



CADD: Collaborative adaptive duty cycle and data reduction for congestion control in LoRaWAN

Laura Acosta-Garcia ^{a,b}, Juan Aznar-Poveda ^{c,*}, Stefan Pedratscher ^c,
Marlon Etheredge ^c, Rafael Asorey-Cacheda ^{a,b}, Joan Garcia-Haro ^{a,b},
Thomas Fahringer ^c, Antonio-Javier Garcia-Sanchez ^{a,b}

^a Department of Information and Communication Technologies, Universidad Politécnica de Cartagena, Spain

^b ERG People Oriented Smart Technologies (POSTLab), European University of Technology, European Union, Spain

^c Distributed and Parallel Systems group, University of Innsbruck, Austria

ARTICLE INFO

Keywords:

Internet of things
Long range
Wireless communications
Transmission parameters
Energy efficiency

ABSTRACT

The widespread adoption of Internet of Things (IoT) technologies in domains such as smart agriculture, environmental monitoring, and industrial automation has significantly increased the number of connected devices and the volume of data generated. LoRaWAN, a protocol originally designed for low-bandwidth and sporadic transmissions, faces challenges in supporting these evolving requirements. In particular, the limited channel capacity and uncoordinated ALOHA-based medium access protocol render LoRaWAN increasingly susceptible to congestion, packet collisions, and energy inefficiencies in dense deployments. This paper presents CADD, a novel channel congestion control mechanism for LoRaWAN networks that addresses end-device and gateway-level constraints while incorporating data priority. The proposed approach combines (i) a dynamic data reduction technique at the end devices to eliminate redundant transmissions and (ii) an adaptive duty cycle regulation mechanism that adjusts transmission opportunities based on channel congestion observed at the gateway. This joint strategy enables efficient use of the communication channel while ensuring the reliable delivery of critical data. The proposed method is evaluated using realistic datasets in large-scale network scenarios. Experimental results demonstrate that CADD effectively reduces transmission frequency and energy consumption, mitigates channel congestion, and improves overall network performance and scalability in high-density IoT deployments. In scenarios with up to 500 devices, CADD achieves up to a $4 \times$ higher PDR and reduces energy consumption by approximately 54% compared to a state-of-the-art congestion control scheme and a static duty cycle baseline.

1. Introduction

LoRa communications, based on the LoRaWAN protocol, have emerged as a key technology for Internet of Things deployments due to their low power consumption and long-range coverage [1,2]. LoRaWAN was conceived for low-bandwidth and infrequent transmissions, primarily for applications involving small data payloads and long-range communications [3–5]. Therefore, it is well-suited for remote sensing or status updates, where energy efficiency and extended battery life are prioritized over high data throughput.

* Corresponding author.

E-mail address: juan.aznar-poveda@uibk.ac.at (J. Aznar-Poveda).

<https://doi.org/10.1016/j.iot.2026.101904>

Received 26 August 2025; Received in revised form 27 January 2026; Accepted 16 February 2026

Available online 21 February 2026

2542-6605/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

However, the evolution of the IoT across sectors like smart agriculture, environmental monitoring, smart cities, and industrial automation has led to a dramatic increase in the number of connected devices and volume of data generated [6–13]. This increasing demand has driven a transition from traditional transmission intervals of hours or days to significantly shorter intervals, often spanning just a few minutes [14–17]. As a result, LoRaWAN networks are prone to congestion due to their limited channel capacity and resource constraints [18]. Congestion can lead to increased energy consumption, packet losses, and collisions, ultimately degrading the network's overall performance. In this context, distinguishing between critical and non-critical data becomes increasingly important, as assigning the same transmission priority to all packets can lead to inefficient energy management and degrade Quality of Service (QoS) [19–21]. This change in usage patterns, combined with the deployment of densely populated networks and higher data rates, further exacerbates these challenges, highlighting the need for adaptive strategies that can manage communication resources efficiently.

LoRaWAN typically employs a star topology [22] in which a set of gateways manages all the traffic from the end devices and forwards it to a centralized network server connected to the Internet. The network server ensures reliable data exchange among sensors, actuators, gateways, and applications. While this topology simplifies deployment and management, it relies on a limited number of gateways, exacerbating scalability challenges [23–25]. To alleviate this issue, LoRaWAN employs an ALOHA-based Medium Access Control (MAC) protocol, which allows end devices to transmit data whenever they have information to send, without requiring prior coordination. ALOHA is a multiple-access protocol that operates in the MAC layer, enabling data transmission from multiple nodes over a shared channel. This reduces overhead, supports asynchronous communication, and is well-suited for low-power devices operating in unlicensed frequency bands such as the 868 MHz (EU) or 915 MHz (US). LoRaWAN also operates with multiple frequency channels or sub-bands, allowing devices to distribute their transmissions across these channels. This strategy helps reduce inter-channel interference and, therefore, congestion by balancing the load across available frequencies. However, the lack of coordination inherent to the ALOHA protocol also leads to an increased risk of co-channel interference and collisions when multiple devices transmit simultaneously on the same frequency. In dense deployments, this can lead to severe channel congestion, reducing the Packet Delivery Ratio (PDR) and causing higher latency [26–28].

In the literature, some works have also optimized transmission parameters to reduce collisions and co-channel interference, and consequently, alleviate channel congestion [29–33]. Collisions and interference are highly influenced by the choice of spreading factor, as this parameter affects both the time-on-air and communication range [28]. Higher spreading factors lead to longer transmissions and broader coverage, which increases the likelihood of packet overlap and co-channel interference, particularly in dense deployments. However, these approaches often overlook channel congestion and lack adaptability to dynamic network conditions [33,34]. Others tackled interference and collisions through detection, synchronization, or channel access schemes [27,35–37], yet these typically relied on precise synchronization or energy-intensive mechanisms. Data compression and reduction techniques were also proposed, as well as prediction techniques to reduce transmission volume and energy use [38–41], but they frequently assumed regular sensing patterns and overlooked gateway overload. Moreover, existing approaches rarely evaluate the impact of data reduction on the fidelity of the transmitted data, i.e., how well the reconstructed signal preserves the essential characteristics of the original measurements. Adaptive duty cycle methods, where the duty cycle refers to the fraction of time a device is allowed to transmit, have also been explored to regulate transmissions based on network or energy conditions [42,43], although they often failed to account for data priority or channel congestion. Overall, existing approaches tend to address isolated aspects of the channel congestion problem, such as interference, collisions, or energy consumption, without providing a comprehensive and dynamic mechanism that jointly considers end device and gateway congestion, resource constraints, and data priority [44].

To address these issues, we propose CADD, a novel channel congestion control approach that increases the scalability, energy efficiency, and communication reliability of LoRaWAN networks while considering different levels of data priority. The method leverages both signal quality and interference indicators to accurately assess channel congestion, enabling dynamic adaptation to changing channel conditions. CADD combines continuous data reduction with periodic adaptations of the duty cycle of each end device using the congestion information measured at the gateway side.

First, a dynamic data reduction strategy is applied at the end devices to limit redundant transmissions, which is crucial to alleviate interference and minimize end device energy consumption. In critical applications where almost every data point may represent valuable information [19–21], the gateway or network server can reconstruct the time series using anything from simple linear interpolation to more advanced machine learning-based reconstruction to improve accuracy and capture nonlinear trends [45,46]. In any case, it is important to highlight that excessive data reduction may lead to significant information loss, potentially affecting the accuracy of the reconstructed data. To address this trade-off, our approach minimizes transmissions while preserving data fidelity below an acceptable threshold. In applications such as environmental monitoring, where small deviations are tolerable, this strategy can achieve substantial energy savings without compromising the overall trends and patterns of the sensed data.

While data reduction effectively minimizes redundant transmissions and conserves energy, interference may still degrade performance in dense networks with many devices that may transmit simultaneously [18]. For this reason, we complement data reduction with a dynamic duty cycle adaptation mechanism. This joint adaptation is essential because congestion depends on the local transmission behavior of end devices, but the overall channel status is measured at the gateway side. By coordinating decisions across both the end device and gateway, CADD effectively responds to dynamic conditions, prevents overload, and prioritizes the delivery of critical information capabilities that isolated, device-only or gateway-only strategies struggle to achieve. When many devices operate simultaneously, and all try to use the maximum allowed duty cycle, the channel can quickly saturate. In such cases, devices may still sense data but fail to transmit due to airtime limits, causing a loss of critical information in dense deployments. To address this, we dynamically adjust each device's duty cycle based on congestion and data priority. Instead of enforcing a fixed duty limit, we regulate transmissions to relieve channel load while preserving critical data delivery. A lightweight heuristic algorithm enables fast, scalable

decisions without complex optimization or sequential learning. This is complemented by data reduction at the end device, which suppresses redundant values. Under high congestion, the reduction becomes more aggressive to prioritize essential information; when the channel is less loaded, more data is transmitted. Unlike offline time-series analysis of historical data, our method adapts the duty cycle and data reduction dynamically at the end devices and gateways by leveraging the most recent channel conditions, interference levels, and data similarity [45,46]. This joint adaptation enables efficient adaptation while preserving the lightweight computation required by constrained devices, and it eliminates the need to offload historical data processing to external devices or infrastructure.

The contributions of this paper can be summarized as follows:

- A novel channel congestion control approach for LoRaWAN networks, designed to improve scalability, energy efficiency, and communication reliability in high-density deployments.
- The integration of a lightweight redundancy suppression mechanism that dynamically reduces unnecessary data transmissions, making it particularly suitable for scenarios with a large number of connected devices and high data volumes.
- An adaptive duty cycle mechanism to dynamically regulate transmission opportunities based on network conditions and data priority. This approach ensures efficient channel usage while maintaining the reliable delivery of critical information under strict regulatory constraints.
- The proposed mechanism achieves up to a $4 \times$ higher PDR while reducing energy consumption by around 54%, thereby enhancing overall performance and scalability in high-density IoT deployments.

The rest of this paper is organized as follows. [Section 2](#) describes and discusses related work. [Section 3](#) introduces the proposed model, including the network architecture, transmission reliability, path loss, and interference models. [Section 4](#) presents the proposed method, detailing the data reduction strategy and adaptive duty cycle control mechanism. [Sections 5](#) and [6](#) describe the experimental setup and the evaluation results, respectively, comparing it with state-of-the-art congestion mechanisms in LoRaWAN. Finally, [Section 7](#) summarizes the major findings of this study and outlines future work.

2. Related work

Channel congestion in LoRaWAN networks, mainly caused by packet collisions and co-channel interference, has become a critical challenge, especially as the number of connected devices and the volume of generated data increase. This challenge is exacerbated by the limited resources of LoRaWAN, the lack of coordination inherent in ALOHA-based protocols, and the reliance on a small number of gateways for managing network traffic.

