69	81.22
Gaussian Naive Bayes	93.58	75.11

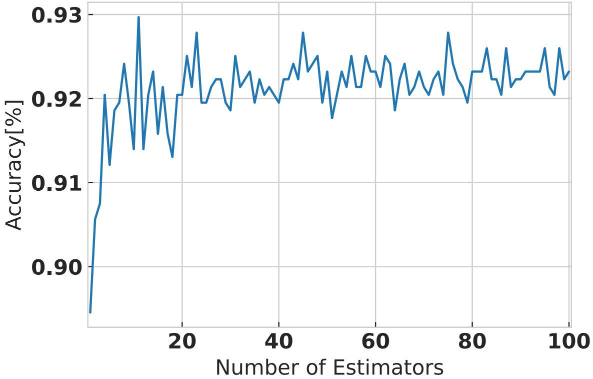


FIGURE 2. Classification accuracy of random forest (RF) algorithm for optimal power allocation.

For evaluation, we first divided the data into 70:30 train/test data split and then conducted cross-validation to find the best hyper-parameters. After learning about the training data, we tested these algorithms to find the accuracy of the test data.

The training and testing accuracy of the six considered algorithms is summarized in Table 4. It can be observed from these results that RF has the best accuracy of 92.96% compared to other solutions. For RF, the best result is obtained using 11 estimators with a random grid search. Fig. 2 shows the accuracy of the RF algorithm for the different numbers of estimators.

C. REINFORCEMENT LEARNING FOR SF AND CR ALLOCATION

In this section, we evaluate and compare the proposed algorithm with LoRa-MAB [9] in terms of packet reception ratio (PRR) and energy per packet (EPP). For training and testing, we simulate the IoT network for around eleven days, during which approximately 1.57 million packets are sent from the end nodes to the LoRa BS.

1) CONVERGENCE ANALYSIS USING PRR

The algorithm's convergence performance with time in terms of PRR is shown in Fig. 3. It can be observed that the EXP4-based proposed algorithm provides a much higher PRR even at the start of the network, proving it does not need much information about the environment to choose the best action due to multiple contextual experts. Although there are some variations at the start because the algorithm is trying to explore the best actions for maximum reward, after a few hours, the algorithm converges to a few actions and starts to exploit. In contrast, EXP3s/EXP3 perform poorly at the start

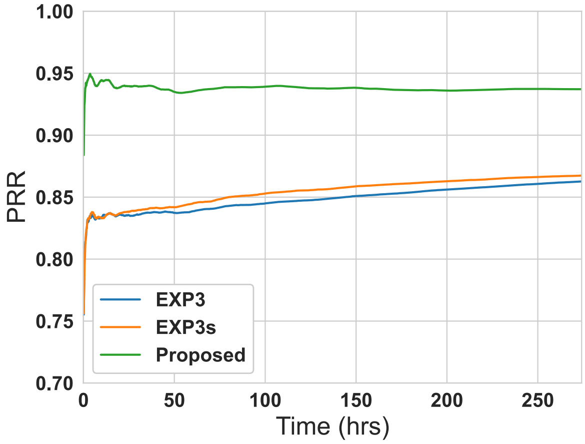


FIGURE 3. Convergence analysis of the proposed and the baseline RL algorithms.

as it, a) initializes the probability vector uniformly and b) do not have any contextual information about the network. The performance of EXP3s and EXP3 starts to improve slowly after some time as they learn about the environment through exploration; however, the curve becomes flat at the long-time horizon. From these results, we can conclude that the EXP4-based algorithm converges much faster than previously considered RL algorithms (approximately ten times faster than EXP3s). Meanwhile, it also provides significantly better network performance after convergence. This is because it already has contextual information about the environment through Expert 2 even before the start of network simulation.

2) ENERGY PER PACKET (EPP)

We also analyze the energy consumption in terms of the EPP of the RL algorithms with time. LoRa-MAB [9] provides EXP3s as a single objective RL algorithm, aiming to maximize PRR only. The consequence of this is that the algorithm sacrifices energy efficiency in its aim to maximize PRR. From Fig. 4, it can be observed that EXP3s consumes 294% more EPP than the proposed algorithm, which reduces slightly to 282% after nine days of learning. As described previously, due to the multi-objective nature of our algorithm, the energy consumption of our algorithm is also minimized by using a supervised ML algorithm to find the optimal power required to transmit the packet successfully without wasting the precious energy of the nodes and sacrificing the PRR. Lastly, we observed that the impact of downlink ACKs on the device energy consumption is insignificant due to the short ToA of an ACK and small current consumption I_{rec} during receive window.

