Probabilistic Performance Evaluation of the Class-A device in LoRaWAN protocol on the MAC layer

Mi Chen, Lynda Mokdad, Jalel Ben-Othman, Jean-Michel Fourneau

Probabilistic Performance Evaluation of the Class-A device in LoRaWAN protocol on the MAC layer

Mi CHEN[a], Lynda Mokdad[a], Jalel Ben-Othman[b,c,d], Jean-Michel Fourneau[e]

[a] *Univ Paris Est Creteil, LACL, F-94010, Creteil, Francee*
[b] *Univ Paris-Saclay, CNRS, CentraleSupélec Lab. L2S, Gif-sur-Yvette, 91190, France*
[c] *Univ Sorbonne Paris Nord, France*
[d] *College of Technological Innovation, Zayed University, UAE*
[e] *Univ. Paris Saclay, UVSQ, lab. DAVID, France*

Abstract

LoRaWAN is a network technology that provides a long-range wireless network while maintaining low energy consumption. It adopts the pure Aloha MAC protocol and the duty-cycle limitation at both uplink and downlink on the MAC layer to conserve energy. Additionally, LoRaWAN employs orthogonal parameters to mitigate collisions. However, synchronization in star-of-star topology networks and the complicated collision mechanism make it challenging to conduct a quantitative performance evaluation in LoRaWAN. Our previous work proposes a Probabilistic Timed Automata (PTA) model to represent the uplink transmission in LoRaWAN. It is a mathematical model that presents the nondeterministic and probabilistic choice with time passing. However, this model remains a work in progress. This study extends the PTA model to depict Class-A devices in the LoRaWAN protocol. The complete characteristics of LoRaWAN's MAC layer, such as duty-cycle limits, bidirectional communication, and confirmed message transmission, are accurately modeled. Furthermore, a comprehensive collision model is integrated into the PTA. Various properties are verified using the probabilistic model checker PRISM, and quantitative properties are calculated under diverse scenarios. This quantitative analysis provides valuable insights into the performance and behavior of LoRaWAN networks under varying conditions.

Keywords: LoRaWAN, Performance Evaluation, Probabilistic model checking, Probabilistic Timed Automata, MAC Layer

1. INTRODUCTION

The Internet of Things (IoT) constitutes a pivotal network communication technology, facilitating the interconnection of objects. One such technology is LoRaWAN, a Low Power Wide Area (LPWA) wireless communications network [1]. It is widely used in IoT and machine-to-machine (M2M) networks due to its low-cost and bidirectional communication capabilities. Employing a spread spectrum technique, LoRaWAN enables end devices (EDs) to communicate over long distances using a star-of-star topology. Additionally, LoRaWAN operates on unlicensed ISM bands, such as the 868 MHz frequency in Europe [2], making it a cost-effective and easily implementable network solution. Consequently, LoRaWAN has gained widespread attention in various long-range IoT applications, including environmental monitoring [3] and intelligent city surveillance [4]. Moreover, LoRaWAN also shows its suitability on Intelligent Transportation Systems (ITS) in [5, 6, 7]. Given its advantages, LoRaWAN has become an essential complementary solution for 5G in massive IoT due to its vast potential [8].

With the rapid expansion of network requirements of LoRaWAN, network performance evaluation plays an essential role in research. However, most of the evaluation works are model-free or in black-box. The lack of transparency and accuracy in the mathematical models leads to a risk of mismatch between different research works. To systematically and comprehensively assess LoRaWAN's properties and performance, a model-based approach that permits both quantitative and qualitative evaluation is imperative.

Motivated by evaluating LoRaWAN's transmission formally and accurately in a model-based way, this study leverages probabilistic model checking, a model-based technique to globally and automatically establish system properties. In our previous works [9, 10], various probabilistic models were formulated to represent LoRaWAN's transmission. However, these models were primarily oriented towards the uplink transmission and still needed to be completed. This study comprehensively models the confirmed transmission procedure of two nodes in the LoRaWAN network with one gateway using the Probabilistic Timed Automata (PTA) model [11]. By integrating the time aspect, the proposed model can represent various mechanisms in LoRaWAN, such as the retransmission, the duty-cycle limitation, and the collision model. Several properties are verified using the model checker tool PRISM [12], and the quantitative properties are verified in different scenarios.

The principal contributions of this study are summarized as follows:

- Firstly, a Markov Decision Process (MDP) model for single-node transmission is investigated.
- Subsequently, by integrating the timing aspect and receiver components, the Probabilistic Timed Automata (PTA) model is investigated. It is a mathematical automaton model that represents a system with time passing, nondeterminism actions, and probabilistic transitions of states. A PTA model for the unconfirmed transmission of two nodes and one gateway in LoRaWAN is investigated in this part.
- Based on the concept of PTA, a comprehensive model for the confirmed transmission procedure is proposed. This model represents a complete bidirectional transmission procedure in LoRaWAN by integrating new downlink components, the bidirectional duty-cycle limitation, retransmission mechanism, etc. More importantly, this model introduces a novel approach to evaluate the network performance of LoRaWAN. By employing this model, performance evaluation of LoRaWAN networks can be conducted model-based, allowing for qualitative and quantitative verification of network properties.
- Several properties of the proposed PTA model are verified using the probabilistic model checker PRISM. The system's probabilistic properties are first verified. Then, several reward properties are verified by representing the PTA model with some abstraction techniques. The results show the accuracy of the proposed model, which further proves the feasibility of the probabilistic model checking for the performance evaluation of LoRaWAN networks.

The rest of this paper is organized as follows. After giving a comprehensive exploration of the LoRaWAN in Section 2, Section 3 presents a related literature survey. Section 4 investigates the MDP model for single-node transmission. The PTA model's syntax and the PTA's modeling for LoRaWAN are presented in Section 5. In Section 6, the properties and verification are given. The main contribution of this study is summarized, and perspectives for future work are provided in Section 7.

2. LoRaWAN NETWORKS

LoRaWAN (**Lo**ng **Ra**nge **W**ide **A**rea **N**etwork) is a wireless communication technology that has emerged as an energy-efficient solution for end-to-end communication in IoT applications [1]. Using a spread spectrum technique, end devices (EDs) in networks can communicates over long distances using a star-of-star topology as given in Figure 1. LoRaWAN operates on unlicensed ISM (Industrial Scientific and Medical) bands, such as 868 MHz in Europe [2], rendering it a cost-effective and easily implementable solution. With its great potential, LoRaWAN becomes one of the essential complementary solutions for 5G networks in massive IoT scenarios [8, 13]. Consequently, LoRaWAN has gained widespread attention and is adopted in long-range IoT applications, such as environmental monitoring [3] and intelligent city surveillance [4].

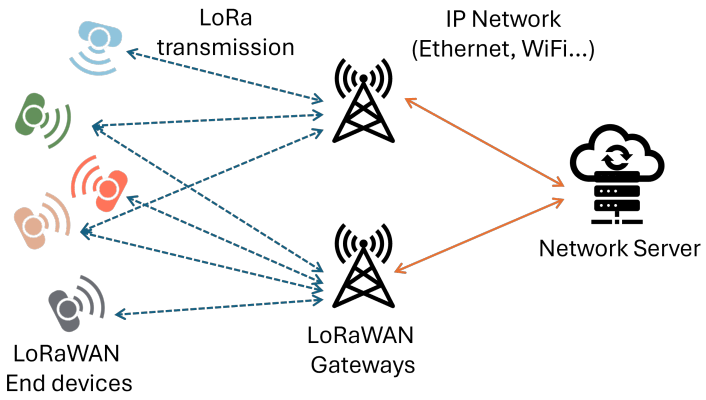
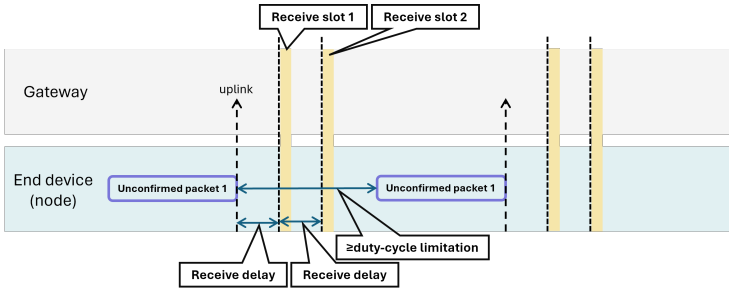


Figure 1: LoRaWAN architecture [1]

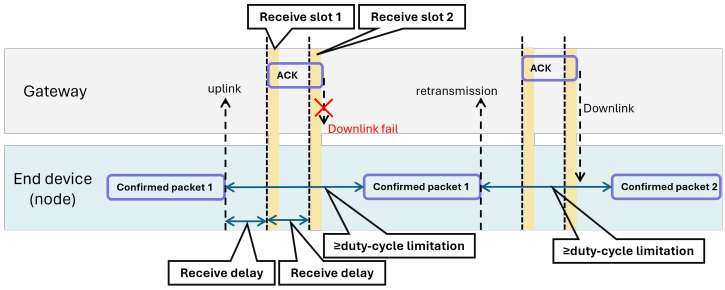
2.1. LoRaWAN transmission

To fulfill the low power consumption requirement, all end devices within the LoRaWAN network must support Class-A ("Aloha") communications. In this mode, nodes adopt a variant Aloha transmission schedule as shown in Figure 2. The node primarily operates in sleep mode and wakes up only when transmitting an uplink message in Aloha. After transmitting, the node opens two receiving windows to listen to downlink messages for both confirmed and unconfirmed transmission, as presented in Figure 2(a) and Figure 2(b). In

the confirmed transmission scenario, an ACK message is sent to the node in one of the two windows, depending on availability. If the node does not receive the ACK message in both windows, it tries to retransmit until the ACK message is received or the limit of retransmission times is exceeded as shown in Figure 2(b) [14]. Moreover, both uplink and downlink transmissions adhere to the duty-cycle limitation [2].



(a) Unconfirmed transmission



(b) Confirmed transmission

Figure 2: Class-A transmission schedule [14]

Based on the architecture and the shcdule, LoRaWAN adopts several mechanisms on transmission which is given in the following parts.