2.1. Optimization of transmission parameters

Several approaches have been explored to mitigate channel congestion due to interference and collisions in LoRaWAN networks. One of the strategies employed was the optimization of transmission parameters [29–32,34,47], such as the dynamic selection of the data rate (DR) and spreading factor (SF). For example, works such as [29] developed a congestion-aware ADR algorithm that adapted the spreading factor based on network conditions, showing improved data delivery rates (DER) and energy consumption (EC) in dense networks. Similarly, [30] proposed a contention-aware ADR method that limited the number of devices allowed to use the same data rate, minimizing the likelihood of collisions, which degraded performance. The authors in [31] proposed a probabilistic model and optimization framework that dynamically selected the most suitable SF for retransmissions, aiming to maximize the probability of success while avoiding congestion on higher SFs. Their approach adaptively reallocated SFs based on each device's transmission history, effectively reducing collisions in dense scenarios. Likewise, the I-ASF algorithm proposed in [32] enhanced SF assignment by considering gateway sensitivity, SF-specific interference, time overlap of colliding packets, and interfering energy. This interference-aware allocation minimized the Time on Air (ToA), leading to a significant improvement in the packet success ratio in dense deployments. However, these methods often lacked adaptability to dynamic changes in network conditions since they focused on end device adjustments while neglecting channel congestion.

2.2. Co-channel interference and collisions

While the optimization of transmission parameters contributed to reducing collisions by improving channel use, other research efforts have focused more directly on the interference and packet collisions that arose in dense deployments [27,35–37]. These works explored diverse strategies such as interference detection, synchronization mechanisms, and channel access enhancements to improve communication reliability. The review in [27] presented a comprehensive classification of interference sources in LoRa networks, covering inter-network and intra-network domains. While their classification focused on network-level interference domains, the paper also discussed key mechanisms, such as co-SF and inter-SF interference, that contributed directly to channel congestion in dense deployments. Their review included various mitigation and avoidance techniques, including dynamic SF allocation, enhanced Adaptive Data Rate (ADR) schemes, and interference cancellation at the receiver end. In [35], the authors proposed a two-fold interference mitigation approach for LoRaWAN. Instead of avoiding collisions, they introduced an interference-aware algorithm that enabled the detection of superposed LoRa signals. Additionally, a combining mechanism at the network server leveraged redundant packet receptions from multiple gateways to improve decoding reliability. These techniques reduced retransmissions and energy consumption and allowed end devices to use lower spreading factors, improving spectral efficiency in large-scale deployments. The work in [36]

proposed a time synchronization platform for LoRaWAN that enabled slot-based communication using a master-slave architecture, aiming to reduce collisions, improve downlink availability, and enhance energy efficiency. Finally, [37] proposed CANL, a novel asynchronous collision avoidance mechanism for LoRaWAN, which implemented a Listen-Before-Talk (LBT) strategy by placing end devices in packet reception mode before transmission. Simulations showed that CANL improved PDR compared to traditional ALOHA while maintaining low energy consumption. However, these approaches often rely on precise time synchronization, multi-gateway combining, or extended listening, which may be impractical and energy-intensive in large-scale LoRaWAN deployments.

2.3. Data compression and reduction

Many works used data compression and reduction techniques to minimize energy consumption in end devices [6,38–40,48–53]. By improving the energy efficiency of end devices, these methods decreased the frequency and volume of transmissions, thereby reducing the network load [38]. evaluated lightweight temporal compression algorithms that enhanced battery life by significantly reducing transmission frequency, emphasizing the role of sleep current consumption in device longevity. Similarly, [48] demonstrated the efficiency of temporal compression methods to extend sensor battery life by adjusting key parameters such as the SF and measurement interval. Various works used prediction algorithms for data reduction, further optimizing transmission by filtering data using methods such as change detection, thereby conserving energy and alleviating congestion. For example, [39] introduced an adaptive method for data reduction that used prediction to reduce communication costs while maintaining high accuracy, leveraging sensor data similarity to decide whether to transmit. In the same vein, [49] proposed a double-layered in-network data reduction method, and [6] presented a dynamic subsampling technique aimed at tackling the increasing data traffic in IoT systems. In environments like agriculture, where conditions do not often change, [40] proposed an analytical prediction algorithm using estimations (APAEs) across sensing, fog, and cloud layers. The APAEs algorithm predicts the next data value and only transmits when the actual data differs significantly from the prediction, effectively reducing unnecessary transmissions and conserving energy and bandwidth. While predictive methods achieved considerable reductions in data transmission, they also introduced computational challenges. For instance, [50] and [51] employed models based on Long Short Term Memory (LSTM) and dual prediction schemes to predict sensor data and reduce unnecessary transmissions. However, the complexity of these models, particularly the use of LSTM, often resulted in increased computational energy consumption at the sensor nodes. To balance prediction accuracy with computational efficiency, studies such as [38,52,53] explored lighter-weight methods. [52] combined data correlation and prediction techniques to avoid redundant transmissions in smart agriculture scenarios, while [53] and [38] investigated the effectiveness of temporal compression and lightweight protocols to extend sensor node lifetime. While these techniques effectively reduced energy use and transmission volume at the device level, they often assumed regular sensing patterns, limiting their applicability in dynamic or event-driven scenarios. Despite efforts to use lightweight methods, some models still imposed a computational burden on constrained nodes, focusing solely on end device efficiency and overlooking gateway congestion, which remains a major scalability bottleneck in dense LoRaWAN deployments. Furthermore, the impact of such reduction techniques on data fidelity (how accurately the reconstructed data reflects the original measurements) was rarely assessed, which could lead to the loss of critical information.

2.4. Adaptive duty cycle control

While data reduction techniques aimed to limit the volume and frequency of transmissions, other approaches focused on regulating when transmissions occur through an adaptive duty cycle [34,42,43,54–56]. For example, [55] proposed a duty cycle data aggregation algorithm to improve energy efficiency based on sensor activity in precision agriculture applications. In [42], the authors addressed the challenges of dynamic data generation in IoT networks by proposing an Adaptive Duty Cycle Control (ADCC) scheme. This approach combined dynamic duty cycle adjustment with efficient energy management to mitigate congestion, reduce latency, and minimize data loss. Similarly, [54] introduced a dynamic duty cycle mechanism that adapted to changes in traffic load and battery status, synchronizing with neighboring devices to improve packet delivery rates and reduce end-to-end delay. Other studies leveraged machine learning (ML) and advanced algorithms to optimize energy efficiency and system responsiveness. For instance, [43] incorporated a BiLSTM-based algorithm that identified traffic load patterns and optimized sleep scheduling based on energy levels and traffic predictions. By adjusting duty cycles according to predicted demand, the system reduces the computational load while maintaining sensor energy. Similarly, [56] used a reinforcement learning (RL) model to balance energy sustainability and Quality of Service (QoS) through adaptive duty cycling in energy-harvesting environments. Although these works effectively extended battery life and improved energy use, channel interference and packet collisions remain significant challenges in high-density networks. These methods also did not consider the criticality of the data, which may lead to the loss of high-priority information.

3. System model

In this section, we present the system model used to support the design and evaluation of CADD. The model characterizes the key LoRaWAN network components, reliability constraints, and interference dynamics. It serves as the analytical basis for capturing network behavior under varying channel congestion levels and guides the development of adaptive control mechanisms.

3.1. Network model

We consider a LoRaWAN network composed of a set of end devices $E = \{e_1, \dots, e_n\}$ and a set of gateways $G = \{g_1, \dots, g_m\}$, where n and m represent the total number of end devices and gateways, respectively. The gateways serve as intermediaries that receive

uplink transmissions from EDs and forward them to the network server. We assume that each ED is equipped with a sensor that collects data from the environment, such as temperature, humidity, or air quality. In this context, the EDs in our model are assumed to be Class A devices [22], which are widely used in LoRaWAN for their simplicity and energy efficiency, as they only transmit upon data generation and otherwise remain in sleep mode. However, not all data collected by the sensor is transmitted. Some data may be discarded because the EDs must comply with duty cycle regulations, each one operating under a transmission constraint, which limits the proportion of time a device can be active for transmission. Specifically, the duty cycle of each device is constrained by the lower and upper bounds, $d_i^{\min}$ and $d_i^{\max}$, which define the minimum and maximum transmission times allowed for each device. These restrictions are reset at fixed time intervals T . As a result, only a subset of the sensed data is transmitted, denoted as s'_i . Finally, each ED is associated with a priority level λ_i , which is used to prioritize the transmission among packets.

3.2. Transmission reliability model

In a LoRaWAN network, the reliability of transmissions is determined by two main factors: (i) the Received Signal Strength Indicator (RSSI) at the gateway must be above its receiver sensitivity threshold, and (ii) the packet should not be corrupted by interference from the transmissions of other end devices [57]. PDR indicates the likelihood that a packet transmitted by $e_i \in E$ is successfully received by $g_k \in G$. It can be computed as the product of two probabilities: the first is the probability that the RSSI is sufficient for successful reception (ψ_{ik}), and the second is the probability that the packet is not corrupted by interference (ζ_{ik}) [28]:

$$PDR_{ik} = \psi_{ik} \cdot \zeta_{ik} \quad (1)$$

To calculate the RSSI of a transmission, we adopt the log-distance path loss model, as it applies to both rural and urban environments. This model considers the transmission power, the distance between the end device and the gateway, and the signal attenuation effects caused by fading.

$$RSSI_{ik} = p_i - PL(d_0) - 10\gamma \log_{10} \left(\frac{d_{ik}}{d_0} \right) - N_\sigma \quad (2)$$

where $p_i \in \mathcal{P}$ is the transmission power of end device e_i , $PL(d_0)$ represents the mean path loss at a reference distance d_0 (typically under line-of-sight conditions), γ is the path loss exponent that indicates how the signal strength decays with distance in the given environment, d_{ik} is the Euclidean distance between end device e_i and gateway g_k , and N_σ is a Gaussian noise term with zero mean and standard deviation σ , which models the random variations in signal strength due to shadow fading.

Then, to determine the probability that the RSSI is sufficient for successful reception, denoted as ψ_{ik} , we decompose the RSSI into a deterministic component z_{ik} and a random noise component N_{ik} , where $RSSI_{ik} = z_{ik} - N_{ik}$, with $z_{ik} = p_i - PL(d_0) - 10\gamma \log_{10} \left(\frac{d_{ik}}{d_0} \right)$ and $N_{ik} \sim \mathcal{N}(0, \sigma)$, a Gaussian random variable representing the noise. The probability that the RSSI exceeds the receiver sensitivity threshold η_{f_i} at a given spreading factor f_i is computed using the error function $\text{erf}(\cdot)$ as follows [58]:

$$\psi_{ik} = P(RSSI_{ik} \geq \eta_{f_i}) = \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{z_{ik} - \eta_{f_i}}{\sqrt{2}\sigma} \right) \quad (3)$$

where η_{f_i} is the GW's sensitivity threshold.