3) SF AND POWER DISTRIBUTION

From Fig. 5a, we can distinguish the SF distribution of the sent packets between EXP3s and the proposed algorithm. EXP3s have a nearly uniform distribution of load for all SFs, which leads to increased collisions as low SFs can accommodate much more traffic load before congestion than higher SFs due to the significant difference in ToA. The higher ToA means it takes longer to transmit a packet, which also increases the probability of other nodes transmitting in the same time period; hence, leading to a collision between the packets. For the suggested algorithm, we observe that it assigns much more packets to lower SF and successfully exploits the lower ToA of low SFs based on Expert 2 advice. In Fig. 5b, we can observe the power distribution of the proposed and the baseline RL algorithms. As expected, EXP3s have uniform power distribution as there is no reward for the RL for choosing low powers even if higher power levels are unnecessary for successfully transmitting the packet. However, the transmission power in our algorithm is selected by the supervised ML algorithm, which minimizes the power by choosing the minimum required power to successfully transmit the packets.

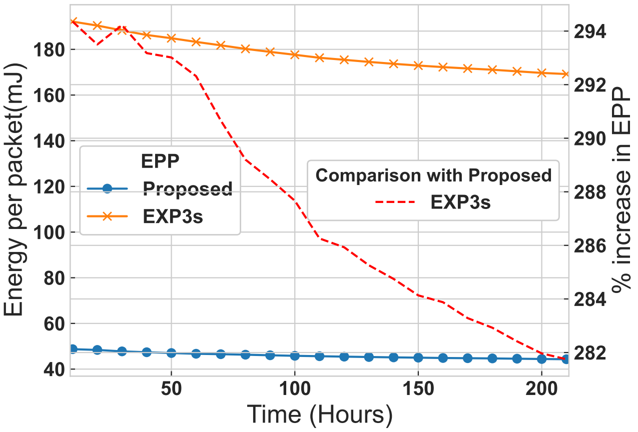
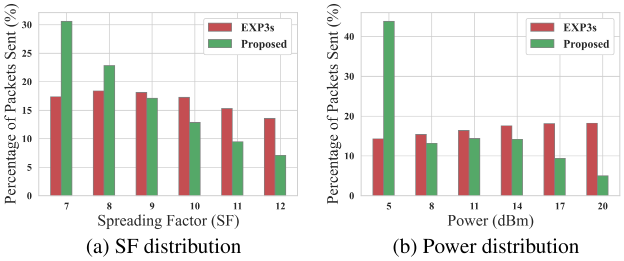


FIGURE 4. EPP for EXP3s and proposed.



(a) SF distribution (b) Power distribution

FIGURE 5. Resource distribution difference between EXP3s and the recommended algorithm.

D. COMPARISON WITH THE ADR EXTENSIONS

We also compare our algorithm with ADR and its two popular extensions, namely, EARN [4] and FADR [3], for different network cell sizes and activity rates. To do so, we modified our system model to match the one adopted in [4]. In particular, we utilized path loss model with $d_0 = 1000\,\text{m}$, $P_{L,d_0} = 128.95\,\text{dB}$, $n = 2.32$ and path loss std dev. $\sigma = 3$ as path loss parameters. Other simulation parameters are the same as given in Tables 2 and 3.

To compare with the baseline ADR schemes, we use the goodput metric, which is another important parameter describing the performance of IoT networks in terms of application-level throughput. The goodput be defined in terms of payload (PL), PRR, and the network activity rate λ, as

$$\text{Goodput} = \text{PRR} \cdot \text{PL} \cdot \lambda, \tag{24}$$

where λ is the traffic load, defined as

$$\lambda = \frac{\text{Number of nodes } (N)}{\text{Avg. send time of end-node}}. \tag{25}$$

1) CELL RADIUS

The goodput and EPP of the network for different cell radii are shown in Fig. 6a and Fig. 6b, respectively, with the considered algorithms. From Fig. 6a, we can observe that the recommended approach is much superior to other algorithms in terms of goodput at all distances. Furthermore, the effect of increasing cell radius is the least on the proposed approach and gives about 57% improvement on EARN-am [4] at 12.02 km. It can be attributed to the optimal power algorithm, which provides suitable power predictions based on the distance of the devices from the BS. When analyzing energy efficiency in terms of EPP for different cell sizes in Fig. 6b, our algorithm is observed to perform slightly better than EARN-am and EARN at smaller radii. However, from 5.71 km, EARN consumes the least EPP out of all the considered algorithms. It is because the proposed algorithm prefers a better PRR in the trade-off between energy consumption and PRR. If a lower EPP is required for the proposed approach, it can be achieved by sacrificing the PRR and goodput slightly in favor of energy efficiency.