2.2. Transmission Parameters and the Spreading Factor (SF)

In the LoRaWAN network, nodes can transmit using a spread-spectrum modulation technique derived from Chirp Spread Spectrum (CSS) technology. This technology provides a balance between sensitivity and data rate, functioning within different transmission parameters, such as the Coding Rate (CR), the Bandwidth (BW), and the Spreading Factor (SF). Equation

(1) gives the calculation between the Data Rate (DR) and parameters in LoRaWAN.

$$DR = SF \times \frac{\left[\frac{4}{4+CR}\right]}{\left[\frac{2^{SF}}{BW}\right]} \tag{1}$$

In most regional parameters, the CR and the BW are fixed [2]. For example, in Europe (EU868), the BW is fixed to 125kHz, and the CR is fixed to 1. However, nodes in LoRaWAN networks can transmit at different data rates by selecting different SF, a near-orthogonal parameter. A lower SF can get higher data rates and lower energy consumption. Conversely, a higher SF can reduce the packet loss probability but result in lower data rates and increased energy consumption [15]. Equation (2) presents the packet airtime (AT) with payload size pl and SF [16]. In EU868 specification, $BW = 125kHz$, $CR = 1$, $DE = 0$ if $SF < 11$, otherwise $DE = 1$.

$$\begin{aligned} T_{sym} &= \frac{2^{SF}}{BW} \\ sym_{pl} &= 8 + \max\left(ceil\left(\frac{8pl - 4SF + 44}{(4(SF - 2DE))}\right)(CR + 4), 0\right) \\ AT &= (12.25 + sym_{pl}) \times T_{sym} \end{aligned} \tag{2}$$

2.3. Received Signal Strength Indication (RSSI)

The Received Signal Strength Indication (RSSI) is a relative measurement of the received signal's strength. LoRaWAN receivers can only receive the packet whose RSSI exceeds the thresholds. Using the Log-distance path loss model [17, 18, 19], the calculation of RSSI is given in (3). Where P_{tx} is the transmit power in dBm, PL_0 is the distance path loss d_0, η is the loss exponent and X_σ is the attenuation (in decibels) caused by flat fading which is modeled with a Gaussian random variable $X_\sigma \sim N(\mu, \sigma^2)$ with $\mu = 0$.

$$RSSI = P_{tx} - \left(PL_0 + 10\eta \log_{10}\left(\frac{d_i}{d_0}\right) + X_\sigma\right) \tag{3}$$

In LoRaWAN networks, different SFs have different RSSI thresholds. Table 1 gives an example of the thresholds (in dBm) at 125kHz of SX1276 receiver chip [20].

Table 1: RSSI thresholds at $BW = 125kHz$ of SX1276 receiver chip [20]

SF7	**SF8**	**SF9**	**SF10**	**SF11**	**SF12**
-123	-126	-129	-132	-133	-136

2.4. Collision Model

Since the SF is a near orthogonal parameter, the collision in LoRaWAN occurs primarily among the packets with the same SF and frequencies. Rahmadhani and Kuipers [21] give the collision conditions of two packets in LoRaWAN. Denoting the AT the packet receiving time, T_c the time that the receiver needs to lock the message, the collision model of two overlapped packets with the same SF and frequency is as follows:

1. If time difference between the arrivals of two packets $\Delta t \leq T_c$, then only the later packet survives if its RSSI p_{late} is great enough (6dBm greater than the earlier one). Otherwise, both packets are lost.

2. If $T_c < \Delta t \leq AT - T_c$, then only the earlier packet survives if its RSSI is great enough (6dBm greater than the later one). Otherwise, both packets are lost.

3. If $AT - T_c < \Delta t \leq AT$, then both packets survive.

In [21], the measurement shows that the receiver needs 3 symbols to lock the message. Therefore, according to Equation 2, $T_c = 3 \times T_{sym}$.

Based on the transmission mechanism above, this study focuses on the performance evaluation of Class-A transmission in LoRaWAN networks using probabilistic model checking. In the following section, several related literature is presented.

3. RELATED WORK

3.1. Performance Evaluation in LoRaWAN

As the demand for networks in LoRaWAN rapidly expands, LoRaWAN encounters challenges from multiple aspects, particularly in performance evaluation. Numerous research endeavors have been conducted to address various aspects, including transmission parameter allocation [22, 23, 24, 25], network modeling [26], and practical applications [3, 27]. Therefore, the evaluation

tool becomes imperative to facilitate comprehensive studies in network performance evaluation.

Bor et al. [28] and Van den Abeele et al. [29], evaluate the scalability of LoRaWAN by random event simulations using different simulators. Callebaut and Van der Perre [17] used real LoRaWAN test-beds to evaluate transmission sensitivity. Croce et al. [30] employed numerical simulations and LoRaWAN test-beds to assess the performance of transmission parameters. In our previous works [31, 32], we developed two simulators based on random event simulations and utilized them to evaluate network performance in different aspects, such as the impact of SF allocation [33, 34], network security [35, 36], and practical applications in ITS solution [7].

However, the evaluations and the experiments primarily rely on numerical and statistical analyses. Many are model-free or black-box, making it challenging to verify system properties exhaustively. Furthermore, the lack of transparency and accuracy in mathematical models increases the risk of mismatch between different research works. To formally evaluate the properties and performance of LoRaWAN, this study employs probabilistic model checking, a model-based method that enables the automated establishment of global system properties.

3.2. Probabilistic Model Checking

Probabilistic model checking is a technique for system modeling that presents probabilistic and non-deterministic behavior [37]. Various probabilistic models can be employed to express protocols or systems, and different logic specifications can be used to evaluate their properties and performance both qualitatively and quantitatively. Bordel et al. [38], use continuous-time Markov chains (CTMCs) as a modeling tool to verify the security properties of chaotic digital watermarking in IoT scenarios and future 5G networks. Using a turn-based stochastic game model, Aslanyan et al. [39] evaluate the attack-defense scenarios and apply the method to an RFID goods management case study. In [40], Unmanned Aerial Vehicle (UAV) security analysis was conducted using probabilistic model checking.

For the network protocols, Fruth [41] model the IEEE 802.15.4 (ZigBee) using Probabilistic Timed Automata (PTA) and verify the properties of contention resolution. Likewise, the IEEE 802.11 (WiFi) protocol was also modeled in PTA by Kwiatkowska et al. [42]. In [43, 44], the protocol IEEE 802.3 (Ethernet) and IEEE 1394 FireWire Root Contention are modeled and evaluated using the PTA model.

In our previous work [9], we developed a Markov Decision Process (MDP) model to analyze the uplink transmission of a class-A device in LoRaWAN. Solving the MDP model proposes an initialization strategy for parameter allocation algorithms. However, this model only considers one single node in the network. The collision probability of nodes is given only by estimation. Additionally, the MDP model can not represent the timing aspect, which is crucial for collision analysis in LoRaWAN.

To address these limitations, we extended our research in a subsequent study [10]. A PTA model for the uplink transmission of two nodes and one gateway is proposed by adding the timing aspect and the collision model mentioned above. The PTA models the unconfirmed transmission procedure of two nodes in the LoRaWAN network with one gateway. Using the model checker tool PRISM, several probabilistic properties of the network are verified under different cases.

However, there is still room for improvement in the models above. In the following part of this paper, several different models are investigated. After investigating the MDP model and the PTA model for uplink transmission proposed in [9] and [10], a novel PTA model is proposed as the main contribution of this paper. The novel model integrates novel features of the confirmed transmission procedures in LoRaWAN, such as the novel network component, the duty-cycle limitation, and the retransmission mechanism in LoRaWAN networks. The details of the models are given in the following sections.

4. MARKOV DECISION PROCESS MODELS

The Markov Decision Process (MDP) is a mathematical model of sequential decision. By defining different states and probabilistic transitions, an MDP can represent randomness strategies and rewards that can be achieved by the agent in an environment where the system state has Markov properties. An MDP can be described as a tuple $M =< S, A, P >$ where:

- S is the finite set of system states;
- A is the set of actions that the system can take;
- $P : S \times A \times S$ is the tensor of transitions between states. The elements in P are the probabilities of transitions denoted as $\mathbb{P}(s, a, s')$, which represent the probabilities of reaching state $s' \in S$ from state $s \in S$ by the action $a \in A$;

With the definition above, an MDP model for the bidirectional transmission of one class-A device is proposed in [9]. This model represents the bidirectional transmission procedure of a single node with nondeterministic SF selection. The overview of the MDP is shown in Figure 3. By denoting the maximum retransmission attempt of a packet as 8, the definition of each element in tuple $M =< S, A, P >$ are as follows:

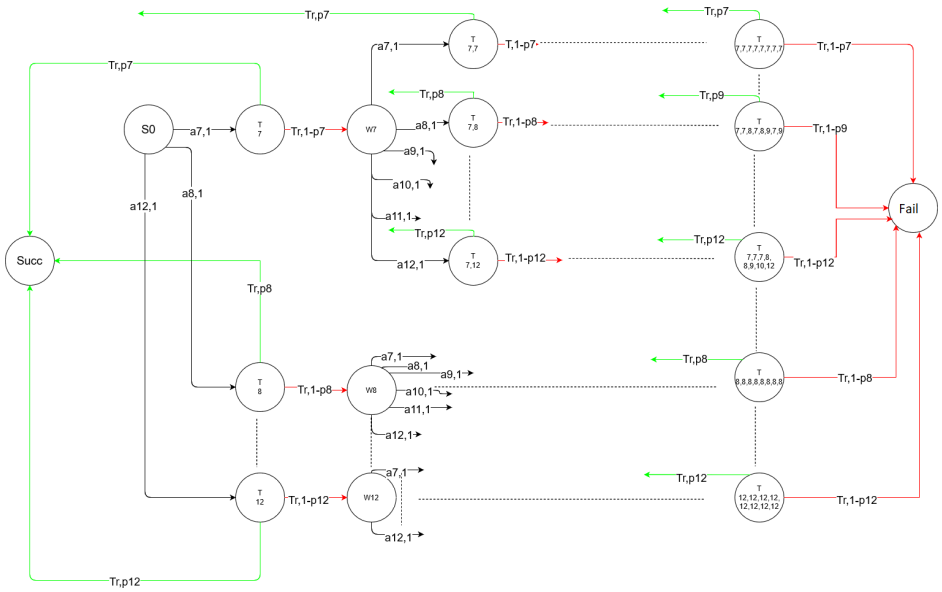


Figure 3: MDP model of node

In this model, the node has a set of states S, which includes:

1. The initial state $S0$. This state is the beginning of the packet transmission.

2. The Transmit subset T. The states $s \in T$ represent that the node is transmitting. The state of the k-th attempt is described as $T/\{sf_i\}_k$ where $\{sf_i\}_k$ is a k-elements sequence and $1 \leq k \leq 8$. The elements sf_i represent the SF chosen at the i-th transmission. For example, $T/\{8, 10, 12\}$ means the node fails in the two previous attempts using SF8 and SF10 and is transmitting with SF12.

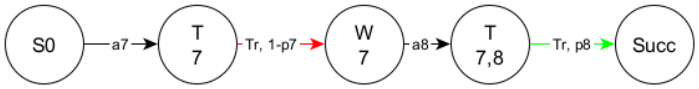
3. The Waiting subset W. The state $s \in W$ is described as $W/\{sf_i\}_k$, meaning that the node waits after the k-th transmission attempt fails.