3.3. Interference model

Co-channel interference has a significant impact on packet delivery in LoRaWAN networks, particularly when multiple end devices transmit on the same frequency channel. This type of interference occurs when the signal from one device is strong enough to degrade the communication quality of another device, resulting in packet loss. To model this interference, we calculate the probability that a packet from end device e_i at gateway g_k is successfully received without being disrupted by transmissions from other EDs, denoted as ζ_{ik} . This probability is determined by considering the potential overlap between the transmissions of different EDs. For two devices, e_i and e_j , the probability that end device e_j interferes with end device e_i depends on the time overlap of their transmissions. Assuming that a transmission can be successfully decoded if at least the last five preamble symbols are received without interference [59], we compute the time interval during which end device e_j must refrain from transmitting to avoid interfering with end device e_i 's transmission as $T'_{ij} = T_j + T_i - (n_{pr} - 5)T_i^{\text{sym}}$, where T_j and T_i are the transmission durations for e_j and e_i , respectively, while n_{pr} is the number of preamble symbols, and T_i^{sym} represents the symbol duration for end device e_i . Using this time window, we calculate the probability that e_j interferes with e_i 's transmission based on the exponential nature of packet generation in the network [28,60] as $h_{ij} = 1 - e^{-\chi T'_{ij}}$, where χ is the mean packet generation rate of the devices in the network. We also need to consider the robustness of LoRa modulation. LoRa's modulation allows packets to be successfully decoded even when the RSSI is below the noise floor, provided that the signal-to-interference ratio (SIR) remains above a certain threshold, which depends on the spreading factor used by end device e_i and end device e_j [61–63]. The probability that end device e_i 's transmission is corrupted by interference from e_j is then calculated as:

$$\mathbb{P}(RSSI_{ik} - RSSI_{jk} < \omega_{f_i f_j}) = \mathbb{P}(N_{jk} - N_{ik} < \omega_{f_i f_j} - (z_{ik} - z_{jk})) \quad (4)$$

$$= \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{\omega_{f_i f_j} - (z_{ik} - z_{jk})}{2\sqrt{2}\sigma} \right) \quad (5)$$

where $\omega_{f_i f_j}$ is the interference threshold, erf denotes the Gaussian error function, and N_{jk} and N_{ik} represent the Gaussian random variables that model the noise for end devices e_j and e_i , respectively. Finally, to compute the total interference probability across the network, we consider the cumulative interference effect from all the other EDs. The probability that a packet from end device e_i at gateway g_k is not corrupted by interference is expressed as:

$$\zeta_{ik} = \prod_{j \in \{1..n\}, i \neq j} (1 - h_{ij} \mathbb{P}(RSSI_{ik} - RSSI_{jk} < \omega_{f_i f_j})) \quad (6)$$

This formulation inherently models the capture effect, allowing packets to be decoded even under temporal overlap when the SIR exceeds the corresponding threshold [58]. Rather than assuming full knowledge of collisions or instantaneous channel states, this approach employs metrics realistically available in LoRaWAN deployments, such as measured RSSI values, spreading factor assignments, and observed traffic patterns. In other words, it does not rely on perfect channel knowledge, including hidden collisions, but operates directly on stochastic and noisy channel metrics. This modeling approach aligns with common practice in the LoRaWAN literature, where interference is treated stochastically based on reception-related statistics and temporal overlap assumptions rather than instantaneous channel knowledge [28,57,64].

4. Method

In LoRaWAN networks, the uncoordinated nature of ALOHA-based access, combined with limited channel capacity and increasing device density, often leads to high channel interference and packet collisions. These issues are further aggravated by the continuous generation of sensor data [14–17], which may include redundant or low-priority information that unnecessarily occupies valuable transmission opportunities. To address these challenges, we propose CADD, a joint channel congestion control approach that combines adaptive data reduction and duty cycle control to improve scalability, energy efficiency, and communication reliability. The design emphasizes low computational overhead and responsiveness to dynamic network conditions, ensuring feasibility for real-world deployments in dense and resource-constrained LoRaWAN environments. In addition, CADD operates independently of the end-device position, relying solely on packet-level metrics continuously updated at the gateway. While our evaluation focuses on static deployments, CADD could potentially be extended to support mobile end devices in future work.

4.1. End device data reduction

In many IoT applications, such as agriculture, environmental monitoring, and smart cities, millions of data readings are generated by sensing nodes. These data generally show slight changes between consecutive readings. For example, parameters like temperature, humidity, and air quality tend to follow predictable, linear patterns. One of the key components of our model is a lightweight device-level data suppression mechanism that avoids transmitting sensor readings that remain similar over time, preserving only significant changes. This helps avoid unnecessary transmissions, reducing the burden on the channel and conserving energy [6,40,49]. Several techniques have been proposed for reducing sensor data, including delta encoding, piecewise approximation methods (e.g., PLA, PAA), and predictive models using time series forecasting. However, these methods may involve higher computational overhead and rely on assumptions about signal behavior that may not be valid in resource-constrained or rapidly changing environments. Our approach uses Symbolic Aggregate Approximation (SAX) [65], a technique that reduces the time series of continuous sensor readings to a string of symbols capturing only the significant changes, offering a good balance between reduction efficiency and computational simplicity. This technique allows the system to transmit only the relevant data, thereby minimizing unnecessary transmissions, reducing energy consumption, and preserving the essential trends in sensor readings. It is important to note that the reduction mechanism operates on each measurement as it arrives, without long-term buffering or temporal aggregation. Therefore, no additional delay is introduced beyond the inherent latency of LoRaWAN communications, and the data reduction process does not affect the timeliness of sensor updates. SAX has been widely used in IoT applications [10–13] that exhibits gradual variations. These properties are typical of environmental and infrastructure monitoring applications in LoRaWAN, such as temperature, humidity, air quality, illumination, and voltage measurements. Such signals often produce large volumes of redundant samples, where consecutive readings differ only slightly [40,66].

An important aspect of our approach is the concept of data fidelity, which refers to how well the reduced and reconstructed signal preserves the characteristics of the original time series [45]. Since SAX generates a symbolic representation of the data, the application can reconstruct an approximate version of the original signal using the transmitted symbols. In our implementation, this reconstruction is performed using linear interpolation between consecutive transmitted values [67].

Given a time series of sensor data $X = [x_1, x_2, \dots, x_l]$ of length l , observed at end device e_i , the first step in SAX is the segmentation of the time series into w consecutive, non-overlapping segments of length $q = \frac{l}{w}$, where the h th segment $s_h \in S$ contains the values from indices $(h-1) \cdot q + 1$ to $h \cdot q$.

$$s_h = [x_{(h-1) \cdot q + 1}, \dots, x_{h \cdot q}], \quad \forall h \in \{1, 2, \dots, w\} \quad (7)$$

For each segment, the mean value $\hat{s}_h$ is calculated as the average of all the values contained in s_h , capturing the representative magnitude of the data within that segment:

$$\hat{s}_h = \frac{1}{q} \sum_{x_h \in s_h} x_h, \quad \forall h \in \{1, 2, \dots, w\} \quad (8)$$

To ensure comparability across different time series, the resulting segment means are normalized using the mean and standard deviation of the original time series X . The normalized segment values are given by:

$$\hat{s}'_h = \frac{\hat{s}_h - \mu_X}{\sigma_X}, \quad \forall h \in \{1, 2, \dots, w\} \quad (9)$$

where μ_X is the mean and σ_X is the standard deviation of the original time series X :

$$\mu_X = \frac{1}{l} \sum_{i=1}^l x_i, \quad \sigma_X = \sqrt{\frac{1}{l} \sum_{i=1}^l (x_i - \mu_X)^2} \quad (10)$$

Each segment value $\hat{s}'_h$ is mapped to a symbol, resulting in a symbolized sequence $Z = [z_1, z_2, \dots, z_w]$, where each $z_h \in Z$ represents the symbolic approximation of the corresponding segment of the time series. We apply SAX to encode the time series and determine whether a given sensor value should be transmitted. To reduce redundant transmissions, a new reading is only sent if its symbolic representation differs from the previous one. This helps reduce network traffic and conserve energy by ensuring that only significant changes are communicated, rather than repetitive data.

Each end device dynamically adjusts the granularity of its symbolic representation according to its assigned duty cycle. When the channel is congested and the duty cycle is restricted, end devices have fewer opportunities to transmit. In this context, reducing the amount of data sent becomes crucial to avoid wasting limited transmission time on low-priority or redundant information. Specifically, the number of SAX regions used to encode the time series is determined by the current duty cycle. Lower duty cycles result in fewer regions (coarser granularity), while higher duty cycles allow more regions (finer granularity), enabling more detailed data to be preserved. This flexibility ensures that the reduction level adapts to the communication constraints of each device. [Algorithm 1](#) summarizes this process. The input time series X is first normalized (line 3), and the number of SAX regions R is computed based on the current duty cycle d_i (line 4). $R_{\min}$ and $R_{\max}$ denote the minimum and maximum number of regions, which define the range of symbolic granularity allowed. A higher duty cycle yields a larger R , enabling finer data representation, whereas a lower duty cycle enforces coarser discretization to reduce channel load. Then, each normalized mean value $\hat{s}'_h$ of each segment is assigned to a symbol z_h from alphabet α according to its corresponding region. It is important to note that this alphabet merely controls the granularity of its symbolic representation, and should not be confused with an alphabet used for decoding at the gateway. Regions are defined by a set of breakpoints β derived from the Gaussian distribution (lines 5–10), where Φ^{-1} represents the inverse of the quantile function of the standard normal distribution. Finally, only symbols that differ from the previously transmitted ones are sent, ensuring that only relevant changes are transmitted, while redundant data is suppressed (lines 12–17). The worst-case time complexity of the SAX transformation is $\mathcal{O}(n)$, where n is the length of the input time series. This results from a single pass through the data for segmentation, normalization, and symbol assignment.

Algorithm 1 CADD(ED).

```

1: Input:  $X, d_i, d_i^{\min}, d_i^{\max}, R_{\min}, R_{\max}, w$ 
2: Output:  $Z^{\text{tx}}$ 
3:  $\hat{S}' \leftarrow$  normalized segment means using Eqs. (7)–(9)
4:  $R \leftarrow R_{\min} + (R_{\max} - R_{\min}) \cdot \frac{d_i - d_i^{\min}}{d_i^{\max} - d_i^{\min}}$ 
5:  $\beta = \left\{ \Phi^{-1} \left( \frac{i}{R+1} \right) : i = 1, 2, \dots, R-1 \right\}$  ▷ Calculate breakpoints
6:  $Z \leftarrow []$ 
7: for  $\hat{s}'_h \in \hat{S}'$  do
8:    $z_h \leftarrow \alpha_j$  where  $j = \min\{k \in \{1, 2, \dots, |\beta|\} : \hat{s}'_h < \beta_k\} \cup \{|\beta|\}$  ▷ Symbol assignments
9:    $Z \leftarrow Z \cup z_h$ 
10: end for
11:  $Z^{\text{tx}} \leftarrow [], z_{\text{prev}} \leftarrow \text{None}$ 
12: for  $z_h \in Z$  do ▷ Data reduction
13:   if  $z_h \neq z_{\text{prev}}$  then
14:      $Z^{\text{tx}} \leftarrow Z^{\text{tx}} \cup z_h$ 
15:      $z_{\text{prev}} \leftarrow z_h$ 
16:   end if
17: end for
18: return  $Z^{\text{tx}}$ 