2) TRAFFIC LOAD

In Fig. 7a and Fig. 7b, we compare the goodput and EPP under different traffic loads, respectively. Fig. 7a shows that the goodput of the proposed approach increases linearly with increasing traffic load. The results demonstrate the ability of the proposed algorithm to effectively learn the environment, depending on the traffic. By evenly distributing network traffic and exploiting the SF orthogonality using RL, the algorithm can minimize packet losses at high traffic loads. In contrast, EARN-am, EARN, and FADR have their maximum goodput at $\lambda = 1.39$, which decreases afterward, indicating high packet losses as the traffic load is increased. As the network radius remains constant in Fig. 7b, the EPP of the suggested approach remains constant, as increasing traffic load does not decrease the PRR due to effective learning of the environment by the model, as described previously. In contrast, the EPP of other algorithms increases exponentially due to higher packet losses and, consequently, lower PRR. At 12.02 km, the second-best performing algorithm EARM-am has 26 times higher EPP than the suggested algorithm. Our results show that the suggested approach performs much better than all other considered algorithms in highly congested networks.

E. PROPOSED APPROACH VARIATIONS

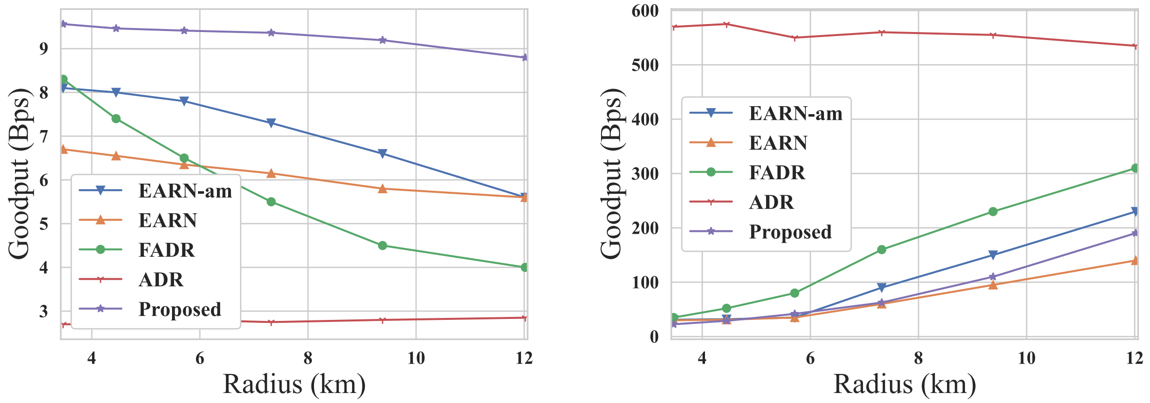
We also analyzed a selected few variations of the proposed algorithm using different strategies for CR and CF allocation and their usage scenarios. For our results, we considered an IoT network with $N = 1000$ nodes with a traffic load of $\lambda = 1.67$ and different cell radii.

1) CODING RATE (CR)

LoRa employs forward error correction for error correction using Hamming codes and has four possible coding rates, including 4/5, 4/6, 4/7, and 4/8. A higher value of CR increases robustness to noise but also increases the power consumption and the probability of collision due to a slightly higher ToA. Fig. 8 and Fig. 9 show the comparison between using two CRs (4/5 and 4/7) and only one fixed CR (i.e., 4/5 as mandated by LoRaWAN specifications [14]). The former provides an improvement from 6.33% to 13.3% in EPP depending on the cell radius while negligible performance difference in PRR. However, having two choices CR increases the action space of the recommended approach to twelve (6 SFs x 2 CRs) and, consequently, doubles the computational requirements. Hence, it is a design choice and depends on the specific conditions of the network. In most cases, doubling the computations does not have a considerable effect on the battery life, and improvement in EPP is more significant, especially for large networks. Therefore, we have used two CR choices (4/5 and 4/7) to allocate the transmission parameters.

2) CENTER FREQUENCY (CF)