$\{sf_i\}_k$ is a k-elements sequence of previous attempts parameter where $1 \leq k \leq 7$. For example, $W/\{9, 11\}$ means the node fails in the two previous attempts using SF9 and SF11 waits for the third attempt.

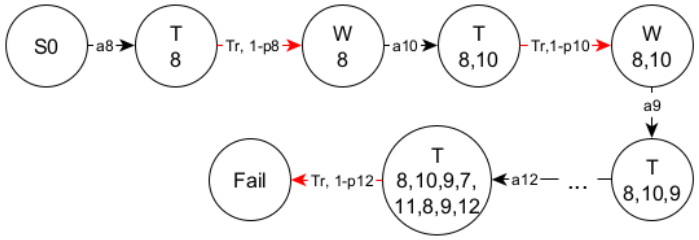
4. Two final states $\{Succ, Fail\}$. For the states $s \in T$, the node goes into state $Succ$ if the transmission succeeds. Otherwise, if a node fails its last attempt (i.e., $s = T/\{sf_i\}_8$), the packet transmission is failed, and the node goes into state $Fail$.

The action set A is defined as $A = \{Tr, a7, a8, ..., a12\}$. The action Tr represents a node's transmission action in state $s \in T$. The actions a_{sf} denote that a node in waiting state $s \in W$ chooses sf as its spreading factor for the next transmission.

In [9], the transition probabilities are divided into two cases. The first case is when $s \in W$, and the maximum retransmission attempt has not been exceeded. In this case, the node transmits with a probability $\mathbb{P}(s, a, s') = 1$. Another case is when $s \in T$, the node has a success probability $\mathbb{P}(s, a, s') = p_i$ of transmission with chosen sf i. An estimation of the success probability is also given in [9].



(a) Success transmission



(b) Failed transmission

Figure 4: Transmission examples of the MDP model

Based on the definition above, the model's paths represent a node's transmission procedures. Figure 4 show examples of a successful and failed transmission. In Figure 4(a), the node starts at state $S0$ and takes SF7 (action

$a7$) for the first transmission. It then goes to a state of transmission ($T/\{7\}$) and takes the transmission action Tr. However, the node does not receive the ACK with the failure probability $1 - p_7$ and goes to a waiting state W_7. Subsequently, the node selects SF8 (action $a8$), prepares the second attempt (state $T/\{7, 8\}$), and transmits the packet (action Tr). This time, the node receives the ACK with probability p_8 and goes to the final state $Succ$.

Figure 4(b) shows an example path of a failed transmission. Like the previous example, the node takes an SF and transmits the packet at each attempt. However, it does not receive the ACK during all 8 attempts. Therefore, after 8 transmission attempts with SF8, SF10, SF9, SF7, SF11, SF8, SF9, and SF12, the nodes goes to the final state $Fail$ with the failure probability $1 - p_{12}$.

Moreover, a reward tensor $R \in \mathbb{R}^{S \times A \times S}$ is proposed in [9]. The elements in R are denoted as $R(s, a, s')$, representing the expected reward of transitioning from state $s \in S$ to state $s' \in S$ with action $a \in A$. Using these rewards and the defined tuple, the MDP model can be solved by maximizing rewards using the value iteration method. The results are implemented in the algorithms for the initial strategy prediction of STEPS. The STEPS is a dynamic SF allocation approach enabling nodes to adjust transmission parameters based on a score table of SFs to enhance transmission performance proposed in [33]. The performance of the STEPS algorithm can vary depending on the initialization of the score table. By implementing the normalized SF distribution given by the optimal path of the proposed MDP as the initial score table for the STEPS algorithm, analytical results in [9] demonstrate that the MDP prediction method reduces transmission energy consumption while maintaining high performance compared to the legacy STEPS algorithm.

However, the MDP is based on a single node, not considering collision probability or nodes' interactions. The proposed estimation requires distributions of nodes' positions and parameters. A probabilistic timed automata (PTA) model is utilized to investigate nodes' interactions during transmission further, integrating the concept of time. The PTA is an automaton that can represent time passing. Several PTA models are proposed in this study. The syntax, semantics, and proposed models are detailed in the following section.

5. PROBABILISTIC TIMED AUTOMATA MODELS

This section focuses on modeling LoRaWAN transmission using probabilistic timed automata (PTA) models. The mathematical definition of the model, including the syntax and semantics, is first introduced. Subsequently, a model for unconfirmed transmission in LoRaWAN networks proposed in our previous work [10] is presented. Additionally, as the main contribution of this study, a comprehensive PTA model of the bidirectional transmission in LoRaWAN networks is proposed.

5.1. Syntax and Semantics

The PTA model is proposed in [11]. It is a timed automaton [45] that represents a system with time passing, non-determinism actions, and probabilistic transitions of states. A PTA can be described as a tuple $\mathbf{PTA} =< L, \bar{l}, \mathcal{X}, \Sigma, inv, prob >$ [46] where:

- L is the finite set of locations;
- $\bar{l}$ is the initial location;
- $\mathcal{X}$ is the finite set of clocks. A clock $x \in \mathcal{X}$ is a variable on the interval $t \subseteq \mathbb{T}$, where $\mathbb{T} \in \{\mathbb{R}^*, \mathbb{N}^*\}$. The set $\mathcal{C}(\mathcal{X})$ is defined as the constraints of the clocks.
- Σ is the finite set of events. The subset $\Sigma_u \subset \Sigma$ represents the ***urgent events*** where the transition happens immediately once it is possible.
- $inv : L \to \mathcal{C}(\mathcal{X})$ is the invariant condition function. It represents the time constraints that the **PTA** can stay at the location $l \in L$;
- $prob \subseteq L \times \mathcal{C}(\mathcal{X}) \times \Sigma \times \mathbf{Dist}(2^{\mathcal{X}} \times L)$ is finite set of the probabilistic transition relation, where $\mathbf{Dist}(A)$ represents the set of distributions over countable subsets of A.

The states of a PTA can be described as a pair (l, v) where $l \in L$, $v \in \mathcal{X}$, and v satisfies $inv(l)$. According to the definition of the initial location, the initial state of the PTA model is $(\bar{l}, v)$. Therefore, if the clocks are set to 0 at the beginning, v becomes a zero vector and the initial state of PTA will be $(\bar{l}, \mathbf{0})$. At each state, the PTA model has two types of transitions:

- ***Delay transitions***: the PTA lets time pass and stay at the current state while v satisfying $inv(l)$
- ***Event transitions***: the PTA made a transition according to $(l, g, \sigma, p) \in prob$, where: l is the current location, g clock constraint satisfied by current clock v, σ is the transition event, and the $p(X, l')$ is the probability of resetting all clocks in X to 0 and moving to the location l'.

In order to better illustrate the model, Daws and Yovine [47] proposed a notion of urgency on the locations. At urgent locations, only ***Event transitions*** are available, and ***Delay transitions*** cannot occur in these locations. Segala and Lynch [48] and Duflot et al. [43] show that the parallel composition of two PTAs is still a PTA and the connections among sub-PTAs can be established through the globally synchronized transitions. This allows the network to be divided into different elements with different clocks. Therefore, to clarify the model description and simplify the model construction, we have split the whole network model into sub-PTA models corresponding to different network components and comprehensively detailed each sub-model structure, enabling readers to replicate the evaluation process more efficiently.

Table 2: DEFINITION OF TIME CONSTANTS AND PROBABILITIES

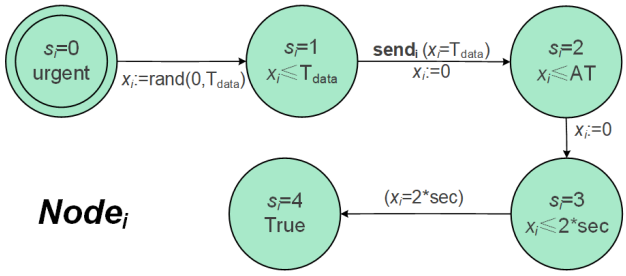
Time constants	
T_{data}	Maximum preparation time defined by user
AT	Packet receiving time
T_c	Time that the receiver needs to lock the message
sec	1 second
DC_i	Uplink duty-cycle limitation of node i
DL_i_DC	Downlink duty-cycle limitation of window $i \in [\![1, 2]\!]$
Probabilities	
P_{ULi}	The transmission of $node_i$ is not lost
P_{CEi}	The receiving power of $node_i$ is 6dBm greater
P_{DLij}	The downlink ACK to $node_i$ is not lost in window j

With the definition and the syntax above, several PTA models of the LoRaWAN protocol on the MAC layer are built. In our previous work [10], a PTA model for unconfirmed transmission in LoRaWAN on the MAC layer is proposed. As an extension, this study gives a PTA model for complete

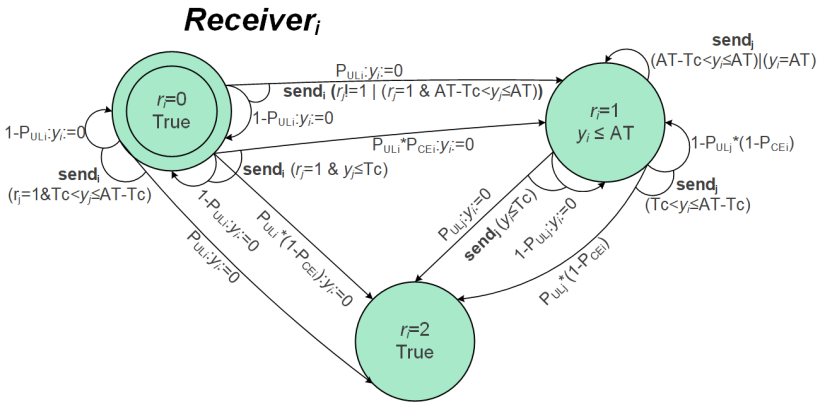
bidirectional transmission in LoRaWAN. Table 2 summarizes the notations used in this study. The network configuration, the modeling assumptions, and details of the PTA models are given in the following part.

5.2. Unconfirmed Transmission Modeling

Using the definition and the syntax above, a PTA model is constructed in [10] to represent the unconfirmed transmission procedure of two nodes in a LoRaWAN network with a single gateway based on the procedure given by Section 2.1 and Figure 2(a). The model consists of four sub-PTAs: two node PTAs and two receiver PTAs, each corresponding to a node and its receiver at the gateway. The nodes are assumed to use the same SF and transmission frequency.