```

4.2. Adaptive duty cycle control

While data reduction at the sensor node level helps reduce potential congestion by analyzing data behavior, this approach may not fully account for all the factors that may lead to congestion in the channel, such as the number of devices transmitting simultaneously and the channel's capacity to handle traffic. For instance, a large number of devices transmitting at the same time can cause packet collisions or co-channel interference, leading to increased retransmissions and delays, especially in networks with limited channel

bandwidth and low transmission power [28]. Additionally, network capacity limitations, such as gateway bandwidth or the ability of the network to process incoming data, can quickly become bottlenecks when many devices attempt to send data simultaneously [2]. To address these factors, we introduce an adaptive duty cycle control at the gateway, which adjusts the transmission opportunities of each end device based on dynamic channel congestion levels. LoRaWAN usually operates under strict duty cycle regulations in unlicensed ISM bands (e.g., 868 MHz in Europe, 915 MHz in the US), which restrict the airtime of end devices to ensure fair spectrum usage and regulatory compliance. These limits, typically ranging from 0.1% to 10% depending on the frequency subband, define how long a device can transmit within a given period. For example, the 868-868.6 MHz subband in Europe enforces a 1% duty cycle. Building on this principle, our method dynamically adjusts the duty cycle of each device to reduce channel congestion, ensure the timely delivery of high-priority information, and maintain an acceptable PDR while adhering to regulatory duty cycle limits. We reduce the duty cycle of devices experiencing high collision rates, interference, or poor channel conditions, and increase it for those operating with few collisions and favorable transmission conditions. The system ensures efficient spectrum use. Critical data is prioritized through higher duty cycle assignments, ensuring timely delivery, while less critical data is transmitted less frequently. As a result, some non-critical information may be lost due to limited transmission opportunities, which is considered an acceptable trade-off to maintain overall network performance. This duty cycle adjustment works jointly with the data reduction mechanism. When fewer transmissions are allowed, data reduction becomes more aggressive to preserve bandwidth and energy. When more transmissions are possible, the reduction is relaxed to allow richer data delivery. This coordinated approach enhances network scalability and communication reliability.

To achieve this, our formulation assumes that each end device $e_i \in E$ generates traffic with a given priority level $\lambda_i \in [0, 1] \forall e_i \in E$, which reflects the criticality of the data it transmits. To ensure that the network remains functional under varying load conditions and avoids excessive packet drops due to overly restrictive duty cycles, we define the minimum effective delivery rate r_i^{eff} , representing the fraction of packets successfully transmitted relative to the total that would be generated under the maximum allowed duty cycle:

$$r_i^{\text{eff}} = \frac{p_i^{\text{tx}}}{p_i^{\text{max}}} \quad (11)$$

where p_i^{tx} denotes the number of packets successfully transmitted by device e_i under its current duty cycle $d_i \in [d_i^{\min}, d_i^{\max}]$, and p_i^{max} is the number of packets that would have been transmitted if the device operated at the maximum allowed duty cycle $d_i^{\max}$. In addition, we consider two key metrics: the probability $\zeta_i \in [0, 1]$ that a transmitted packet is not corrupted by interference, and the PDR $PDR_i \in [0, 1]$, defined in Eq. (1). The former, ζ_i , reflects the level of interference affecting the transmissions of device e_i , while the latter provides a direct measure of the overall reliability of the link, as it combines the effects of both signal quality and interference. Both serve as complementary indicators of channel congestion. A high value of ζ_i indicates that packets from device e_i are probably not being corrupted by interference, suggesting that the device is not currently experiencing congestion. Conversely, a low ζ_i suggests the presence of significant interference, which is typically associated with channel congestion due to overlapping transmissions from other devices. Similarly, a high PDR_i reflects successful packet delivery, whereas a low value may indicate channel congestion or poor transmission conditions, such as increased distance to the gateway or signal attenuation. When either ζ_i or PDR_i drops below acceptable levels, this implies that the device is experiencing performance degradation. In these scenarios, reducing the duty cycle can help limit unnecessary transmissions, alleviate channel load, and avoid contributing to further congestion. These three metrics— r_i^{eff} , ζ_i , and PDR_i —are periodically computed at the gateway for each end device $e_i \in E$, based on the packets received during the latest time window of duration T . This allows the gateway to assess the reliability and congestion level of the channel for each device, using up-to-date observations of transmission success and interference. Based on these metrics, the goal is to determine the appropriate duty cycle $d_i \in [d_i^{\min}, d_i^{\max}]$ for each end device to maximize network reliability, measured by the global PDR. At the same time, the solution must ensure that each device is able to transmit at least a minimum amount of data, even under unfavorable conditions, while allowing the system to adapt to interference and channel conditions without overloading the network.

4.2.1. Duty cycle adaptation algorithm

To compute immediate, discrete duty cycle adjustments per end device based on current communication performance, we propose a heuristic-based duty cycle adjustment strategy for LoRaWAN networks [57,68,69]. The proposed heuristic, detailed in Algorithm 2, is executed periodically at the gateway every period T and determines the updated duty cycle d_i for each end device $e_i \in E$ based on the PDR (PDR_i), the probability of non-interference (ζ_i), and the effective delivery rate (r_i^{eff}). We also consider priority λ_i , which reflects the criticality of the data transmitted by each end device. By jointly adapting these parameters, the strategy enables efficient in-network operation with minimal computational overhead, thereby removing the need to depend on external infrastructure for historical-data processing.

The algorithm starts by calculating network density ρ (line 3). For this purpose, we employ n_{max} , a preconfigured parameter representing the maximum expected number of devices within the gateway's coverage area, which serves as a reference to normalize the current number of active devices and compute the network density ρ . In line 4, if PDR_i and ζ_i are above certain thresholds (τ_{PDR} and τ_ζ , respectively), indicating reliable and interference-free communication, the channel is assumed to be favorable and the duty cycle is maximized. These two metrics are the main factors used to adapt the duty cycle, together with effective delivery rate, and the priority of the node. A low PDR_i with a high ζ_i (line 6) suggests strong fading and poor channel conditions rather than interference. Therefore, we minimize the duty cycle to reduce energy consumption. When both PDR_i and ζ_i are below their respective thresholds (line 8), the channel is considered congested. Suppose the end device has an effective delivery rate lower than θ_i (line 9). In this case, the algorithm increases the duty cycle proportionally to data priority λ_i (line 10) to improve transmission reliability. Otherwise (line 11), the duty cycle is determined through a weighted combination of priority and PDR_i and scaled according to the estimated network

Algorithm 2 CADD(GW).

```

1: Input:  $PDR_i, \zeta_i, r_i^{eff}, \lambda_i, \theta_i, d_i, d_i^{\min}, d_i^{\max}, n$ 
2: Output:  $d'_i$ 
3:  $\rho \leftarrow \min(\frac{n}{n_{\max}}, 1.0)$ 
4: if  $PDR_i \geq \tau_{PDR} \wedge \zeta_i \geq \tau_\zeta$  then ▷ Favorable channel conditions
5:    $d'_i \leftarrow d_i^{\max}$ 
6: else if  $PDR_i < \tau_{PDR} \wedge \zeta_i \geq \tau_\zeta$  then ▷ Degraded channel conditions
7:    $d'_i \leftarrow d_i^{\min}$ 
8: else
9:   if  $r_i^{eff} < \theta_i$  then ▷ Low effective rate
10:     $d'_i \leftarrow d_i + \pi_l \cdot \lambda_i$ 
11:   else
12:     $d'_i \leftarrow d_i^{\min} + (d_i^{\max} - d_i^{\min}) \cdot (\pi_a \cdot \lambda_i + (1 - \pi_a) \cdot PDR_i)$ 
13:   end if
14:    $d'_i \leftarrow d'_i \cdot (1 - \frac{\rho^{\pi_d}}{10})$ 
15: end if
16:  $d'_i \leftarrow \max(d_i^{\min}, \min(d'_i, d_i^{\max}))$ 
17: if  $|d_i - d'_i| \geq \pi_u$  then
18:    $d_i \leftarrow d'_i$  ▷ Update duty cycle via downlink message
19: end if
20: return  $d_i$ 

```

density to avoid overloading the channel (lines 12–14). The π_a parameter allows us to control this trade-off. Higher values bias the decision toward priority (favoring critical data), while lower values give more weight to the measured PDR , (favoring transmissions with better performance). Finally, the duty cycle is clipped to remain within regulatory bounds (line 16), and a downlink is triggered only if the change exceeds a predefined threshold π_u to avoid redundant downlink messages (lines 17–18). In our algorithm, end devices use unconfirmed uplink messages [59,70,71], and thus the gateway relies on periodic downlink control messages to update the duty cycle every T , and only when the change exceeds the predefined threshold. In multi-gateway deployments, each gateway executes an instance of the CADD algorithm, but only the gateway closest to a given end device, determined by the LoRaWAN network server, reports duty-cycle adaptations via downlink. This leverages the network server's existing functions for uplink deduplication and closest-gateway selection, ensuring consistent adaptations and seamless transfer of responsibility if the closest gateway changes. The downlink transmissions are assumed to be reliable, as our method focuses on uplink congestion and scalability. This assumption isolates the impact of duty cycle adaptation and data reduction without introducing additional variability. In practical LoRaWAN deployments, downlinks may occasionally be lost; however, CADD mitigates this by sending updates only when the duty cycle change exceeds a threshold, and by allowing end devices to operate with their last valid configuration until a new update is successfully received. In addition, the weighted combination of priority and PDR, together with the scaling by network density, also ensures that the algorithm incorporates fairness considerations. Although devices with higher PDR naturally benefit from more reliable links, nodes with poor link quality are not disadvantaged when their data is critical, as the algorithm increases their duty cycle proportionally to their priority. Likewise, even devices with both low PDR and low priority retain guaranteed transmission opportunities through the minimum duty cycle $d_i^{\min}$. The weighted combination of priority and PDR, together with the scaling by network density, prevents any subset of nodes from monopolizing the channel and ensures balanced access across heterogeneous link conditions, preserving fairness without compromising overall network efficiency. Finally, the proposed duty cycle algorithm exhibits a worst-case time complexity of $\mathcal{O}(1)$. Table 1 summarizes the notation, including configuration parameters and metrics, used by CADD.

5. Evaluation setup

This section describes the evaluation setup, detailing the baselines, infrastructure, datasets, performance metrics, and evaluation scenarios. During this evaluation, we focus on large-scale networks to assess the scalability and robustness of the proposed algorithm under various network conditions. Specifically, we evaluate the effectiveness of the proposed mechanism to alleviate channel congestion while improving scalability, energy efficiency, and communication reliability, especially in high-density IoT deployments.

5.1. Simulation framework

The evaluation experiments were conducted using a custom simulation framework developed in Python. We integrate FAST-LoRa [58], a lightweight and efficient LoRaWAN simulator, with SimPy [72], a discrete-event simulation library ideal for managing concurrent processes. FAST-LoRa is used to simulate wireless network behavior, including transmission, interference, and reception events, with low computational overhead. SimPy complements this by handling the temporal scheduling and event-driven execution of packet transmissions and receptions. In our simulations, we consider a single 125 kHz uplink channel, allowing us to focus on intra-channel interference and packet collisions. This configuration is standard in LoRaWAN analytical and simulation studies [73–76] and

Table 1
Summary of the main configuration parameters and metrics.