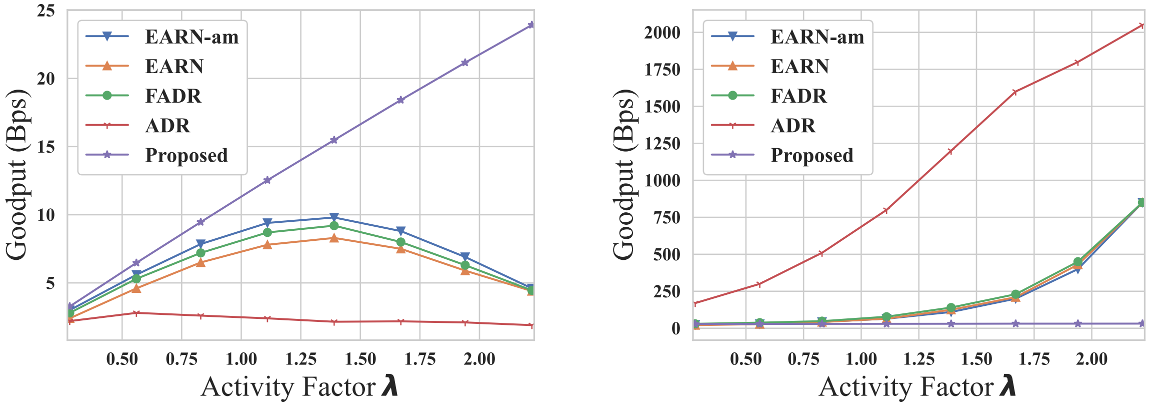
The LoRaWAN channel plans are region specific; for instance, in Europe, EU 863-870 MHz, known as the EU868 band, is harmonized in all EU countries under the ETSI [EN300.220] standard. As per LoRaWAN regional parameters [15], the EU868 ISM band supports a maximum of 16 channels, which are stored by EDs using a channel data structure. However, every end device must implement three default channels with center frequencies (CFs) of [868.10, 868.30, 868.50] MHz and maintain a list of 5 optional channels. The other channels can be freely modified/populated into 5 optional channel list (i.e., `CFlist`). To this end, for the comparative analysis with the related studies, we consider only these default channels in the EU868 band with a bandwidth of 125 kHz. From Fig. 8 and Fig. 9, we observe the difference in performance when RL is used to assign CF in contrast to uniform CF assignment to nodes. We note that using either of the above-mentioned approaches provides negligible improvement or performance degradation. However, if we assign CF using RL, it further increases the action space three times (as it would have three more choices to decide from). Therefore, the computations performed by the end nodes are tripled, which wastes precious battery life without providing any benefit. Hence, we have chosen a uniform frequency allocation strategy to allocate the same number of nodes in all channels. However, as LoRaWAN operates on the ISM band, sometimes there can be other wireless technologies present in the channel, of which the network would be unaware. In these specific scenarios, RL is a better choice as it does not require any prior knowledge about channel congestion.



(a) Goodput

(b) EPP

FIGURE 6. Comparing the proposed algorithm with the ADR-based schemes for different cell radii for traffic load $\lambda = 0.83$.



(a) Goodput

(b) EPP

FIGURE 7. Comparing the proposed algorithm with the ADR-based schemes for different traffic loads for cell radius $R = 4.5$ km.

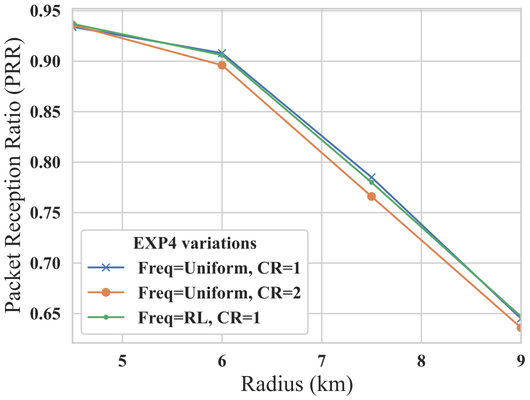


FIGURE 8. Packet reception ratio (PRR) for different proposed algorithm variations, for $N = 500$ EDs.

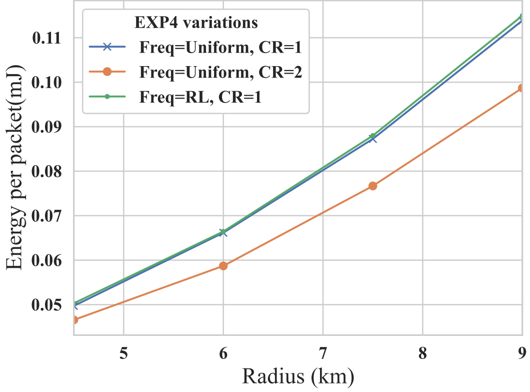


FIGURE 9. Energy per packet (EPP) for different proposed algorithm variations, for $N = 500$ EDs.

F. MOBILE NODES

Although our algorithm is primarily for stationary IoT devices, it can be extended to mobile nodes with a few modifications. Many IoT applications require mobility, e.g., connected farms and vehicular networks. The significant change for mobile nodes is in p_t allocation as the distance from the BS and environment keeps changing with time, and p_t needs to be updated similarly.

To ensure good performance for mobile nodes, we make slight modifications to our algorithm. To prove the validity of our modified algorithm for mobile nodes, we consider a cell radius of $R = 5$ km, $\lambda = 1.22$, $N = 300$ nodes. Out of $N = 300$, 50% nodes are mobile and move freely with a velocity of 50 km/h inside the network. We assume that although the end nodes are mobile, their distance from the gateway is still known within the accuracy of 250m (although

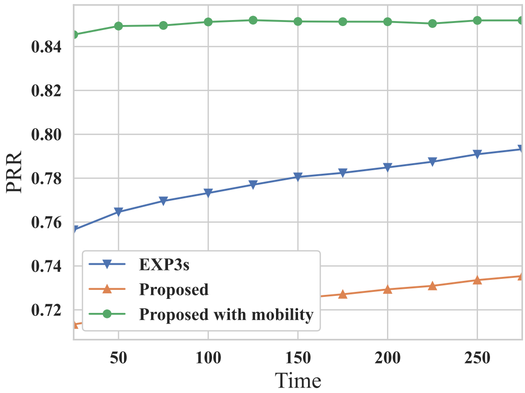


FIGURE 10. PRR with 300 nodes and 50% mobile nodes.