(a) PTA model of ***node$_i$*** in unconfirmed transmission



(b) PTA model of ***receiver$_i$*** in unconfirmed transmission

Figure 5: PTA model of unconfirmed transmission

Figure 5 gives the sub-PTAs. The node's PTA model is shown in Figure 5(a). For PTA of the ***Node$_i$***, it is composed of a clock x_i, and 5 locations

$s_i \in [\![0, 4]\!]$ where:

$s_i = 0$: is the initial location. It is an urgent location where the node immediately sets the transmission data. The node sets the clock with $x_i := rand(0, T_{data})$ randomly and goes to location 1.

$s_i = 1$ is the preparation location. Node prepares its data during the preparation time. When the the clock arrives the maximum time $x_i = T_{data}$, the node takes the urgent event $\mathbf{send_i}$, and takes the transition to location 2.

$s_i = 2$ is the transmission location. The node transmits the packet during the airtime AT. When $x_i = AT$, the node takes the transition to location 3.

$s_i = 3$ is the downlink window location. In unconfirmed transmission scenario, the node does not listen to downlink message. Therefore, after a delay for 2 seconds ($2 * sec$), the node takes the transition to location 4.

$s_i = 4$ is the final location. It is the end of the transmission.

Figure 5(b) demonstrates a receiver PTA model. This model consists of 3 locations $r_i \in [\![0, 2]\!]$ and a clock variable y_i. The locations and transitions are given as follows:

$r_i = 0$ is the initial location. $\boldsymbol{Receiver_i}$ stays at this location until the event $\mathbf{send_i}$ happens. After $\boldsymbol{node_i}$ transmits, $\boldsymbol{receiver_i}$ synchronizes with the other receiver to check if the gateway can process the message. Using the collision model in Section 2.4, the transitions at this location are summarized as follows:

1. The gateway is available or the message is late enough (event $\mathbf{send_i}(r_j! = 1 \| (r_j = 1 \& AT - T_c < y_j \leq AT))$). The receiver takes the transition to location 1 with a probability P_{ULi} or stays at location 0 with a probability $1 - P_{ULi}$.

2. The gateway has locked a message from the other node (event $\mathbf{send_i}(r_j = 1 \& T_c < y_j \leq AT - T_c)$). The receiver stops receiving and stays at location 0 with probability $1 - P_{ULi}$ or goes to location 2 with probability P_{ULi}.

3. The gateway has not yet locked the message from the other node (event $\mathbf{send_i}(r_j = 1 \& y_j \leq T_c)$). The receiver takes the transition to location 1 of receiving with probability $P_{ULi}P_{CEi}$ if $\boldsymbol{node_i}$ overlaps the message from the other node, or to location 2 with probability $P_{ULi} * (1 - P_{CEi})$ if packet from $\boldsymbol{node_i}$ is not powerful enough, or stays at location 0 with probability $1 - P_{ULi}$ if the packet loses.

$r_i = 1$ is the receive location. $\boldsymbol{Receiver_i}$ keeps receiving the message during the airtime AT. If the other node $\boldsymbol{node_j}$ transmits, the receiver judges whether the receiving packet is collided using the collision model mentioned in Section 2.4. This time, the current packet becomes the earlier one (three events $\mathbf{send_j}$). If the packet collides, the receiver transits to location 2. Otherwise, it stays at location 1 when $y_i = AT$.

$r_i = 2$ is the collided location. The receiver stays at this location when the packet is collided.

In [10], several probabilistic properties are verified utilizing the model above. However, due to the lack of downlink transmission, only the uplink probabilities were verified in [10], such as the Packet Delivery Ratio (PDR), and the collision probability of one single transmission. Facing this limitation, this study addresses the issue by introducing a comprehensive PTA model. This novel model incorporates downlink ACK transmission from the gateway while also modeling the downlink channels. Moreover, the retransmission mechanism and the bidirectional duty-cycle limitation are implemented into the model to enhance reliability and better emulate the real-world LoRaWAN protocol. The novel model is detailed in the following section.

5.3. Bidirectional Transmission Modeling

Based on the confirmed transmission procedure given in Section 2.1 and Figure 2(b), this study proposes a comprehensive PTA model to formally evaluate the confirmed transmission in LoRaWAN networks. The LoRaWAN network is divided into six sub-PTAs. These include two PTAs for the nodes, two PTAs for the downlink windows, and two PTAs for the receivers at the gateway. The nodes are assumed to use the same SF and transmission frequency for simplicity. The schedule and the details of the models are given in the following parts.

5.3.1. Node Modeling

The node's PTA model is shown in Figure 6. For PTA of the $\boldsymbol{node_i}$, there are 6 abstracted locations $s_i \in [\![0, 5]\!]$, a clock x_i, a retransmission counter $retx_i$, and a local variable res_i that indicate the states of the transmission, where:

- $res_i = 0$ represents that the node finishes its transmission and waits for the ACK in the first receiving window.

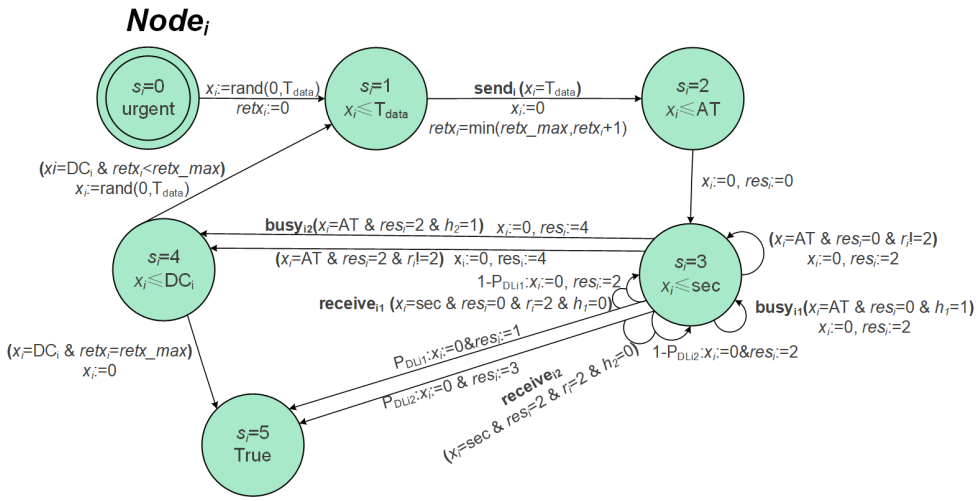


Figure 6: PTA model of $\boldsymbol{node_i}$ in confirmed transmission

- $res_i = 1$ represents that the node receives the ACK in the first window and finishes the transmission.
- $res_i = 2$ represents that the node does not receive the ACK in the first receiving window and listens in the second window.
- $res_i = 3$ represents that the node receives the ACK in the second window, and the transmission is successful.
- $res_i = 4$ represents that the node misses the ACK in both windows, and the transmission failed.

For the location s_i and the transitions, the schedule of the PTA is given as follows:

$s_i = 0$ is the initial location. It is an urgent location where the node immediately sets the transmission data, resets the retransmission counter $retx_i$, and goes to location 1. Since the transmission is in Aloha, the nodes sets randomly the preparation time with $x_i := rand(0, T_{data})$.

$s_i = 1$ is the preparation location. Node prepares its data in this location during the preparation time. When the node arrives the sending time $x_i = T_{data}$, it takes the urgent event $\mathbf{send_i}$, add the $retx_i$ by 1, and goes to the transmission location (location 2).

$s_i = 2$ is the transmission location. The node transmits the packet during the airtime AT. When $x_i = AT$, the transmission finishes. Node sets res_i to 0 and goes to the downlink window location (location 3).

$s_i = 3$ is the downlink window location. Node waits for 1 second (sec) and opens the first downlink window. If the node receives the ACK, it takes the urgent events $\mathbf{receive_{i1}}$ and goes to the final location (location 5) with a probability P_{DLi1}. Otherwise, if the downlink transmission is lost $(1 - P_{DLi1})$, or the first channel is busy ($\mathbf{busy_{i1}}, \mathbf{h1} = \mathbf{1}$), or the gateway is unable to answer ($ri! = 2$), the node sets res_i to 2 and restart the time counter to wait for the second receive window. When node finishes waiting another second ($res_i = 2 \& x_i = sec$), node opens the second receiving window. If an ACK is received, the node takes the urgent events $\mathbf{receive_{i2}}$ and transits to the final location (location 5) with a probability P_{DLi2}. Otherwise, if the downlink transmission is lost $(1 - P_{DLi2})$, or the second channel is busy ($\mathbf{busy_{i2}}, \mathbf{h2} = \mathbf{1}$), or the gateway is unable to answer ($ri! = 2$), the transmission is fail. The node sets res_i to 4 and transits to the duty-cycle location (location 4).

$s_i = 4$ is the duty-cycle location. If the node reaches its retransmission limits ($retx_i = retx_{lim}$), it takes the urgent event to the final location (location 5) and ends the procedure. Otherwise, the node stays at this location until $x_i = DC_i$, where DC_i is the uplink duty-cycle limit for the node. When the duty-cycle is over, the node resets the data $x_i := rand(0, T_{data})$ and goes to the preparation location (location 1) to wait for the retransmission of the packet.

$s_i = 5$ is the final location. It represents that the node finishes the packet transmission.

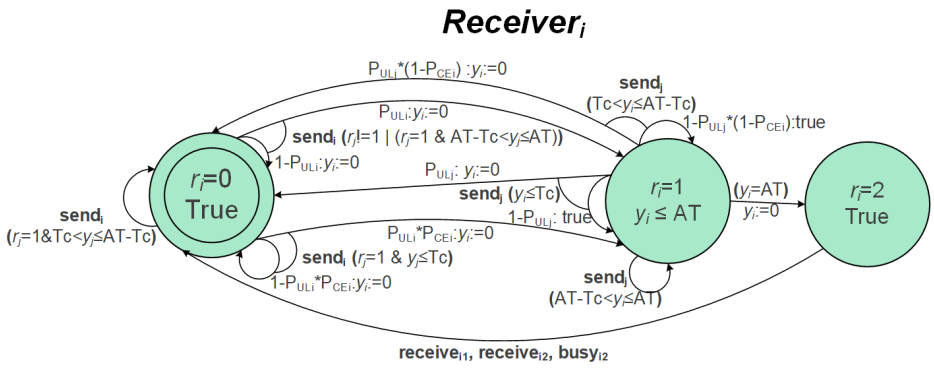
After $\boldsymbol{node_i}$ transmits (event $\mathbf{send_i}$), the gateway's corresponding receiver $\boldsymbol{receiver_i}$ starts receiving and checking the collision. The PTA model of the receiver is given in the following subsection.