Parameter	Description
d_i	Current duty cycle of e_i
$d_i^{\min}, d_i^{\max}$	Minimum and maximum duty cycle constraints of e_i
T	Duty cycle reset interval
λ_i	Priority of e_i
s_h	h th segment of the time series sensor data
$\bar{s}_h$	Mean value of the h th segment
$\hat{s}'_h$	Normalized value of the h th segment
PDR	Packet Delivery Ratio
ζ_i	Probability that a packet is not interfered with at e_i
r_i^{eff}	Effective delivery rate of e_i
θ_i	Effective delivery threshold of e_i
p_i	Transmission power (dBm) of e_i
$RSSI_{ik}$	Received Signal Strength Indicator of e_i at g_k (dBm)
ψ_{ik}	Probability of sufficient RSSI of e_i at g_k
η_{f_i}	Receiver sensitivity at spreading factor f_i (dBm)
ω_{f_i, f_j}	SIR threshold between SFs f_i and f_j
n	Number of active end devices in the network
$n_{\max}$	Maximum supported number of end devices
R	Number of SAX regions (symbols)
w	Number of SAX segments
Z^{tx}	Transmitted symbolic sequence
$\pi_l, \pi_a, \pi_d, \pi_u$	Configurable parameters for duty cycle adaptation and frequency update

can be extended to multiple channels if required. Each ED is modeled as a process, where packets are generated at random intervals, since we assume Class A devices, while respecting their duty cycle. The GW processes the incoming packets, calculates network metrics, and adjusts the duty cycle accordingly. Although our evaluation focuses on static EDs, the simulation framework can be extended to support mobility by incorporating mobile ADR schemes that dynamically adjust transmission parameters as nodes move. The source code of CADD has been made publicly available¹

5.2. Baselines

In our evaluation, we validate CADD correctness and accuracy using three baseline algorithms: SFC-Aware-ADR, FDCC, and FDCC + SAX. The first baseline is the Spreading Factor Congestion-Aware ADR (SFC-Aware-ADR) algorithm [29], which is widely recognized in the literature as a representative ADR enhancement. It extends the standard ADR mechanism by incorporating congestion information into the data rate optimization process, thereby improving the collision probabilities of end device traffic in LoRaWAN networks. This baseline allows us to assess how the proposed mechanism performs against existing congestion-control solutions that do not alter the data generation process. The second baseline, FDCC (Fixed Duty Cycle Control), represents a simplified version of our proposal in which the duty cycle of all nodes is fixed and no data reduction is applied. While this scheme does not correspond to an advanced state-of-the-art mechanism, it is consistent with prior works in the field [42] that employ fixed duty cycle configurations as a baseline reference. This choice allows us to isolate and quantify the individual contributions of the adaptive duty cycle control and the redundancy suppression mechanism integrated in CADD. To disentangle the effects of data reduction from those of adaptive duty cycle control, we additionally consider FDCC + SAX. In this scheme, the duty cycle remains fixed, as in FDCC, while redundant sensor readings are suppressed using a static, SAX-based data reduction mechanism. This baseline enables a direct comparison between fixed duty cycle operation with and without data reduction, allowing us to quantify the benefits of redundancy suppression independently of duty cycle adaptation. In FDCC + SAX, the duty cycle is fixed to the maximum allowed value (1.0%), and the SAX segmentation parameter is set to the corresponding window size ($w = 72$), matching the configuration that CADD would select under the same duty cycle constraint. Together, these baselines support an incremental evaluation of the proposed mechanisms. They allow us to separately assess the impact of static data reduction, fixed duty cycle operation, and the additional gains achieved when local redundancy suppression at the end devices is jointly optimized with adaptive duty cycle control at the gateway. This progressive evaluation framework provides a clear understanding of how the integration of these mechanisms improves congestion mitigation, reliability, and energy efficiency in LoRaWAN networks.

5.3. Datasets

For our experiments, we use real-world datasets available at [77]. These datasets were collected using 54 Mica2Dot sensors equipped with weather boards, which recorded timestamped measurements of temperature, humidity, light, and voltage, along with sensor identifiers and topology metadata. Each sensor generated readings approximately every 30 s. These datasets are particularly

¹ GitHub repository link will be added here after the peer review process.

Table 2
Evaluation parameters used in the experiments.

Parameter	Value
Number of active EDs (n)	1–500
Frequency band	868 MHz (EU)
Bandwidth (BW)	125 kHz
Spreading Factors (SF)	7–12
Transmission power of end device e_i (p_i)	14 dBm
Payload size	26 bytes
Time window (T)	1 h
Number of executions	5 runs per scenario
Coding rate (CR)	4/5
Path loss at d_0 (PL_{d_0})	30 dB
Path loss exponent (γ)	3.5
Reference distance (d_0)	1 m
Shadowing std. dev. (σ)	2 dB
Distance ED-GW (d_{ik})	0–3.5 km
Noise floor	$-174 + 10 \log_{10}(\text{BW})$ dBm
Duty cycle range ($d_i^{\min}, d_i^{\max}$)	0.1 – 1.0%
Priority level (λ_i)	[0, 1] (uniform)
Effective delivery rate (r_i^{eff})	0–1
Effective delivery threshold (θ_i)	0.7
PDR threshold (τ_{pdr})	0.8
Non-interference threshold (τ_c)	0.8
Channel states	Optimal: $PDR_i \geq 0.8$ and $\zeta_i \geq 0.8$ Degraded: $PDR_i < 0.8$ and $\zeta_i \geq 0.8$ Congested: $PDR_i < 0.8$ and $\zeta_i < 0.8$
Delivery status	Insufficient: $r_i^{\text{eff}} < 0.7$ Satisfactory: $r_i^{\text{eff}} \geq 0.7$
Maximum supported number of EDs ($n_{\max}$)	500
Heuristic tuning parameters ($\pi_l, \pi_a, \pi_d, \pi_u$)	0.0 – 1.0
Minimum and maximum number of SAX regions ($R_{\min}, R_{\max}$)	4 – 12
Data reduction window (w)	4 – 72

suitable for our evaluation because they reflect typical environmental monitoring scenarios where data is generated at high frequency and exhibits gradual variations over time.

5.4. Metrics

To evaluate the performance of CADD, we use the following metrics:

- Packet Delivery Ratio (PDR): average PDR across all devices, indicating the reliability of packet reception.
- Number of received packets (RX): number of successfully decoded packets at the gateway.
- Energy Consumption (EC): total energy consumed by all devices, following established energy models [34,47].
- Duty Cycle (DC): mean duty cycle of nodes, indicating how transmission opportunities are assigned over time.
- Mean Absolute Error (MAE): average absolute difference between original and reconstructed time series, measuring information loss due to SAX-based reduction and data fidelity preservation.

5.5. Evaluation scenarios

To thoroughly validate the robustness and practicality of CADD across diverse network conditions, we design three evaluation scenarios that assess its performance in three different dimensions: scalability, adaptability, and data fidelity. The first scenario evaluates the scalability of CADD by analyzing how its performance evolves as the number of end devices increases. This allows us to assess the impact of network density on reliability, energy consumption, and channel congestion, comparing our approach with existing baselines. The second scenario focuses on adaptability by examining how CADD dynamically adjusts the duty cycle of each end device. By fixing the number of devices and observing variations in duty cycle allocation, we analyze how CADD balances resource usage across the channel and its effect on communication performance. While lower duty cycles reduce interference and channel congestion, they also limit transmission opportunities, potentially affecting the timely delivery of data. Finally, the third scenario evaluates data fidelity by analyzing the performance of the SAX-based data reduction mechanism implemented at the end device. In particular, we assess whether the SAX-based suppression of redundant readings leads to any meaningful loss of information at the gateway, ensuring that the reduction process itself does not compromise data fidelity. We compare raw sensor data against

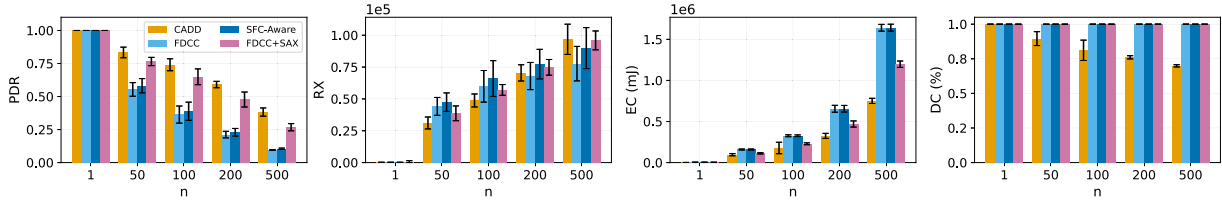


Fig. 1. Comparison of the performance of CADD and baseline schemes (FDCC and SFC-Aware) as the number of end devices (EDs) increases. The plots show the average Packet Delivery Ratio (PDR), successfully decoded packets at the gateway (RX), total Energy Consumption (EC), and the average Duty Cycle (DC) assigned to the EDs. Error bars indicate standard deviations across different random seeds.

reduced representation, quantifying the energy efficiency achieved and the impact of data reduction. The evaluation is conducted as a function of the number of SAX segments w , which directly controls the granularity of the reduced data and, consequently, the trade-off between fidelity and energy savings. The first two scenarios are simulated for $T = 3600$ time units (equivalent to one hour of operation), and each experiment runs for a total of $5T = 5 \times 3600$ time units (approximately 5 hours), using five different random seeds for the stochastic nature of event generation and channel access. We report both the average and standard deviation across runs. Table 2 summarizes the main parameters used across all the scenarios.

6. Evaluation results

6.1. Scalability under increasing network density

This scenario is designed to evaluate the scalability of CADD as the number of end devices in the network increases. As network density grows, LoRaWAN faces a higher risk of packet collisions and interference, especially under duty cycle constraints. The aim is to determine how well CADD maintains communication reliability and energy efficiency under such conditions. To achieve this, we progressively increase the number of active end devices while keeping all other network parameters fixed. Concretely, the density is swept from $n = 1$ to $n = 500$ EDs, a range intentionally chosen to simulate dense LoRaWAN deployments where congestion effects become prominent, and widely adopted in large-scale performance studies of LoRaWAN networks [78–80]. Fig. 1 presents a comparative analysis of four key metrics: the average Packet Delivery Ratio (PDR), the number of successfully decoded packets (RX), total Energy Consumption (EC), and the average Duty Cycle (DC). As the number of EDs increases, CADD consistently outperforms the baselines in reliability and efficiency. With $n = 500$ EDs, the most challenging configuration in our evaluation, CADD achieves a PDR about $4 \times$ higher than FDCC, $3.6 \times$ higher than SFC-Aware, and approximately $1.4 \times$ higher than FDCC + SAX. The comparison between FDCC and FDCC + SAX shows that static data reduction improves reliability, as FDCC + SAX consistently achieves higher PDR. However, this gain saturates at high network densities, and FDCC + SAX remains clearly below CADD, indicating that data reduction without duty cycle adaptation is insufficient to mitigate congestion effects. At this density, in terms of successfully decoded packets at the gateway, CADD delivers 26% more packets than FDCC, 6% more than SFC-Aware, and 1% than FDCC + SAX. This gain is noteworthy because, although CADD transmits fewer packets overall due to the SAX-based reduction and adaptive duty cycle, a larger proportion of them is successfully received, maximizing the utility of each transmission while avoiding the energy consumption of redundant or low-probability packets. In terms of energy efficiency, CADD reduces energy consumption by 54% compared to FDCC and SFC-Aware. At this density, FDCC + SAX already lowers energy consumption by approximately 27% with respect to FDCC, while CADD achieves an additional reduction of about 37% compared to FDCC + SAX, indicating that jointly adapting the duty cycle and data reduction provides further energy efficiency gains under congestion. Concerning the duty cycle, all schemes operate at the regulatory limit across all densities, since no adjustment is applied. As the number of end devices increases, CADD progressively adapts, often reducing the duty cycle, to mitigate co-channel interference and collisions while considering device priority and channel conditions. Despite this reduction, PDR remains high, and the number of successfully received packets exceeds that of the baselines, showing that transmission opportunities are used more efficiently rather than more frequently. Overall, these results demonstrate that CADD maintains high reliability and energy efficiency, even under severe channel congestion.