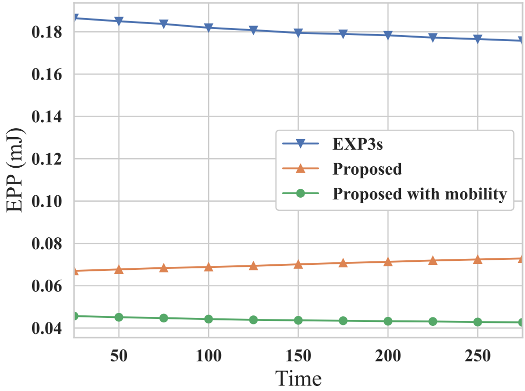


FIGURE 11. EPP with 300 nodes and 50% mobile nodes.

much higher ranging accuracy has already been demonstrated in LoRaWANs [44]. Furthermore, the node remains inside the cell radius of the IoT network. The allocation of SF remains the same as previously, while a fixed CR of 4/5 is selected. The main difference is in p_t allocation; we find multiple power matrices at the start of network simulation instead of just one previously since the distance from the gateway keeps changing. For our system model, we calculated five power levels for different distances such that the network radius is uniformly divided. Now, depending on the distance from the BS, one of the five levels is chosen to get optimal p_t.

To test our modified algorithm, we considered our unmodified algorithm, EXP3s [9], and modified the proposed algorithm. From Fig. 10, we observe that the modified suggested algorithm performs the best out of the three algorithms followed by EXP3s. The unmodified recommended algorithm performs worse as its power allocation is dependent on distance, which keeps changing, and the change is not catered for in the algorithm. Therefore, many packets are lost if the node moves far away from the gateway when the initial P_t was calculated for distance closer to the gateway. Fig. 11 proves that the modified proposed algorithm is also the most energy-efficient algorithm due to its dynamic power allocation. As expected, EXP3s performs worst as it is a single-objective algorithm based on PRR. These results prove that a slight modification in the algorithm for mobile nodes allows our algorithm to perform very well compared with other algorithms.

VI. CONCLUSION

In this article, we focused on improving the performance metrics of a single-cell, congested LoRaWAN network in the dynamic channel and deployment scenarios. In this respect, we developed an analytical model for energy-per-packet (EPP) while accounting LoRa PHY parameters, with the EPP model acting as a measure to capture the trade-off between packet reception ratio (PRR) and energy consumption (EC). Herein, the PRR and EC are crucial key performance indicators (KPIs) in low-power IoT networks. Using the EPP model, we defined an optimization problem to maximize the overall network PRR while minimizing the EC, with the available degrees of freedom in selecting LoRaWAN parameters as decision variables. To simplify the original (combinatorial and mixed-integer) NP-hard problem, we defined a two-stage problem. First, to minimize average energy consumption per device, we determined the appropriate transmit power of devices to achieve at least 95% PRR for the given LoRaWAN parameters by adopting a supervised ML approach. To this end, we generated realistic experimental data with different fading parameters to compare the appropriate transmit power level prediction performance of three classification algorithms: random forest (RF), logistic regression, and Bayesian regression, with RF (nineteen estimators) providing more than 91% accuracy. This approach allowed us to reduce the action space in solving the second problem of maximizing PRR by finding an appropriate configuration of spreading factor and coding rate parameters. We mapped the PRR maximization problem to the contextual multi-armed bandit RL technique, specifically the EXP4 algorithm with two experts. As a result, the algorithm is able to converge around a hundred times faster than previously proposed MAB approaches, and the computations required at the resource constraint end nodes are also significantly reduced.

We developed an algorithm based on these two independent solutions and compared its performance with state-of-the-art algorithms under a realistic simulation setup, accounting for wireless channel fading and time/power capture effect. Our results showed that the proposed algorithm could achieve higher energy efficiency, PRR, and goodput of the LoRa network, especially for large and highly congested networks (approximately 26 times better energy efficiency than other state-of-the-art algorithms for highly congested networks) in fixed as well as mobile device deployment scenarios.

Despite the superior performance of the proposed algorithm, it requires feedback (i.e., ACK) from the GW for every uplink packet from EDs in order to update its reward and, consequently, probabilities for the next action. As a result, precious channel/bandwidth and energy resources are used for downlink ACKs, which could also potentially lead to uplink/downlink interference when using the same channel. Therefore, in the future, we aim to study how we can reduce the algorithm's dependency on feedback from the GW as well as how to accurately model the uplink/downlink interference. Still, it is noteworthy that the downlink ACKs are required only during the network training phase. Our results have shown that 24-hour long training is sufficient to achieve good performance of the algorithm. This training time is relatively short compared to the IoT network's lifetime and the downlink control ACKs can be eliminated after training.