5.3.2. Receiver Modeling

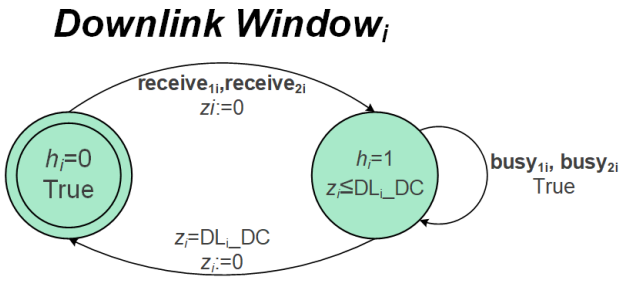
The PTA model of the receiver at the gateway is given in Figure 7(a). For each $\boldsymbol{receiver_i}$ of $node_i$, there are 3 locations $r_i \in [\![0, 2]\!]$ and 1 clock y_i.

Based on the collision model in Section 2.4, the schedule of the PTA is given as follows:

$r_i = 0$ is the initial location. $\boldsymbol{Receiver_i}$ stays at this location until the event $\mathbf{send_i}$ happens. After $\boldsymbol{node_i}$ transmits, $\boldsymbol{receiver_i}$ synchronizes with the other receiver to check if the gateway can process the message based on the collision model in Section 2.4:



(a) PTA model of ***receiver$_i$*** in bidirectional transmission



(b) PTA model of ***downlink window$_i$*** in bidirectional transmission

Figure 7: PTA of receivers and downlink windows in bidirectional transmission

- If the gateway is available, which means the other receiver is not receiving, or the message is late enough to avoid the interference (event $\mathbf{send_i}(r_j! = 1 \| (r_j = 1 \& AT - T_c < y_j \leq AT))$), then the receiver starts receiving the message and takes the transition to the receiving location (location 1) with a probability P_{ULi} or stay at location 0 with a probability $1 - P_{ULi}$.

- If the gateway has already locked a message from another node, the receiver stop receiving and stay at location 0
(event $\mathbf{send_i}(r_j = 1 \& T_c < y_j \leq AT - T_c)$).

- When the gateway has not locked the message from another node yet, the receiver has a probability $P_{ULi}P_{CEi}$ to overlap the message from the other node and goes to location 1 to start receiving. Otherwise,

receiver stays at location 0 (event $\mathbf{send_i}(r_j = 1 \& y_j \leq T_c)$)

$r_i = 1$ is the receiving location. $\boldsymbol{Receiver_i}$ keeps receiving the message during the airtime AT. If the other node sends a message, the receiver judges whether the current packet is collided as in location 0 using the collision model above (three events $\mathbf{send_j}$). If the packet does not collide, the receiver goes to the downlink location (location 2) at $\mathbf{y_i = AT}$ and try to send an ACK to $\boldsymbol{node_i}$.

$r_i = 2$ is the downlink location. The receiver stays at this location until one of the ACK events for $\boldsymbol{node_i}$ happens. When the ACK for $\boldsymbol{node_i}$ is sent in the downlink window j (events $\mathbf{receive_{ij}}$), or both two downlink windows are occupied (event $\mathbf{busy_{i2}}$), the receiver stops the downlink and goes back to location 0 for the next receive.

5.3.3. Downlink Window Modeling

Figure 7(b) shows the PTAs of downlink windows. Each window has 2 location $h_i \in [\![0, 1]\!]$ and 1 clock z_i, where:

$h_i = 0$ is the idle location. The downlink window stays at this location until the gateway starts transmitting ACK to $\boldsymbol{node_j}$ and goes to the occupied location (location 1) (events $\mathbf{receive_{ji}}$).

$h_i = 1$ is the occupied location. The downlink window transmits an ACK to nodes and keeps being occupied. If the gateway tries to transmit an ACK at this location, the downlink window refuses the transmission and stays until the downlink duty-cycle (DL_i_DC) is reached (events $\mathbf{busy_{ji}}$).

With the PTA models above, the transmission procedure of the PTA model can be presented by the interactions of the sub-PTA models. In the initial state, both nodes stay at $s_i = 0, res_i = 0$, the receivers stay at $r_i = 0$, and the downlink windows stay at $h_i = 0$. Then, nodes go to $s_i = 1$ and prepare the packet. Once the $node_i$ finishes the preparation, it sends the packet $send_i$ and goes to $s_i = 2$. At the same time, receivers judge whether the packet is collided or lost based on the probability of RSSI and the collision model given in Section 2. Based on the result, receivers decide to reset the state $r_i = 0$ or continue the reception $r_i = 1$. When the reception finishes, receivers try to activate the downlink ACK in one of the downlink windows if it is available $h_i = 0$. During this time, the node stays at $s_i = 3$ and opens two receiving windows to listen to the ACK message. If the ACK is received or the maximum retransmission attempt is exceeded, the node

finishes the procedure ($s_i = 5$). Otherwise, the node tries to retransmit after the duty-cycle limitation ($s_i = 4$).

Based on the bidirectional PTA model, several properties are verified. The properties and the verification descriptions are given in the following section.

6. PROPERTIES AND VERIFICATION

With the probabilistic model checker "PRISM", several performance of the LoRaWAN protocol are evaluated quantitatively and qualitatively in this section. The properties and the verification are detailed in the following subsection.

6.1. Properties

The model properties describe the analytical performance of a system or a protocol. With certain probabilistic temporal logic, such as the Probabilistic Computation Tree Logic (PCTL) [49], PRISM can verify properties of non-deterministic and probabilistic models. The most important type of property is the probabilistic property, which enables reasoning about the probability of event occurrence or the reachability of certain system states. In this study, we first defined four probability properties as the sanity check of the model. These properties are characterized by fixed expected values, typically 0 or 1, representing behaviors that the model will certainly take or not take. By verifying these probabilistic properties, we can validate the sanity of the model, ensuring it behaves as expected and avoids adopting behaviors that contradict reality. The description of the sanity properties (SP) are as follows:

- **SP1** is the minimum probability that both nodes arrive at location $s = 5$, which means both nodes finish the procedure. This property should be one.
- **SP2** is the maximum probability that both receivers are receiving uplink packets ($r = 1$), and the time difference of two packets $\Delta t \leq AT - Tc$. This contradicts the collision model in LoRaWAN networks given in 2.4. Therefore, it should be zero.
- **SP3** is the maximum probability that the procedure finishes ($s_1 = s_2 = 5$) and at least one node does not transmit ($retx = 0$). This property contradicts the original intent of the modeling and should be zero.

- **SP4** is the maximum probability that the procedure finishes ($s_1 = s_2 = 5$) and at least one node is still listening to the ACK message ($res \in \{0, 2\}$). This property contradicts the LoRaWAN schedule and should be zero.

For the network performance, we calculate two probability properties, referred to as **PP1** and **PP2**, under different conditions. These properties allow us to quantify and assess specific aspects of the protocol's behavior. The description of the two properties are as follows:

- **PP1** is the maximum probability that the packet is eventually transmitted successfully ($P[s = 5 \ \& \ res \in \{1, 3\}]$).
- **PP2** is the maximum probability that both nodes successfully transmit the packet in K transmissions in sum ($P[s_1 = s_2 = 5 \ \& \ res_1, res_2 \in \{1, 3\} \ \& \ retx_1 + retx_2 \leq K]$).

Additionally, after associating the events and the locations with certain rewards, the reward properties are investigated by calculating the expected values in this study. The reward properties are defined as follows:

- **RP1** is the minimum expected energy consumption at the end of the transmission ($R_{energy}[s = 5]$).
- **RP1**$_{succ}$ is the minimum expected energy consumption for 1 success packet ($R_{energy}[s = 5 \ \& \ res \in \{1, 3\}]$). According to the documentation of PRISM [50], since the reachability probability of successful transmission (**PP1**) is less than 1, this reward is marked as infinity. However, it can be manually calculated by **RP1****PP1**.
- **RP2** is the minimum expected transmissions the node needs to take for a packet ($R_{transmission}[s = 5]$).
- **RP2**$_{succ}$ is the minimum expected transmissions the node needs to take for a successful transmission (($R_{transmission}[s = 5 \ \& \ res \in \{1, 3\}]$). Similary to **RP1**$_{succ}$, this reward is also manually calculated by **RP2****PP1**.

6.2. Model Building in PRISM

6.2.1. Assumptions and Configurations

In our previous work [9], the calculation of P_{CE}, P_{DL}, and P_{UL} based on distance and other network parameters are given. In this study, we fix the $\boldsymbol{node_2}$ at a distance of 1000m from the gateway. In order to verify the collision model in bidirectional transmission scenarios, both nodes use the highest SF (SF12) to get the lowest packet loss probability. The maximum data preparation time is set to the airtime of one transmission. The path loss model, bandwidth, coding rate, and transmission power are the same values as in [9]. The receiver sensitivities are the same as [51]. The duty-cycle of the uplink and the first downlink window is 1%, and the duty-cycle of the second downlink window is 10%.

Additionally, both nodes transmit in the same packet sizes. In this study, we verified two scenarios with packet sizes of 5 bytes and 10 bytes. Since LoRaWAN is a communication technology that aims to provide low-rate wide-area networks, most network nodes are low-consumption devices that transmit small packets. For example, the sensors report their detected values periodically. These packet sizes are large enough to send 1 or 2 values in most data types (int, float, char...), which is consistent with most of the application scenarios of LoRaWAN networks.

6.2.2. Representation of PTA in PRISM

In [10], the PTA model was initially implemented and analyzed using the PTA module [52] within the PRISM. However, due to the implementation constraints, the default PTA computation engine (**stochastic games**) cannot verify reward properties. Using the **digital clocks** engine allows the calculation of reward properties [53]. This engine first performs a discretization and a language-level model translation to the problem to one of model checking over a finite-state MDP. Subsequently, the engine computes reward properties using the temporary MDP module [54]. The translation adds a clock variable to all locations in the model. This variable is assigned the global maximum value as an upper bound in locations without time constraints. However, in LoRaWAN, due to duty-cycle mechanisms, the global maximum value of the time variable is very large (100 times the transmission time), leading to a problem of state space explosion in the translated model. According to the PRISM documentation [54], a regular PC can calculate and verify models with 10^7 states.