6.2. Impact of adaptive resource allocation

This scenario examines the impact of dynamically adjusting the duty cycle of each end device in CADD as a function of end device priority. For this purpose, we set the number of end devices to 300 and observe how priorities and duty cycles are distributed among them. The goal is to understand which devices are prioritized and how transmission opportunities are allocated based on device characteristics and channel conditions.

Fig. 2 (left) shows the distribution of end devices according to their assigned priority. The histogram indicates that the network contains end devices with varying priorities following a normal distribution. Fig. 2 (center) illustrates a clear increasing trend: devices with higher priority are assigned larger duty cycles, allowing them to transmit more frequently. Conversely, devices with lower priority are allocated significantly low duty cycles, limiting their transmission opportunities. This reflects how CADD prioritizes

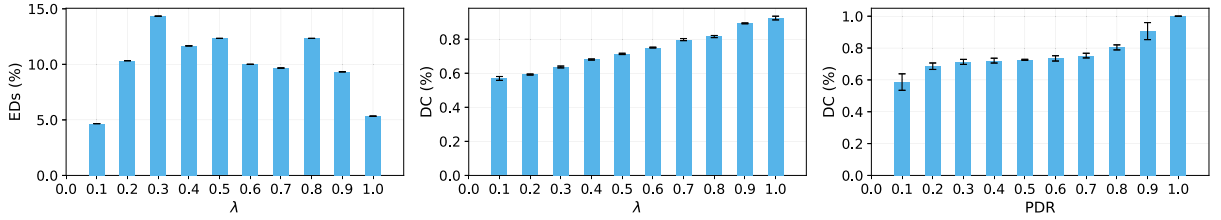


Fig. 2. Distribution and allocation of duty cycles under CADD for 300 end devices. The left subplot shows the density of devices according to their priority (λ). The center subplot reports the average duty cycle assigned to devices grouped by priority, while the right subplot reports the average duty cycle grouped by Packet Delivery Ratio (PDR). In the center and right plots, the lines show the mean duty cycle within each group, with shaded areas indicating the standard deviation across different random seeds.

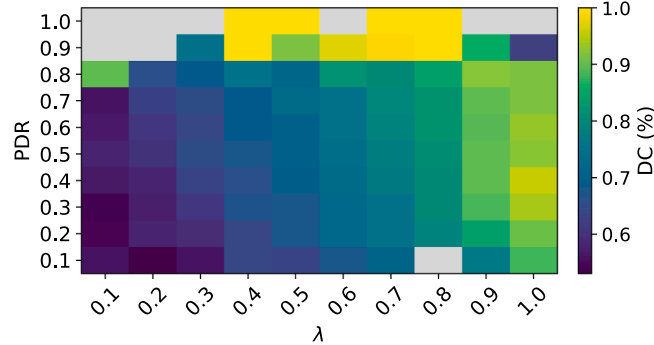


Fig. 3. Empirical heatmap of the average duty cycle selected by CADD as a function of device priority (λ) and the observed Packet Delivery Ratio (PDR), obtained by binning simulation samples.

channel utility, favoring high-priority data while removing less critical transmissions to optimize energy consumption and channel usage. Packet loss in low-priority devices is, therefore, more acceptable, as the information they generate is less relevant at the application level. Fig. 2 (right) examines how the assigned duty cycle correlates with the measured delivery ratio. The results indicate that devices with low PDRs, i.e., those whose transmissions are less often successfully received by the gateway, tend to receive lower duty cycles. This behavior reflects CADD's strategy of avoiding allocating excessive transmission opportunities to devices with reduced delivery performance. Instead, CADD concentrates resources on devices with higher PDRs, where transmissions are more likely to be successful. Taken together, the center and right subplots highlight how CADD jointly considers device priority and channel reliability when allocating duty cycles, ensuring that scarce transmission opportunities are directed toward data that is both important and more likely to reach the gateway, thus maximizing overall channel efficiency. Fig. 3 integrates the insights previously presented in the center and right subplots of Fig. 2 by representing them within a unified heatmap. This visualization jointly encodes device priority and delivery performance, illustrating how the duty cycle values selected by CADD are distributed across different combinations of both metrics. In general, regions associated with higher priority and stronger delivery ratios tend to exhibit higher duty cycle values, whereas areas reflecting lower priority or poorer reliability are typically associated with reduced allocations. Deviations from this general trend arise because the duty cycle adaptation also accounts for additional network variables such as interference level, effective delivery rate, and network density. Empty or sparsely populated regions correspond to combinations of priority and delivery performance that do not arise in the evaluated scenarios and therefore have no representative samples in the heatmap.

6.3. Data fidelity under SAX-based reduction

This scenario evaluates the effectiveness of the SAX-based data reduction mechanism implemented at end devices by analyzing its impact on data fidelity and energy efficiency. As introduced in Section 4, data fidelity refers to how well the reconstructed signal approximates the original sensor data after reduction. Since the application reconstructs the time series using linear interpolation of the transmitted values, we use the Mean Absolute Error (MAE) to quantify the deviation between the raw data and its reconstructed version. Our approach reduces the number of transmissions by filtering out consecutive values that remain within similar ranges, which is crucial for saving energy while preserving the most relevant information. Fig. 4 illustrates this trade-off by applying SAX with different granularities ($w = \{4, 16, 36, 72\}$) to temperature and humidity traces from a representative sensor. As expected, finer resolution ($w = 72$) closely follows the raw signal, while coarser resolutions ($w = 4$) and $w = 16$ discard more intermediate values, significantly reducing the number of transmissions. Intermediate settings, such as $w = 36$, offer a balance, capturing main fluctuations while filtering out smaller variations. To generalize this analysis, Fig. 5 quantifies the average and standard deviation of the MAE and energy consumption as a function of w , using multiple datasets from different sensors. The results confirm that larger w values lead to higher energy consumption but lower reconstruction error, whereas smaller w values reduce energy consumption by more than an

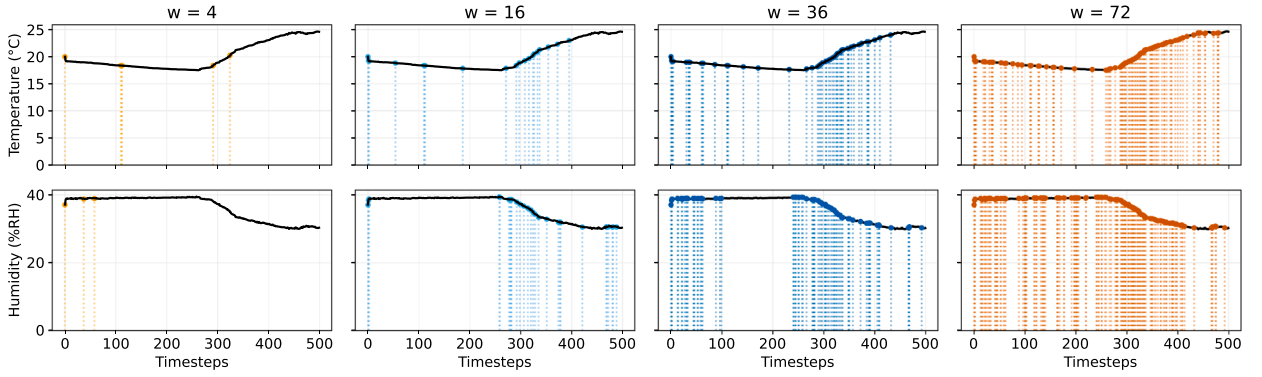


Fig. 4. Example of SAX-based reduction applied to temperature and humidity traces of a representative sensor. The plots compare the raw signal with its reduced versions using different granularities, $w = \{4, 16, 36, 72\}$, where vertical lines represent transmitted data points.

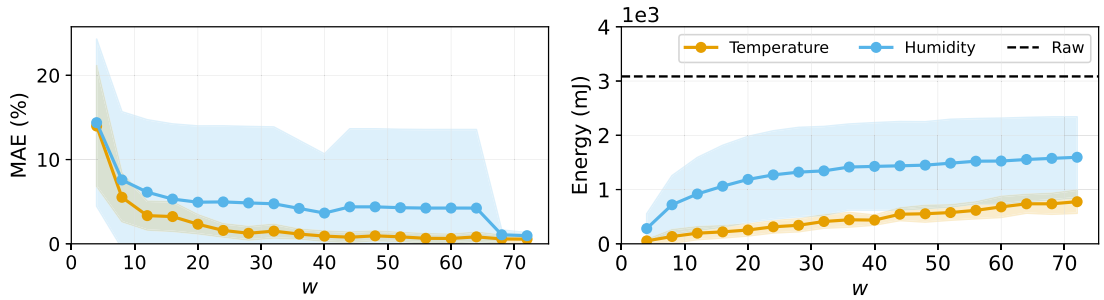


Fig. 5. Average and standard deviation of the Mean Absolute Error (MAE) and Energy Consumption (EC) for temperature and humidity sensors as a function of SAX granularity w . The EC is compared against the raw data transmission of sensors.

order of magnitude at the cost of a moderate increase in the MAE. Importantly, even with coarse resolutions (e.g., around $w = 16$), the MAE remains below 4%, showing that SAX-based reduction can significantly lower transmission overhead while maintaining acceptable information quality. A considerable degradation occurs at very small w values ($w = 4$), where the signal is too sparsely represented to capture relevant variations, and these settings are illustrative rather than practical. This behavior is advantageous, as it demonstrates that in the practical operating range of w , our method preserves high data fidelity. At the same time, the reduction in the number of transmitted samples yields substantial energy savings compared to raw transmission. Even in the case of the highest granularity ($w = 72$), where the number of transmitted samples is largest and the reduction is smallest, energy consumption is about 72% lower for temperature and 48% lower for humidity than raw transmission, while the MAE remains below 1% for both variables. Together, these results confirm that SAX-based reduction achieves an effective balance between energy efficiency and information quality.