Moreover, the proposed algorithm can potentially be adapted to a multi-gateway scenario, e.g., to alleviate the congestion/interference on the gateways by appropriate parameter selection. Further, the effect on the network performance under different or more than two experts can also be studied.

REFERENCES

[1] B. Harald. *Ericsson Mobility Report*. Accessed: Jun. 2022. [Online]. Available: https://www.ericsson.com/mobility-report

[2] 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.

[3] K. Q. Abdelfadeel, V. Cionca, and D. Pesch, "Fair adaptive data rate allocation and power control in LoRaWAN," in *Proc. IEEE 19th Int. Symp. 'World Wireless, Mobile Multimedia Netw.' (WoWMoM)*, Jun. 2018, pp. 14–15.

[4] J. Park, K. Park, H. Bae, and C.-K. Kim, "EARN: Enhanced ADR with coding rate adaptation in LoRaWAN," *IEEE Internet Things J.*, vol. 7, no. 12, pp. 11873–11883, Dec. 2020.

[5] R. Marini, W. Cerroni, and C. Buratti, "A novel collision-aware adaptive data rate algorithm for LoRaWAN networks," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2670–2680, Feb. 2021.

[6] V. Moysiadis, T. Lagkas, V. Argyriou, A. Sarigiannidis, I. D. Moscholios, and P. Sarigiannidis, "Extending ADR mechanism for LoRa enabled mobile end-devices," *Simul. Model. Pract. Theory*, vol. 113, Dec. 2021, Art. no. 102388. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X21000952

[7] G. Park, W. Lee, and I. Joe, "Network resource optimization with reinforcement learning for low power wide area networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2020, no. 1, p. 176, Sep. 2020, doi: 10.1186/s13638-020-01783-5.

[8] I. Ilahi, M. Usama, M. Omer Farooq, M. Umar Janjua, and J. Qadir, "Intelligent resource allocation in dense LoRa networks using deep reinforcement learning," 2020, *arXiv:2012.11867*.

[9] D.-T. Ta, K. Khawam, S. Lahoud, C. Adjih, and S. Martin, "LoRa-MAB: Toward an intelligent resource allocation approach for LoRaWAN," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[10] M. Chen, L. Mokdad, J. B. Othman, and J.-M. Fourneau, "STEPS–score table based evaluation and parameters surfing approach of LoRaWAN," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[11] M. Chen, L. Mokdad, C. Charmois, J. Ben-Othman, and J.-M. Fourneau, "An MDP model-based initial strategy prediction method for LoRaWAN," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2022, pp. 4836–4841.

[12] T. Fedullo, A. Morato, F. Tramarin, P. Ferrari, and E. Sisinni, "Smart measurement systems exploiting adaptive LoRaWAN under power consumption constraints: A RL approach," in *Proc. IEEE Int. Workshop Metrol. Ind. 4.0 IoT (MetroInd4.0&IoT)*, Jun. 2022, pp. 354–359.

[13] L.-T. Tu, A. Bradai, O. B. Ahmed, S. Garg, Y. Pousset, and G. Kaddoum, "Energy efficiency optimization in LoRa networks—A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 21, 2022, doi: 10.1109/TITS.2022.3183073.

[14] *LoRaWAN^{TM} 1.1 Specifications*, LoRa Alliance, Beaverton, OR, USA, Jul. 2017.

[15] *RP002-1.0.3 LoRaWAN Regional Parameters*, LoRa Alliance, Fremont, CA, USA, May 2021.

[16] N. Abramson, "The ALOHA system: Another alternative for computer communications," in *Proc. Fall Joint Comput. Conf.*, New York, NY, USA, 1970, pp. 281–285, doi: 10.1145/1478462.1478502.

[17] S. Li, U. Raza, and A. Khan, "How agile is the adaptive data rate mechanism of LoRaWAN?" in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 206–212.

[18] A. Hoeller, R. D. Souza, O. L. Alcaraz Lopez, H. Alves, M. de Noronha Neto, and G. Brante, "Exploiting time diversity of LoRa networks through optimum message replication," in *Proc. 15th Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2018, pp. 1–5.

[19] S. U. Minhaj, S. H. Ali, M. T. Bhatti, S. A. Hassan, A. Mahmood, and M. Gidlund, "How SIC-enabled Lora fares under imperfect orthogonality?" in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, Jun. 2021.

[20] T. Polonelli, D. Brunelli, and L. Benini, "Slotted Aloha overlay on LoRaWAN—A distributed synchronization approach," in *Proc. IEEE 16th Int. Conf. Embedded Ubiquitous Comput. (EUC)*, Oct. 2018, pp. 129–132.

[21] T.-H. To and A. Duda, "Simulation of LoRa in NS-3: Improving LoRa performance with CSMA," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7.

[22] L. Beltramelli, A. Mahmood, P. Osterberg, and M. Gidlund, "LoRa beyond ALOHA: An investigation of alternative random access protocols," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 3544–3554, May 2021.