Therefore, in this work, we manually adjusted the model based on the

Table 3: Size of models

Retx	Packet size (bytes)	Nb states	Nb transitions	Model building time (s)	PP1 verification time (s)	RP1 verification time (s)
1	5	26987	27239	14.077	0.298	19.949
	10	38017	38385	18.406	0.486	21.535
2	5	58515	60105	34.693	42.277	25.628
	10	81371	83759	68.367	54.577	31.479
3	5	90043	92971	71.524	90.707	31.635
	10	124725	129133	111.713	133.413	35.918
4	5	121571	125837	107.071	114.177	35.752
	10	168079	174507	178.207	224.753	46.621
5	5	153099	158703	164.611	182.957	39.982
	10	211433	219881	278.684	321.603	53.7
6	5	184627	191569	208.009	251.478	49.885
	10	254787	265255	399.521	498.999	65.786
7	5	216155	224435	298.13	346.083	58.574
	10	298141	310629	475.149	597.095	81.995
8	5	247683	257301	345.892	464.795	64.167
	10	341495	356003	545.73	675.791	97.61

temporary file obtained from the translation using the **digital clocks** engine to address this issue. We removed the time variables and timing transitions for locations without time constraints. The hybrid computation engine that combines symbolic and explicit-state data structures in PRISM is also utilized for model solving. This strategic selection empowers us to calculate and evaluate the network performance based on the reward properties.

Moreover, a time abstraction approach is adopted to discretize the clocks in the novel model. By selecting a granularity corresponding to the smallest period in the model (i.e., the locking time of a packet T_c), the representation effectively reduces the model's size so PRISM can compile the model.

Table 3 shows the model size and computation cost with different retransmission times after the abstraction. After validating the absence of deadlock using the **the simulator tool** in PRISM [55], the sanity properties of the model are verified. The results are given in Table 4. It can be seen that all the results are as expected, which means the model behaves as expected and avoids adopting behaviors that contradict reality. For the performance properties, the verification results are given in the following part.

Table 4: Sanity properties

Retx	Packet size (bytes)	**SP1** (should be 1)	**SP2** (should be 0)	**SP3** (should be 0)	**SP4** (should be 0)
1	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
2	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
3	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
4	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
5	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
6	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
7	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0
8	5	1.0	0.0	0.0	0.0
	10	1.0	0.0	0.0	0.0

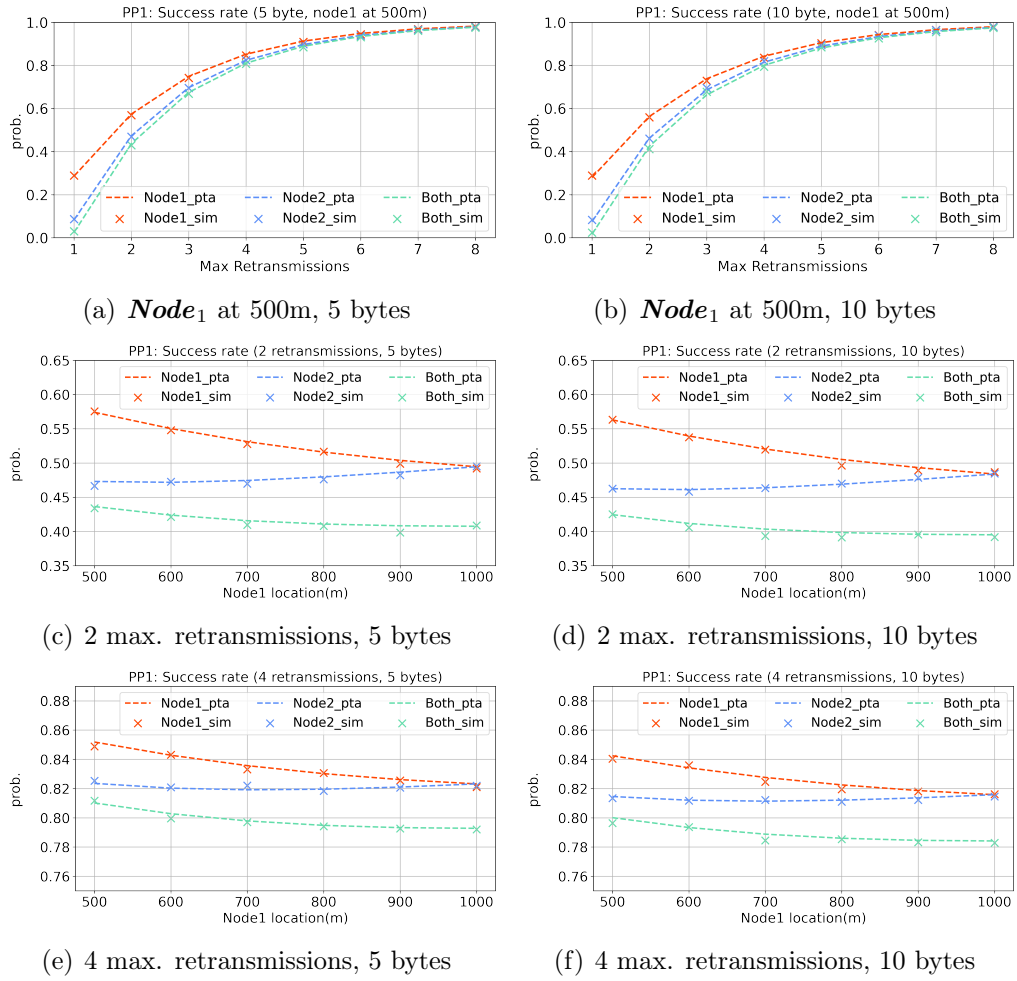
6.3. Verification Results

6.3.1. Probability Properties

To demonstrate the validity of the proposed model, cross-validation is also conducted using the simulator MELoNS. MELoNS is a modular and extendable simulation tool for LoRaWAN networks proposed in [32]. The half-duplex gateway, the complete procedure of downlink transmission, and the MAC commands are implemented in MELoNS. Moreover, the components in the network are independent in MELoNS, giving the simulation a higher degree of freedom. By defining the same environment parameters and node behaviors, we simulated in 20000 packets and verified properties defined above using the results given by MELoNS.

The verification results of **PP1** are shown in Figure 8. We first set the $\boldsymbol{node_1}$ at a distance of 500m from the gateway and verified the success rate with different max retransmission numbers. Figure 8(a) and 8(b) illustrate the verification results. It can be seen that as the nodes are afforded additional opportunities for retransmission, the packet success rate exhibits a commensurate increase. Since the $\boldsymbol{node_1}$ is closer to the gateway, it is more likely to get a larger $RSSI$ than the $\boldsymbol{node_2}$. Therefore, $\boldsymbol{node_1}$ gets a higher likelihood of survival under collision, bringing a higher **PP1** than the $\boldsymbol{node_2}$. These observations are also proved by verification results given by Figure 8(c)

to 8(f). In these verifications, we fixed both nodes' maximum retransmission numbers (2 and 4) and varied the location of the $\boldsymbol{node_1}$. It can be seen that when the two nodes get closer, the interference probability increases, thus decreasing the total packet success rate.



(a) $\boldsymbol{Node_1}$ at 500m, 5 bytes

(b) $\boldsymbol{Node_1}$ at 500m, 10 bytes

(c) 2 max. retransmissions, 5 bytes

(d) 2 max. retransmissions, 10 bytes

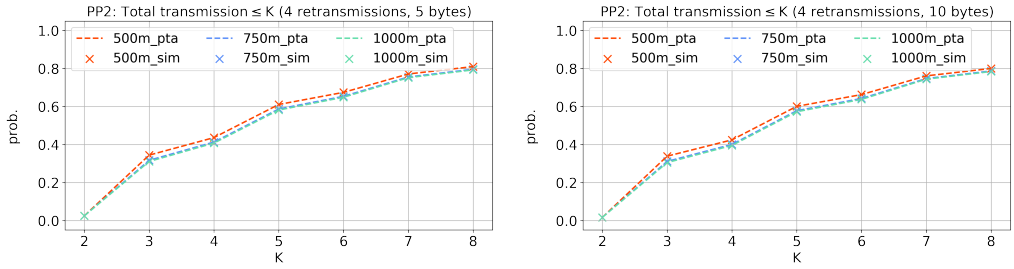
(e) 4 max. retransmissions, 5 bytes

(f) 4 max. retransmissions, 10 bytes

Figure 8: Verification results on packet success rate **PP1**

An interesting point observed from Figure 8(c) to 8(f) is that although the probability P_{CE2} increases when the two nodes get closer, the success rate of the $\boldsymbol{node_2}$ varies slightly. This is because when the $\boldsymbol{node_1}$ is further from the gateway, the success rate decreases, and the need for more packet transmissions brings a higher probability of collision to both nodes.

Additionally, all figures in Figure 8 demonstrate that the success rate of packet size of 10 bytes is lower than that of packet size of 5 bytes (about 1%).



(a) 4 max. retransmissions, 5 bytes (b) 4 max. retransmissions, 10 bytes

Figure 9: Verification results on **PP2**

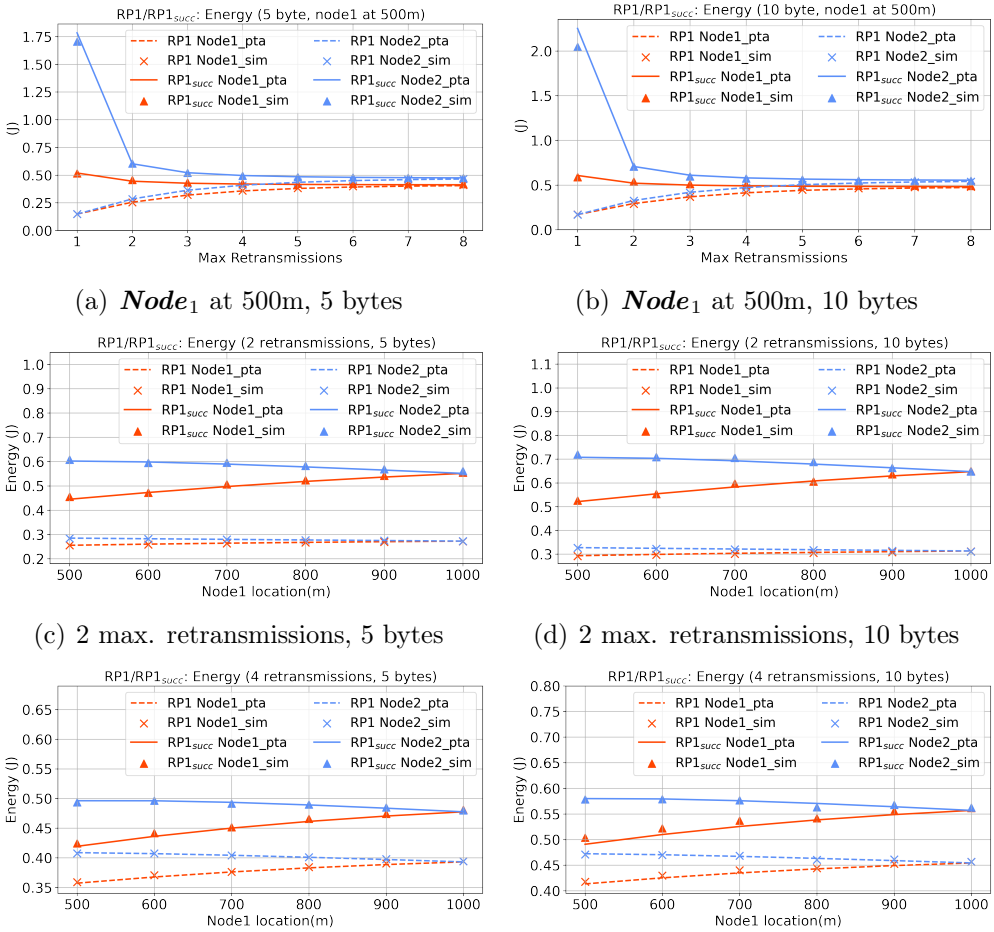
Figure 9 presents the verification results of **PP2**. In these verifications, we fixed the maximum retransmission number at 4 and varied the location of the $\boldsymbol{node_1}$. This property presents the success rate of the whole system within different transmission consumptions. Notably, when K is set to 8, the values align with that given in Figure 8(e) and 8(f), demonstrating the model's robustness and reliability.