7. Conclusions

In this work, we introduced CADD, a novel congestion control mechanism for LoRaWAN networks that jointly combines dynamic data reduction at the end device and adaptive duty cycle at the gateway. Our approach enhances the scalability of LoRaWAN by jointly considering local device constraints and global channel conditions, using signal quality and interference indicators to guide transmission decisions. By dynamically adjusting the duty cycle based on these indicators and data priority levels, CADD can effectively prioritize critical information while suppressing redundant transmissions, thus reducing energy consumption and preventing channel overload. Our evaluation demonstrates that CADD significantly improves scalability and energy efficiency compared to baseline schemes. In Scenario 1, with up to 500 devices, CADD achieves up to a $4 \times$ higher PDR than FDCC and $3.6 \times$ higher than SFC-Aware, while delivering 26% more successfully decoded packets than FDCC and 6% more than SFC-Aware, reducing energy consumption by around 54%. Scenario 2 confirmed that CADD effectively allocates duty cycle based on data priority and delivery performance, ensuring reliable transmission of critical data. Finally, Scenario 3 showed that the SAX-based reduction mechanism achieves energy savings of about 72% for temperature sensors and 48% for humidity sensors at the highest granularity ($w = 72$) while keeping the MAE below 1%. Even with coarser reductions (e.g., $w = 16$), the MAE remains under 4%, confirming that SAX provides substantial energy efficiency gains without compromising information quality. In short, CADD provides a scalable and energy-efficient approach that balances data fidelity and network reliability. In future work, we will focus on extending CADD to multi-gateway deployments and incorporating adaptive spreading factor selection. Additionally, real-world experiments with LoRaWAN hardware will be conducted to validate the performance of CADD in dynamic and large-scale environments. In particular, future work will investigate reliable

interference estimation methods based on measurements from real LoRaWAN deployments, addressing the inherent partial observability of interference at the gateway. We also plan to investigate the integration of complementary lightweight techniques, such as local predictive models, to better capture abrupt or non-linear variations in data sources with faster or more irregular dynamics within the same adaptive framework, and to enable a more comprehensive comparison against alternative data reduction approaches. Finally, future work will also include an explicit analysis of packet loss effects on data fidelity.

CRedit authorship contribution statement

Laura Acosta-Garcia: Writing – review & editing; **Juan Aznar-Poveda:** Writing – review & editing; **Stefan Pedratscher:** Writing – review & editing, Validation, Software, Methodology, Formal analysis; **Marlon Etheredge:** Writing – review & editing, Visualization, Software, Formal analysis, Data curation; **Rafael Asorey-Cacheda:** Writing – review & editing; **Joan Garcia-Haro:** Writing – review & editing; **Thomas Fahringer:** Writing – review & editing, Supervision, Project administration, Funding acquisition, Formal analysis; **Antonio-Javier Garcia-Sanchez:** Writing – review & editing.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the grant PID2023-148214OB-C21 funded by MICIU/AEI/10.13039/501100011033 and by FEDER/EU. This work was also supported by the PCI2024-153485 grant funded by MICIU/AEI/10.13039/501100011033 and the European Union; the PRIMA Program under Grant Agreement No. 2431 (FUSION: Comprehensive and sustainable solution to minimize food loss and waste and promote food security in the Mediterranean region); the result of the ThinkInAzul and AgroAINext programs, supported by MICIU with funding from the European Union NextGenerationEU (PRTR-C17.I1) and by Fundación Séneca with funding from Comunidad Autónoma Región de Murcia (CARM); and the Recovery and Resilience Plan of the Slovak Republic under the call "Transformation and Innovation Consortia" (project code: 09I02-03-V01-00012).

During the preparation of this work, the authors used ChatGPT solely to assist with minor textual edits and improvements to the flow of the manuscript. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

- [1] J.P.S. Sundaram, W. Du, Z. Zhao, A survey on LoRa networking: research problems, current solutions, and open issues, *IEEE Commun. Surv. Tutorials* 22 (1) (2019) 371–388.
- [2] P. Gkotsiopoulos, D. Zorbas, C. Douligeris, Performance determinants in LoRa networks: a literature review, *IEEE Commun. Surv. Tutorials* 23 (3) (2021) 1721–1758.
- [3] S. Devalal, A. Karthikeyan, Lora technology-an overview, in: 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE, 2018, pp. 284–290.
- [4] H. Bougaddou, N. Cherkaoui, O. Mouhib, Long-range, low-power for IoT devices: the lora network a review, in: 2024 Mediterranean Smart Cities Conference (MSCC), IEEE, 2024, pp. 1–6.
- [5] L. Aldhaheri, N. Alshehhi, I.I.J. Manzil, R.A. Khalil, S. Javaid, N. Saeed, M.-S. Alouini, Lora communication for agriculture 4.0: opportunities, challenges, and future directions, *IEEE Internet Things J.* 12 (2) (2024) 1380–1407.
- [6] J. Botero-Valencia, L. Castano-Londono, D. Marquez-Viloria, M. Rico-Garcia, Data reduction in a low-cost environmental monitoring system based on lora for WSN, *IEEE Internet Things J.* 6 (2) (2018) 3024–3030.
- [7] A. Pagano, D. Croce, I. Tinnirello, G. Vitale, A survey on LoRa for smart agriculture: current trends and future perspectives, *IEEE Internet Things J.* 10 (4) (2022) 3664–3679.
- [8] S. Wielandt, S. Uhlemann, S. Fiolleau, B. Dafflon, TDD LoRa and delta encoding in low-power networks of environmental sensor arrays for temperature and deformation monitoring, *J. Signal Process Syst.* 95 (7) (2023) 831–843.
- [9] B. Cengiz, I.Y. Adam, M. Ozdem, R. Das, A survey on data fusion approaches in IoT-based smart cities: smart applications, taxonomies, challenges, and future research directions, *Inf. Fusion* 121 (2025) 103102.
- [10] G. Baldini, R. Giuliani, G. Steri, I. Sanchez, C. Gentile, The application of the symbolic aggregate approximation algorithm (SAX) to radio frequency fingerprinting of IoT devices, in: 2017 IEEE Symposium on Communications and Vehicular Technology (SCVT), IEEE, 2017, pp. 1–6.
- [11] A.K.M. Al-Qurabat, S.A. Abdulzahra, A.K. Idrees, Two-level energy-efficient data reduction strategies based on SAX-LZW and hierarchical clustering for minimizing the huge data conveyed on the internet of things networks, *J. Supercomput.* 78 (16) (2022) 17844–17890.
- [12] V. Jha, P. Tripathi, Probabilistic SAX: a cognitively-inspired method for time series classification in cognitive iot sensor network, *Mobile Netw. Appl.* 29 (3) (2024) 809–824.
- [13] M. Patel, Anomaly Detection in Heterogeneous IoT Systems: Leveraging Symbolic Encoding of Performance Metrics for Anomaly Classification, Master's thesis, California Polytechnic State University, 2024.
- [14] I. Kevin, K. Wang, S. Wu, A. Ivoghlian, Z. Salcic, A. Austin, X. Zhou, LWS: a LoRaWAN wireless underground sensor network simulator for agriculture applications, in: 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI), IEEE, 2019, pp. 475–482.