[23] The Things Network. (Oct. 2019). *LoRaWAN Adaptive Data Rate*. Accessed: Nov. 18, 2022. [Online]. Available: https://www.thethingsnetwork.org/docs/lorawan/adaptive-data-rate.html

[24] Semtech. (Oct. 2016). *LoRaWAN–Simple Rate Adaptation Recommended Algorithm*. Accessed: Nov. 18, 2022. [Online]. Available: https://www.thethingsnetwork.org/forum/uploads/default/original/2X/7/7480e044aa93a54a910dab8ef0adfb5f515d14a1.pdf

[25] A. Mahmood, E. G. Sisinni, L. Guntupalli, R. Rondón, S. A. Hassan, and M. Gidlund, "Scalability analysis of a LoRa network under imperfect orthogonality," *IEEE Trans. Ind. Informat.*, vol. 15, no. 3, pp. 1425–1436, Mar. 2019.

[26] N. Benkahla, H. Tounsi, Y.-Q. Song, and M. Frikha, "Enhanced ADR for LoRaWAN networks with mobility," in *Proc. 15th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2019, pp. 1–6.

[27] N. Benkahla, H. Tounsi, Y.-Q. Song, and M. Frikha, "VHMM-based E-ADR for LoRaWAN networks with unknown mobility patterns," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, Jun. 2021, pp. 86–91.

[28] R. Hamdi, E. Baccour, A. Erbad, M. Qaraqe, and M. Hamdi, "LoRa-RL: Deep reinforcement learning for resource management in hybrid energy LoRa wireless networks," *IEEE Internet Things J.*, vol. 9, no. 9, pp. 6458–6476, May 2022.

[29] 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.

[30] X. Huang, J. Jiang, S.-H. Yang, and Y. Ding, "A reinforcement learning based medium access control method for LoRa networks," in *Proc. IEEE Int. Conf. Netw., Sens. Control (ICNSC)*, Oct. 2020, pp. 1–6.

[31] S. A. Almarzoqi, A. Yahya, Z. Matar, and I. Gomaa, "Re-learning EXP3 multi-armed bandit algorithm for enhancing the massive IoT-LoRaWAN network performance," *Sensors*, vol. 22, no. 4, p. 1603, Feb. 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/4/1603

[32] D.-T. Ta, K. Khawam, S. Lahoud, C. Adjih, and S. Martin, "LoRa-MAB: A flexible simulator for decentralized learning resource allocation in IoT networks," in *Proc. 12th IFIP Wireless Mobile Netw. Conf. (WMNC)*, Sep. 2019, pp. 55–62.

[33] D. Zorbas, "Improving LoRaWAN downlink performance in the EU868 spectrum," *Comput. Commun.*, vol. 195, pp. 303–314, Nov. 2022.

[34] B. Reynders, W. Meert, and S. Pollin, "Range and coexistence analysis of long range unlicensed communication," in *Proc. 23rd Int. Conf. Telecommun. (ICT)*, May 2016, pp. 1–6.

[35] L. Beltramelli, A. Mahmood, P. Ferrari, P. Osterberg, M. Gidlund, and E. Sisinni, "Synchronous LoRa communication by exploiting large-area out-of-band synchronization," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 7912–7924, May 2021.

[36] C. Lehong, B. Isong, F. Lugayizi, and A. M. Abu-Mahfouz, "A survey of LoRaWAN adaptive data rate algorithms for possible optimization," in *Proc. 2nd Int. Multidisciplinary Inf. Technol. Eng. Conf. (IMITEC)*, Nov. 2020, pp. 1–9.

[37] L. Beltramelli, A. Mahmood, M. Gidlund, P. Osterberg, and U. Jennehag, "Interference modelling in a multi-cell LoRa system," in *Proc. 14th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2018, pp. 1–8.

[38] *SX1272/73–860 MHz to 1020 MHz Low Power Long Range Transceiver*, Semtech Corporation, Camarillo, CA, USA, Jan. 2019.

[39] R. Kerkouche, R. Alami, R. Feraud, N. Varsier, and P. Maille, "Node-based optimization of LoRa transmissions with multi-armed bandit algorithms," in *Proc. 25th Int. Conf. Telecommun. (ICT)*, Jun. 2018, pp. 521–526.

[40] C. Phillips, D. Sicker, and D. Grunwald, ''Bounding the error of path loss models,'' in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, May 2011, pp. 71–82.

[41] S. Bubeck and N. Cesa-Bianchi, ''Regret analysis of stochastic and nonstochastic multi-armed bandit problems,'' 2012, *arXiv:1204.5721*.

[42] M. M. Asad, I. Marouf, and Q. Abu Al-Haija, ''Review of fast multiplication algorithms for embedded systems design,'' *Int. J. Sci. Technol. Res.*, vol. 6, no. 8, pp. 238–242, Aug. 2017.