6.3.2. Reward Properties

The reward properties are also verified in this study. Figure 10 illustrates the verification results regarding the expected energy consumption for one packet (**RP1**) and for one successful transmission ($\mathbf{RP1}_{succ}$). The verification scenarios for these properties are the same as that for the property **PP1**.)

In Figure 10(a) and 10(b), we initially positioned the $\boldsymbol{node_1}$ at a distance of 500m from the gateway. It can be observed that with more retransmission attempts, the expected energy consumption for one packet (**RP1**) increases. However, as the chances of retransmission rise, the packet success rate also increases. Consequently, the expected energy consumption for one successful packet ($\mathbf{RP1}_{succ}$) decreases. These two values converge when the success rate (**PP1**) is high enough. Nonetheless, given that the $\boldsymbol{node_1}$ is closer to the gateway compared to the $\boldsymbol{node_2}$, it has a higher probability of surviving

from collisions (P_{CE1}). Hence, its energy consumption consistently remains lower than the $\boldsymbol{node_2}$.



(a) $\boldsymbol{Node_1}$ at 500m, 5 bytes

(b) $\boldsymbol{Node_1}$ at 500m, 10 bytes

(c) 2 max. retransmissions, 5 bytes

(d) 2 max. retransmissions, 10 bytes

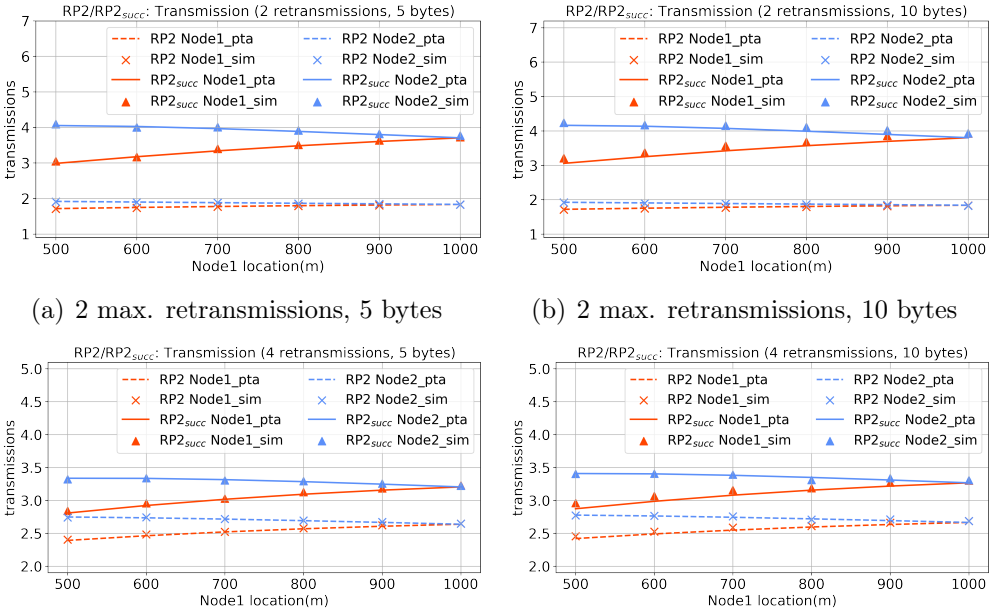
(e) 4 max. retransmissions, 5 bytes

(f) 4 max. retransmissions, 10 bytes

Figure 10: Verification results on expected energy consumption $\mathbf{RP1}/\mathbf{RP1}_{succ}$

Figure 10(c) to 10(f) show the impact of the node distance to the energy consumption. It can be seen that when the $\boldsymbol{node_1}$ is closer to the gateway, it has a higher probability of surviving under collision. Thus, the $\boldsymbol{node_1}$ spends less transmission to have a successful packet than the $\boldsymbol{node_2}$ ($\mathbf{RP1}_{succ}$). Therefore, the energy consumption ($\mathbf{RP1}$) of the $\boldsymbol{node_1}$ is less than the $\boldsymbol{node_2}$. When the two nodes are at the same distance, the P_{CE} becomes the same. Both nodes have the same expected transmission time and energy

consumption. Therefore, the curves of the two nodes intersect.



(a) 2 max. retransmissions, 5 bytes

(b) 2 max. retransmissions, 10 bytes

(c) 4 max. retransmissions, 5 bytes

(d) 4 max. retransmissions, 10 bytes

Figure 11: Verification results on expected energy consumption $\mathbf{RP2}/\mathbf{RP2}_{succ}$

Other verifications are conducted for the expected transmissions for one packet (**PP2**) and one successful transmission ($\mathbf{PP2}_{succ}$). These verifications are made under varied locations of the $\boldsymbol{node_1}$ and the same maximum retransmission numbers (2 and 4) of both nodes. Figure 11 shows the verification results. It can be seen that when the $\boldsymbol{node_1}$ is closer to the gateway, it has a higher probability of surviving under collision. Thus, the $\boldsymbol{node_1}$ spends less transmission to have a successful packet than the $\boldsymbol{node_2}$ ($\mathbf{PP2}_{succ}$). Therefore, the $\boldsymbol{node_1}$ finishes the transmission earlier than the $\boldsymbol{node_2}$ and transmits less (**PP2**). Moreover, although the expected transmission numbers (**PP2**) in 2 maximum retransmissions are less than that in 4 maximum retransmission because of the limitation, the expected transmissions for one success packet ($\mathbf{PP2}_{succ}$) is higher. This is because when the retransmission limitation increases, nodes in the network have more chances to send the packet, and the packet success rate (**PP1**) increases.

Additionally, from pictures 1 to 3, the difference between all verification and simulation results is minimal. This shows that the formal performance

evaluation results based on the proposed probabilistic model are consistent with the statistics-based evaluation result, proving the reliability and rationality of our proposed model.

7. CONCLUSION

This study investigates performance evaluation through the probabilistic model checking in LoRaWAN on the MAC layer. A formal approach that covers the complete behavior of the models of the Class-A device to evaluate the LoRaWAN network is proposed. Probabilistic Timed Automata (PTA) models are built to model the characteristics of the MAC layer of LoRaWAN, such as the duty-cycle limits, the collision, and the bidirectional communication. The probabilistic and the rewards properties of the model are defined. The properties and performance are evaluated in different scenarios using the model checker PRISM. In the future, directions remain to be explored.

- Firstly, it is planned to investigate more properties. For example, to verify the properties of the gateway's availability by increasing the packet number.
- Secondly, the non-deterministic decision-making, such as the different choices of SFs, is also planned to be integrated into the model.
- Since the model is validated with the simulation results, cross-validation of the probabilistic checking results with a real LoRaWAN test-bed is also planned.

Acknowledgement

This research is supported by ASPIRE, the technology program management pillar of Abu Dhabi's Advanced Technology Research Council (ATRC), via the ASPIRE Visiting International Professorship program. Project IT5G ASPIRE/ZU R22036 EU2105

References

[1] LoRa Alliance Technical Committee, Lorawan® l2 1.0. 4 specification (ts001-1.0. 4), LoRa Alliance: Fremont, CA, USA (2020).

[2] LoRa Alliance Technical Committee, RP002-1.0.4 LoRaWAN Regional Parameters, Technical Report, Tech. rep. Version: RP002-1.0.4, 2022.

[3] L. G. Manzano, H. Boukabache, S. Danzeca, N. Heracleous, F. Murtas, D. Perrin, V. Pirc, A. R. Alfaro, A. Zimmaro, M. Silari, An iot lorawan network for environmental radiation monitoring, IEEE Transactions on Instrumentation and Measurement 70 (2021) 1–12. doi:10.1109/tim.2021.3089776.

[4] P. J. Basford, F. M. Bulot, M. Apetroaie-Cristea, S. J. Cox, S. J. Ossont, Lorawan for smart city iot deployments: A long term evaluation, Sensors 20 (2020) 648. doi:10.3390/s20030648.

[5] D. Asiain, D. Antolín, Lora-based traffic flow detection for smart-road, Sensors 21 (2021) 338. doi:10.3390/s21020338.

[6] R. Salazar-Cabrera, Á. Pachón de la Cruz, J. M. Madrid Molina, Proof of concept of an iot-based public vehicle tracking system, using lora (long range) and intelligent transportation system (its) services, Journal of Computer Networks and Communications 2019 (2019) 1–10. doi:10.1155/2019/9198157.

[7] M. Chen, J. B. Othman, L. Mokdad, Intelligent urban expressway managing architecture using lorawan and edge computing, in: GLOBECOM 2023-2023 IEEE Global Communications Conference, IEEE, 2023, pp. 758–763.

[8] LoRa alliance, Lora alliance lorawan and 5g infographic, https://resources.lora-alliance.org/infographic/lora-alliance-lorawan-and-5g-infographic, 2022 (accessed April, 2024).

[9] M. Chen, L. Mokdad, C. Charmois, J. Ben-Othman, J.-M. Fourneau, An mdp model-based initial strategy prediction method for lorawan, in: ICC 2022-IEEE International Conference on Communications, IEEE, 2022, pp. 4836–4841. doi:10.1109/ICC45855.2022.9838364.

[10] M. Chen, L. Mokdad, J. B. Othman, J.-M. Fourneau, Probabilistic model checking for unconfirmed transmission in lorawan on the mac layer, in: GLOBECOM 2023-2023 IEEE Global Communications Conference, IEEE, 2023, pp. 2457–2462.