- [15] S.J. Johnston, P.J. Basford, F.M.J. Bulot, M. Apetroaie-Cristea, G.L. Fosterx, M. Loxham, S.J. Cox, IoT Deployment for city scale air quality monitoring with low-power wide area networks, in: 2018 Global Internet of Things Summit (GloTS), IEEE, 2018, pp. 1–6.
- [16] E. Levintal, Y. Ganot, G. Taylor, P. Freer-Smith, K. Suvocarev, H.E. Dahlke, An underground, wireless, open-source, low-cost system for monitoring oxygen, temperature, and soil moisture, *Soil Discuss.* 2021 (2021) 1–22.
- [17] J. Dafonte, M.Á. González, E. Comesaña, M.T. Teijeiro, J.J. Cancela, Soil water status monitoring system with proximal low-Cost sensors and lora technology for smart water irrigation in woody crops, *Sensors* 24 (24) (2024) 8104.
- [18] M. Jouhari, N. Saeed, M.-S. Alouini, E.M. Amhoud, A survey on scalable LoRaWAN for massive IoT: recent advances, potentials, and challenges, *IEEE Commun. Surv. Tutorials* 25 (3) (2023) 1841–1876.
- [19] M.S.A. Muthanna, A. Muthanna, A. Rafiq, M. Hammoudeh, R. Alkanhel, S. Lynch, A.A. Abd El-Latif, Deep reinforcement learning based transmission policy enforcement and multi-hop routing in QoS aware LoRa IoT networks, *Comput. Commun.* 183 (2022) 33–50.
- [20] G. El Helou, M. Ibrahim, R. El Tawil, Y. Mohanna, QoS based IoT applications in LoRa networks using multi-Dimensional allocation strategy, *IEEE Access* (2024).
- [21] E. Lima, J. Moraes, H. Oliveira, E. Cerqueira, S. Zeadally, D. Rosário, Adaptive priority-aware LoRaWAN resource allocation for internet of things applications, *Ad Hoc Netw.* 122 (2021) 102598.
- [22] L. Specification, LoRaWAN specification v1. 1, LoRaWANtm-specification-v11 (2017).
- [23] J. Haxhibeqiri, E. De Poorter, I. Moerman, J. Hoebeke, A survey of LoRaWAN for IoT: from technology to application, *Sensors* 18 (11) (2018) 3995.
- [24] R.P. Centelles, F. Freitag, R. Meseguer, L. Navarro, Beyond the star of stars: an introduction to multihop and mesh for LoRa and LoRaWAN, *IEEE Pervasive Comput.* 20 (2) (2021) 63–72.
- [25] A.M. Yousuf, E.M. Rochester, B. Ousat, M. Ghaderi, Throughput, coverage and scalability of LoRa LPWAN for internet of things, in: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), IEEE, 2018, pp. 1–10.
- [26] M. Housse, C. Caillouet, A. Duda, Performance of unslotted ALOHA with capture and multiple collisions in LoRaWAN, *IEEE Internet Things J.* 10 (20) (2023) 17824–17838.
- [27] B. Shilpa, H.P. Gupta, R.K. Jha, S.S. Hashmi, LoRa interference issues and solution approaches in dense IoT networks: a review, *Telecommun. Syst.* 87 (2) (2024) 517–539.
- [28] X. Zhang, Z. Lin, S. Gong, B. Gu, D. Niyato, Multiagent reinforcement learning with an attention mechanism for improving energy efficiency in lora networks, in: GLOBECOM 2023-2023 IEEE Global Communications Conference, IEEE, 2023, pp. 4152–4157.
- [29] C. Lehong, B. Isong, F. Lugayizi, A. Abu-Mahfouz, A spreading factor congestion status-aware adaptive data rate algorithm, *J. Sensor Actuator Netw.* 10 (4) (2021) 70.
- [30] S. Kim, Y. Yoo, Contention-aware adaptive data rate for throughput optimization in LoRaWAN, *Sensors* 18 (6) (2018) 1716.
- [31] N. El Rachkidy, A. Guitton, M. Kaneko, Collision resolution protocol for delay and energy efficient LoRa networks, *IEEE Trans. Green Commun. Netw.* 3 (2) (2019) 535–551.
- [32] A. Farhad, D.-H. Kim, P. Sthapit, J.-Y. Pyun, Interference-aware spreading factor assignment scheme for the massive LoRaWAN network, in: 2019 International Conference on Electronics, Information, and Communication (ICEIC), IEEE, 2019, pp. 1–2.
- [33] L. Acosta-García, J. Aznar-Poveda, A.-J. García-Sánchez, J. García-Haro, T. Fahringer, Proactive adaptation of data rate in mobile LoRa-based IoT devices using machine learning, in: 2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring), IEEE, 2024, pp. 1–5.
- [34] L. Acosta-García, J. Aznar-Poveda, A.J. García-Sánchez, J. Hollenstein, J. García-Haro, T. Fahringer, ADRL: a reconfigurable energy-efficient transmission policy for mobile LoRa devices based on reinforcement learning, *Internet Things* 31 (2025) 101577.
- [35] A. Petroni, M. Biagi, Interference mitigation and decoding through gateway diversity in LoRaWAN, *IEEE Trans. Wireless Commun.* 21 (11) (2022) 9068–9081.
- [36] R.K. Singh, R. Berkvens, M. Weyn, Synchronization and efficient channel hopping for power efficiency in LoRa networks: a comprehensive study, *Internet of Things* 11 (2020) 100233.
- [37] G. Gaillard, C. Pham, CANL LoRa: collision avoidance by neighbor listening for dense lora networks, in: 2023 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2023, pp. 1293–1298.
- [38] O. Väinänen, T. Hämäläinen, LoRa-based sensor node energy consumption with data compression, in: 2021 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0 & IoT), IEEE, 2021, pp. 6–11.
- [39] Y. Fathy, P. Barnaghi, R. Tafazolli, An adaptive method for data reduction in the internet of things, in: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), IEEE, 2018, pp. 729–735.
- [40] G. Anand, M. Vyas, R.N. Yadav, S.K. Nayak, On reducing data transmissions in fog-enabled lora-based smart agriculture, *IEEE Internet Things J.* 11 (5) (2023) 8894–8905.
- [41] F. Nadeem, T. Fahringer, Using templates to predict execution time of scientific workflow applications in the grid, in: 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009, pp. 316–323. <https://doi.org/10.1109/CCGRID.2009.77>
- [42] J. Bai, Z. Zeng, K.M. Abualnaja, N.N. Xiong, ADCC: An effective adaptive duty cycle control scheme for real time big data in green IoT, *Alexandria Eng. J.* 61 (8) (2022) 5959–5975.
- [43] S. Jeyalakshmi, R.M. Ganesh, Adaptive duty-cycle scheduling using bi-directional long short-term memory (BiLSTM) for next generation IoT applications, in: 2022 International Conference on Computing, Communication and Power Technology (IC3P), IEEE, 2022, pp. 75–80.
- [44] M. Lassnig, T. Fahringer, V. Garonne, A. Molfetas, M. Branco, Identification, modelling and prediction of non-periodic bursts in workloads, *CCGRID '10, IEEE Computer Society, USA*, 2010, p. 485–494. <https://doi.org/10.1109/CCGRID.2010.118>
- [45] T. Zainab, L. Harms, J. Karstens, O. Landsiedel, Foresee: ML-driven, communication-Efficient time-series forecasting, in: 2025 21st International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT), IEEE, 2025, pp. 118–126.
- [46] U. Raza, A. Camerra, A.L. Murphy, T. Palpanas, G.P. Picco, What does model-driven data acquisition really achieve in wireless sensor networks?, in: 2012 IEEE International Conference on Pervasive Computing and Communications, IEEE, 2012, pp. 85–94.
- [47] L. Acosta-García, J. Aznar-Poveda, A.J. García-Sánchez, J. García-Haro, T. Fahringer, Dynamic transmission policy for enhancing LoRa network performance: a deep reinforcement learning approach, *Internet Things* 24 (2023) 100974.
- [48] O. Väinänen, T. Hämäläinen, Efficiency of temporal sensor data compression methods to reduce LoRa-based sensor node energy consumption, *Sensor Rev.* 42 (5) (2022) 503–516.
- [49] W.M. Ismael, M. Gao, A.A. Al-Shargabi, A. Zahary, An in-networking double-layered data reduction for internet of things (IoT), *Sensors* 19 (4) (2019) 795.
- [50] T. Shu, J. Chen, V.K. Bhargava, C.W. de Silva, An energy-efficient dual prediction scheme using LMS filter and LSTM in wireless sensor networks for environment monitoring, *IEEE Internet Things J.* 6 (4) (2019) 6736–6747.
- [51] H. Cheng, Z. Xie, L. Wu, Z. Yu, R. Li, Data prediction model in wireless sensor networks based on bidirectional LSTM, *EURASIP J. Wirel Commun. Netw.* 2019 (1) (2019) 203.
- [52] C. Salim, N. Mitton, K-predictions based data reduction approach in WSN for smart agriculture, *Computing* 103 (3) (2021) 509–532.
- [53] S. Suryavansh, A. Benna, C. Guest, S. Chaterji, A data-driven approach to increasing the lifetime of IoT sensor nodes, *Sci. Rep.* 11 (1) (2021) 22459.
- [54] Z. Ahmed, M.M. Rehan, R. Chughtai, M.W. Rehan, Ad-rc: a novel adaptive dynamic radio duty cycle mechanism for low-power iot devices, *IEEE Internet Things J.* 9 (15) (2022) 13376–13389.
- [55] R. Dhall, H. Agrawal, An improved energy efficient duty cycling algorithm for IoT based precision agriculture, *Procedia Comput. Sci.* 141 (2018) 135–142.
- [56] N. Charef, Energy sustainable reinforcement learning-based adaptive duty-cycling in wireless sensor networks-based internet of things networks, 2023.
- [57] W. Gao, W. Du, Z. Zhao, G. Min, M. Singhal, Towards energy-Fairness in lora networks, in: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), IEEE, 2019, pp. 788–798.
- [58] L.A. García, J.A. Poveda, F. Margreiter, A.-J.G. Sánchez, J.G. Haro, T. Fahringer, J.L. López, J.-V. Rodríguez, FAST-LoRa: an efficient simulation framework for evaluating LoRaWAN networks and transmission parameter strategies, 2025. <https://arxiv.org/abs/2507.23342>.

- [59] M.C. Bor, U. Roedig, T. Voigt, J.M. Alonso, Do LoRa low-power wide-area networks scale?, in: *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2016, pp. 59–67.
- [60] M. Slabicki, FLoRa - a framework for LoRa simulations with OMNeT++ , 2022. <https://flora.aalto.fi/> [Accessed: 18.05.2024].
- [61] L. Amichi, M. Kaneko, E.H. Fukuda, N. El Rachkidy, A. Guitton, Joint allocation strategies of power and spreading factors with imperfect orthogonality in LoRa networks, *IEEE Trans. Commun.* 68 (6) (2020) 3750–3765.
- [62] V. Toro-Betancur, G. Premasankar, M. Slabicki, M. Di Francesco, Modeling communication reliability in LoRa networks with device-level accuracy, in: *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, IEEE, 2021, pp. 1–10.
- [63] Z. Xu, J. Luo, Z. Yin, S. Wang, C. Chen, J. Lin, R. Xiong, T. He, X-MAC: achieving high scalability via imperfect-orthogonality aware scheduling in LPWAN, in: *2022 IEEE 30th International Conference on Network Protocols (ICNP)*, IEEE, 2022, pp. 1–11.
- [64] J.-T. Lim, Y. Han, Spreading factor allocation for massive connectivity in LoRa systems, *IEEE Commun. Lett.* 22 (4) (2018) 800–803.
- [65] J. Lin, E. Keogh, S. Lonardi, B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, in: *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, 2003, pp. 2–11.
- [66] F.M.R. Junior, R.A.C. Bianchi, R.C. Prati, K. Kolehmainen, J.-P. Soininen, C.A. Kamiński, Data reduction based on machine learning algorithms for fog computing in IoT smart agriculture, *Biosyst. Eng.* 223 (2022) 142–158.
- [67] T. Blu, P. Thévenaz, M. Unser, Linear interpolation revitalized, *IEEE Trans. Image Process.* 13 (5) (2004) 710–719.
- [68] J. Moraes, N. Matni, A. Riker, H. Oliveira, E. Cerqueira, C. Both, D. Rosário, An efficient heuristic loraWAN adaptive resource allocation for IoT applications, in: *2020 IEEE Symposium on Computers and Communications (ISCC)*, IEEE, 2020, pp. 1–6.
- [69] J. Moraes, H. Oliveira, E. Cerqueira, C. Both, S. Zeadally, D. Rosário, Evaluation of an adaptive resource allocation for lorawan, *J. Signal Process. Syst.* 94 (1) (2022) 65–79.
- [70] V. Di Vincenzo, M. Heusse, B. Tourancheau, Improving downlink scalability in LoRaWAN, in: *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019, pp. 1–7.
- [71] V. Gupta, S.K. Devar, N.H. Kumar, K.P. Bagadi, Modelling of IoT traffic and its impact on LoRaWAN, in: *GLOBECOM 2017-2017 IEEE Global Communications Conference*, IEEE, 2017, pp. 1–6.
- [72] Overview - SimPy 4.1.2.dev8 + g81c7218 documentation. [Online; accessed 2025-04-23], <https://simpy.readthedocs.io/en/latest/>.
- [73] S. Kim, H. Lee, S. Jeon, An adaptive spreading factor selection scheme for a single channel LoRa modem, *Sensors* 20 (4) (2020) 1008.
- [74] C. Zhong, X. Nie, A novel single-channel edge computing LoRa gateway for real-time confirmed messaging, *Sci. Rep.* 14 (1) (2024) 8369.
- [75] R. Mishra, S. Godhaneeya, A. Soni, Minimizing intra-Interference in LoRaWAN for seamless connectivity in ubiquitous networking, in: *2024 16th International Conference on Communication Systems & NETWORKS (COMSNETS)*, IEEE, 2024, pp. 123–128.
- [76] O.D.D. Brito, F. De Rango, P. Pace, Optimized adaptive interference-aware rate (AIAR) strategy for LoRaWAN networks in co-channel interference environments, in: *2025 International Conference on Smart Applications, Communications and Networking (SmartNets)*, IEEE, 2025, pp. 1–4.
- [77] Intel Lab Data. [Online; accessed 2025-06-30], <https://db.csail.mit.edu/labdata/labdata.html>.
- [78] V. Torres-Sanz, J.A. Sanguesa, F.J. Martinez, P. Garrido, C.T. Calafate, An autoconfiguration strategy for very large scale long range wide area network deployments in smart cities, *IET Smart Cities* 7 (1) (2025) e12096.
- [79] J.C. da Silva, D. d. L. Flor, V.A. de Sousa Junior, N.S. Bezerra, A.A.M. de Medeiros, A survey of LoRaWAN simulation tools in ns-3, *J. Commun. Inf. Syst.* 36 (1) (2021) 17–30.
- [80] N.I.M. Osman, E.B. Abbas, Performance evaluation of LoRa and sigfox LPWAN technologies for IoT, *Acad. J. Res. Sci. Publ* 4 (10.52132) (2022).