[43] T. Voigt, M. Bor, U. Roedig, and J. Alonso, ''Mitigating inter-network interference in LoRa networks,'' 2016, *arXiv:1611.00688*.

[44] M. Anjum, M. A. Khan, S. A. Hassan, A. Mahmood, H. K. Qureshi, and M. Gidlund, ''RSSI fingerprinting-based localization using machine learning in LoRa networks,'' *IEEE Internet Things Mag.*, vol. 3, no. 4, pp. 53–59, Dec. 2020.

SYED USAMA MINHAJ received the B.E. degree in electrical engineering from the National University of Sciences and Technology (NUST), Islamabad, Pakistan, in 2021. He is currently pursuing the M.Sc. degree in communications engineering with the Technical University of Munich, Bavaria, Germany. He is currently working with Infineon Technologies, Munich. His research interests include low-power wide area networks, radio resource allocation, and applied ML/AI for wireless networks.

AAMIR MAHMOOD (Senior Member, IEEE) received the B.E. degree in electrical engineering from National University of Sciences and Technology (NUST), Pakistan, in 2002, and the M.Sc. and D.Sc. degrees in communications engineering from the School of Electrical Engineering, Aalto University, Finland, in 2008 and 2014, respectively. He worked as a Research Intern at the Nokia Research Center, Finland, in 2014; a Visiting Researcher at Aalto University; and a Postdoctoral Researcher at Mid Sweden University, Sweden, from 2015 to 2018, where he has been an Assistant Professor, since 2019. His research interests include RF interference/coexistence, LPWANs, intelligent radio resource, and network management for critical IoT networks.

SARDER FAKHRUL ABEDIN (Member, IEEE) received the B.S. degree in computer science from Kristianstad University, Kristianstad, Sweden, in 2013, and the Ph.D. degree in computer engineering from Kyung Hee University (KHU), South Korea, in 2020. He served as a Postdoctoral Researcher at the Department of Computer Science and Engineering, KHU, and also at Mid Sweden University, Sweden, where he is currently an Assistant Professor with the Department of Computer and Electrical Engineering. His research interests include the Internet of Things (IoT) network management, edge computing, machine learning, network optimization, and wireless networking.

SYED ALI HASSAN (Senior Member, IEEE) received the B.E. degree in electrical engineering from the National University of Sciences and Technology (NUST), Islamabad, Pakistan, in 2004, the M.S. degree in electrical engineering from the University of Stuttgart, Germany, in 2007, the M.S. degree in mathematics from Georgia Tech, in 2011, and the Ph.D. degree in electrical engineering from the Georgia Institute of Technology, Atlanta, USA, in 2011.

He was a Visiting Professor at Georgia Tech, in Fall 2017. He also held industry positions with Cisco Systems Inc., CA, USA, and with the Center for Advanced Research in Engineering, Islamabad. He is currently working as a Professor with the School of Electrical Engineering and Computer Science (SEECS), NUST. His research interests include signal processing for communications with a focus on cooperative communications for wireless networks, stochastic modeling, estimation and detection theory, and smart grid communications.

MUHAMMAD TALHA BHATTI received the B.E. degree in electrical engineering from the National University of Sciences and Technology (NUST), in 2021. He is currently pursuing the master's degree in information and communication technology and specialize in digital communications, wireless networks, embedded systems design, and ML/AI with Friedrich-Alexander-Univeristy Erlangen-Nürnberg, Bavaria, Germany. He worked as a Controls and Communication Engineer at the Fatima Group, one of the largest business conglomerates in Pakistan, gaining automation experience in the fertilizers industry.

SYED HAIDER ALI received the B.E. degree in electrical engineering from the National University of Sciences and Technology (NUST), Islamabad, Pakistan, in 2021. He is currently pursuing the M.Sc. degree in smart cities and communities with Heriot-Watt University, Riccarton, Scotland, U.K., under the Erasmus Mundus Scholarship. His research interests include power generation, power distribution, power electronic, smart grids, and machine learning.

MIKAEL GIDLUND (Senior Member, IEEE) received the Licentiate of Engineering degree in radio communication systems from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2004, and the Ph.D. degree in electrical engineering from Mid Sweden University, Sundsvall, Sweden, in 2005.

From 2006 to 2007, he was a Research Engineer and a Project Manager at Acreo AB, Hudiksvall, Sweden. From 2007 to 2008, he was the Project Manager and a Senior Specialist at Nera Networks AS, Bergen, Norway. From 2008 to 2015, he was a Senior Principal Scientist and a Global Research Area Coordinator of wireless technologies at ABB Corporate Research, Västerås, Sweden. Since 2015, he has been a Professor of computer engineering with Mid Sweden University. He holds more than 20 patents (granted and pending applications) in the area of wireless communication. His current research interests include wireless communication and networks, wireless sensor networks, access protocols, and security. He is an Associate Editor of the IEEE Transactions on Industrial Informatics.