[11] M. Kwiatkowska, G. Norman, R. Segala, J. Sproston, Automatic verification of real-time systems with discrete probability distributions, Theoretical Computer Science (1999). doi:10.1016/S0304-3975(01)00046-9.

[12] M. Kwiatkowska, G. Norman, D. Parker, Prism 4.0: Verification of probabilistic real-time systems, in: Computer Aided Verification: 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings 23, Springer, 2011, pp. 585–591.

[13] S. Böcker, C. Arendt, P. Jörke, C. Wietfeld, Lpwan in the context of 5g: Capability of lorawan to contribute to mmtc, in: 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), IEEE, 2019, pp. 737–742.

[14] Semtech Corporation, An in-depth look at lorawan class a devices, https://lora-developers.semtech.com/uploads/documents/files/LoRaWAN_Class_A_Devices_In_Depth_Downloadable.pdf, 2019 (accessed April, 2024).

[15] F. H. Khan, M. Portmann, Experimental evaluation of lorawan in ns-3, in: 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), IEEE, 2018, pp. 1–8. doi:10.1109/ATNAC.2018.8615313.

[16] Semtech, Sx1261/sx1262 datasheet, [Online]. Available: https://semtech.my.salesforce.com/sfc/p/#E0000000JelG/a/2R000000Un7F/yT.fKdAr9ZAo3cJLc4F2cBdUsMftpT2vsOICP7NmvMo, 2021 (accessed April, 2024).

[17] G. Callebaut, L. Van der Perre, Characterization of lora point-to-point path loss: Measurement campaigns and modeling considering censored data, IEEE Internet of Things Journal 7 (2019) 1910–1918. doi:10.1109/JIOT.2019.2953804.

[18] P. Jörke, S. Böcker, F. Liedmann, C. Wietfeld, Urban channel models for smart city iot-networks based on empirical measurements of lora-links at 433 and 868 mhz, in: 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), IEEE, 2017, pp. 1–6.

[19] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, M. Pettissalo, On the coverage of lpwans: range evaluation and channel attenuation model for lora technology, in: 2015 14th international conference on its telecommunications (itst), IEEE, 2015, pp. 55–59.

[20] Semtech Corporation, Sx1276/77/78/79–137 mhz to 1020 mhz low power long range transceiver, https://www.semtech.com/products/wireless-rf/lora-connect/sx1276#documentation, 2015 (accessed April, 2024).

[21] A. Rahmadhani, F. Kuipers, Understanding collisions in a lorawan, Surf Wiki (2017).

[22] H. Rajab, T. Cinkler, T. Bouguera, Iot scheduling for higher throughput and lower transmission power, Wireless Networks 27 (2021) 1701–1714. doi:10.1007/s11276-020-02307-1.

[23] Y. A. Al-Gumaei, N. Aslam, X. Chen, M. Raza, R. I. Ansari, Optimising power allocation in lorawan iot applications, IEEE Internet of Things Journal (2021). doi:10.1109/JIOT.2021.3098477.

[24] A. Farhad, D.-H. Kim, J.-Y. Pyun, R-arm: Retransmission-assisted resource management in lorawan for the internet of things, IEEE Internet of Things Journal (2021). doi:10.1109/JIOT.2021.3111167.

[25] A. Kaburaki, K. Adachi, O. Takyu, M. Ohta, T. Fujii, Autonomous decentralized traffic control using q-learning in lpwan, IEEE Access 9 (2021) 93651–93661. doi:10.1109/ACCESS.2021.3093421.

[26] D. Bankov, E. Khorov, A. Lyakhov, Mathematical model of lorawan channel access with capture effect, in: 2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC), IEEE, 2017, pp. 1–5. doi:10.1109/PIMRC.2017.8292748.

[27] L. E. Marquez, A. Osorio, M. Calle, J. C. Velez, A. Serrano, J. E. Candelo-Becerra, On the use of lorawan in smart cities: A study with blocking interference, IEEE Internet of Things Journal 7 (2019) 2806–2815. doi:10.1109/JIOT.2019.2962976.

[28] 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. doi:10.1145/2988287.2989163.

[29] F. Van den Abeele, J. Haxhibeqiri, I. Moerman, J. Hoebeke, Scalability analysis of large-scale lorawan networks in ns-3, IEEE Internet of Things Journal 4 (2017) 2186–2198. doi:10.1109/JIOT.2017.2768498.

[30] D. Croce, M. Gucciardo, I. Tinnirello, D. Garlisi, S. Mangione, Impact of spreading factor imperfect orthogonality in lora communications, in: International Tyrrhenian Workshop on Digital Communication, Springer, 2017, pp. 165–179. doi:10.1007/978-3-319-67639-5_13.

[31] M. Chen, L. Mokdad, J. Ben-Othman, J.-M. Fourneau, Mulane-a lightweight extendable agent-oriented lorawan simulator with gui, in: 2021 IEEE Symposium on Computers and Communications (ISCC), IEEE, 2021, pp. 1–6. doi:10.1109/ISCC53001.2021.9631494.

[32] M. Chen, L. Mokdad, J. B. Othman, Melons - a modular and extendable simulator for lorawan network, in: 2023 International Wireless Communications and Mobile Computing (IWCMC), IEEE, 2023, pp. 923–928. doi:10.1109/IWCMC58020.2023.10183205.

[33] M. Chen, L. Mokdad, J. Ben-Othman, J.-M. Fourneau, Steps-score table based evaluation and parameters surfing approach of lorawan, in: 2021 IEEE Global Communications Conference (GLOBECOM), IEEE, 2021, pp. 1–6. doi:10.1109/GLOBECOM46510.2021.9685783.

[34] M. Chen, L. Mokdad, J. Ben-Othman, J.-M. Fourneau, Dynamic parameter allocation with reinforcement learning for lorawan, IEEE Internet of Things Journal (2023). doi:10.1109/JIOT.2023.3239301.

[35] M. Chen, L. Mokdad, J. Ben-Othman, J.-M. Fourneau, Loraloft-a local outlier factor-based malicious nodes detection method on mac layer for lorawan, in: 2022 IEEE Global Communications Conference (GLOBECOM), IEEE, 2022, pp. 1–6. doi:10.1109/GLOBECOM48099.2022.10000852.

[36] M. Chen, L. Mokdad, J. Ben-Othman, Robustness and resilience of lorawan facing greedy behaviors on the mac layer, in: ICC 2023-IEEE International Conference on Communications, IEEE, 2023, pp. 1–6.

[37] M. Kwiatkowska, G. Norman, D. Parker, Probabilistic model checking and autonomy, Annual Review of Control, Robotics, and Autonomous Systems 5 (2022) 385–410. doi:10.1146/annurev-control-042820-010947.

[38] B. Bordel, R. Alcarria, T. Robles, M. S. Iglesias, Data authentication and anonymization in iot scenarios and future 5g networks using chaotic digital watermarking, IEEE Access (2021). doi:10.1109/ACCESS.2021.3055771.

[39] Z. Aslanyan, F. Nielson, D. Parker, Quantitative verification and synthesis of attack-defence scenarios, in: 2016 IEEE 29th Computer Security Foundations Symposium (CSF), 2016, pp. 105–119. doi:10.1109/CSF.2016.15.

[40] E. Shaikh, N. Mohammad, S. Muhammad, Model checking based unmanned aerial vehicle (uav) security analysis, in: 2020 International Conference on Communications, Signal Processing, and their Applications (ICCSPA), 2021, pp. 1–6. doi:10.1109/ICCSPA49915.2021.9385754.

[41] M. Fruth, Probabilistic model checking of contention resolution in the ieee 802.15.4 low-rate wireless personal area network protocol, in: Second International Symposium on Leveraging Applications of Formal Methods, Verification and Validation (isola 2006), 2006, pp. 290–297. doi:10.1109/ISoLA.2006.34.

[42] M. Kwiatkowska, G. Norman, J. Sproston, Probabilistic model checking of the ieee 802.11 wireless local area network protocol, Lecture Notes in Computer Science (2002). doi:10.1007/3-540-45605-8_11.

[43] M. Duflot, L. Fribourg, T. Herault, R. Lassaigne, F. Magniette, S. Messika, S. Peyronnet, C. Picaronny, Probabilistic model checking of the csma/cd protocol using prism and apmc, Electronic Notes in Theoretical Computer Science (2005). doi:10.1016/j.entcs.2005.04.012.

[44] M. Kwiatkowska, G. Norman, J. Sproston, Probabilistic model checking of deadline properties in the ieee 1394 firewire root contention protocol, Formal Aspects of Computing 14 (2003) 295–318.

[45] R. Alur, D. L. Dill, A theory of timed automata, Theoretical Computer Science (1994). doi:10.1016/0304-3975(94)90010-8.

[46] M. Kwiatkowska, G. Norman, D. Parker, J. Sproston, Performance analysis of probabilistic timed automata using digital clocks, Formal Methods in System Design 29 (2006) 33–78.

[47] C. Daws, S. Yovine, Two examples of verification of multirate timed automata with kronos, in: Proceedings 16th IEEE Real-Time Systems Symposium, 1995, pp. 66–75. doi:10.1109/REAL.1995.495197.

[48] R. Segala, N. Lynch, Probabilistic simulations for probabilistic processes, Nordic Journal of Computing (1994). doi:10.1007/978-3-540-48654-1_35.

[49] H. Hansson, B. Jonsson, A logic for reasoning about time and reliability, Formal Aspects of Computing (1990). doi:10.1007/BF01211866.

[50] PRISM Model Checker, Reward-based properties, https://www.prismmodelchecker.org/manual/PropertySpecification/Reward-basedProperties, 2021 (accessed April, 2024).

[51] E. D. Ayele, C. Hakkenberg, J. P. Meijers, K. Zhang, N. Meratnia, P. J. Havinga, Performance analysis of lora radio for an indoor iot applications, in: 2017 International Conference on Internet of Things for the Global Community (IoTGC), IEEE, 2017, pp. 1–8. doi:10.1109/IoTGC.2017.8008973.

[52] PRISM Model Checker, Real-time models, https://www.prismmodelchecker.org/manual/ThePRISMLanguage/Real-timeModels, 2021 (accessed April, 2024).

[53] PRISM Model Checker, Properties of real-time models, https://www.prismmodelchecker.org/manual/PropertySpecification/Real-timeModels, 2021 (accessed April, 2024).

[54] PRISM Model Checker, Computation engines, https://www.prismmodelchecker.org/manual/ConfiguringPRISM/ComputationEngines, 2021 (accessed April, 2024).

[55] PRISM Model Checker, Debugging models with the simulator, https://www.prismmodelchecker.org/manual/RunningPRISM/DebuggingModelsWithTheSimulator, 2021 (accessed April, 2024).