A Scalable and Fast Model for Performance Analysis of IEEE 802.15.4 TSCH Networks

Hamideh Hajizadeh, Majid Nabi, Rasool Tavakoli, and Kees Goossens
Department of Electrical Engineering, Eindhoven University of Technology, the Netherlands
Email:{h.hajizadeh, m.nabi, r.tavakoli, k.g.w.goossens}@tue.nl

***Abstract*—The IEEE 802.15.4 Time-Slotted Channel Hopping (TSCH) protocol has received considerable attention in many industrial applications. However, analytical models for fast performance estimation of TSCH-based networks by considering the interaction between Medium Access Control (MAC) and Physical (PHY) layers is an open problem. In this paper, we propose a stochastic model for performance analysis of TSCH-based networks including dedicated and shared links with non-ideal wireless link properties. The proposed model is scalable and is able to evaluate the MAC performance of a large-scale network quickly. The developed model is verified by simulations and real-world experiments. The results confirm the accuracy of the proposed model for large-scale networks with orders of magnitude faster execution compared to the existing model in the literature. This confirms the speed and scalability of the model, which makes it a perfect tool for network design and optimization.**

***Index Terms*—IEEE 802.15.4, Time-slotted channel hopping, TSCH, Performance evaluation, Analytical model**

I. Introduction

The IEEE 802.15.4 [1] standard specifies an efficient protocol for Physical (PHY) and Medium Access Control (MAC) layers of low-power and low-cost Wireless Sensor Networks (WSNs). The Time-Slotted Channel Hopping (TSCH) mode of this standard uses time-slotted communications to manage the internal interference and channel hopping to reduce the impact of external interference and multi-path fading. Considering the improved reliability and predictable performance of TSCH, it has received strong attention especially for industrial applications running in harsh environments [2].

TSCH divides time into equal-size timeslots in which a node can transmit a data packet and receive its Acknowledgement packet (ACK). A number of timeslots are grouped together forming a slotframe that repeats over time. Two types of timeslots are defined for communications. Dedicated timeslots are exclusively assigned for contention-free communications of a link, while shared timeslots can be accessed by multiple links via a slotted CSMA-CA mechanism to avoid repeated collisions. The slotted CSMA-CA procedure starts with the transmission of a packet in the first shared timeslot after generation of the packet. If the communication fails, the transmitter backs off for a random number of shared timeslots in the range of $[0, 2^{BE}-1]$. BE is the Back-off Exponent and is initialized to $macMinBE$. With every failure of the retransmissions, BE is increased by one, up to a given maximum value, $macMaxBE$. The retransmission continues until an ACK is received, or the maximum allowed number of retransmissions ($maxR$) is reached.

In recent years, some studies have focused on performance analysis of TSCH-based networks [3]–[7]. One of the main problems that is not addressed properly in the existing literature is to extract a general and scalable performance model that covers both dedicated and shared timeslots, taking into account non-ideal wireless communication links. Such a performance model is essential to estimate performance of the network, and to optimize the target real-world network by configuring its parameters. The state-of-the-art analytical models are very computation intensive, and only consider special cases of the standard.

This paper presents an analytical model for performance evaluation of the TSCH MAC layer in terms of packet reception probability, packet latency, and energy consumption. This model considers both dedicated and shared timeslots for TSCH analysis. Moreover, physical layer communication success probabilities are considered as inputs for this model to address non-ideal channel conditions. Our model focuses on a cluster within a large network, in which several nodes send their data packets towards a common receiver. It considers the general case in which each transmitting node has its own dedicated timeslot in each slotframe, in addition to several shared timeslots to be used by all nodes for retransmission of the failed packets.

The proposed model is verified by extensive simulations in the Cooja [8] simulator, as well as real-world experiments. Simulation results show that our model provides more than 98% accuracy for the networks with more than 8 nodes. A comparison with the state-of-the-art model in [5], for a special TSCH schedule with only shared timeslots, shows that our model is faster by orders of magnitude, while the results of the two models differ by less than 1% for the networks with more than 4 nodes.

The paper is organized as follows. A review of the existing analytical models for TSCH performance evaluation is given in Section II. The proposed analytical model is presented in Section III. The simulation and experimental setups together with the achieved results are discussed in Section IV. Section V concludes.

II. Related Work

A mathematical model for performance analysis of the IEEE 802.15.4 non-beacon-enabled mode is provided in [9]. This model evaluates statistical distribution of the traffic generated by the nodes in a star network. With the assumptions that all nodes are synchronized and the medium is ideal, the probability of successful packet reception is extracted. Also, optimum packet size which maximizes the probability of successful packet reception is derived through the model. Following the same approach, [10] investigates the performance of the IEEE 802.15.4 beacon-enabled mode through a mathematical model. It considers both star and tree-based topologies.

In [11], [12], performance analysis is performed for the IEEE 802.15.4 beacon-enabled star topology networks. In [11], slotted CSMA/CA back-off process of IEEE 802.15.4 is modeled by a Markov chain. It considers analytical evaluation for both saturated and unsaturated periodic data traffic with the assumption of ideal channel conditions; throughput and energy consumption are predicted by the model. Taking into account the retransmission limit, the performance model is generalized in [12] by extending the Markov chain to three dimensions; reliability, delay, and energy consumption are modeled. Also, simple approximations of the performance metrics are presented since exact performance estimation requires heavy computations and may not be applicable for optimization purposes where performance must be estimated many times.

According to the standard, the TSCH CSMA/CA algorithm differs from the original CSMA/CA algorithm of IEEE 802.15.4 in a number of ways [13]. Despite this, [4], [6] follow an approach similar to [12], and use Markov chains to model the back-off behavior of a single node using TSCH CSMA/CA algorithm assuming ideal channels. The model presented in [5] provides a first performance model of a TSCH network, which considers entire network including all transmitting nodes. In [5] single node analysis is modeled by Markov chains. Based on the single-node analysis, the behavior of the transmitting nodes sending their data packets to a common receiver node is modeled by Discrete-Time Markov Chains (DTMC). To derive the DTMC, all possible transitions between transmission, back-off wait, success, and drop states in Markov model for all transmitting nodes are extracted through a brute-force method, and their probabilities are generated using the weak composition algorithm. Increasing the number of transmitting nodes and MAC configuration parameters considerably increase the possible states and transitions of transmitting nodes in the Markov model. Generating and evaluating all these transitions of transmitting nodes leads to high order of complexity and requires extremely heavy computation and execution time. Moreover, the described model in [5] considers TSCH networks using *only* shared timeslots with the assumption of ideal wireless channels, which not the case in many real-world TSCH networks and operating environments. In [7], Markov model is also used to study the behavior of TSCH nodes when using only shared timeslot to transmit a burst traffic assuming ideal channels.

Here, we take a stochastic approach to derive a fast model for performance of a TSCH network in the general case of having both dedicated and shared timeslots taking into account non-ideal channel conditions, which are not supported by the existing TSCH models. As a special case of our model, a scenario with of timeslots only shared can be modeled too. In a real scenario for a TSCH-based network, a designer may decide to allocate dedicated and shared links, or only shared links based on channel conditions, packet generation rate, traffic in the network, and performance prediction.

III. TSCH Performance Analysis Model

This section starts by presenting the network model, including various probabilities of successful communication in different timeslots of a TSCH slotframe. Then the algorithm for deriving the probabilities is presented. Finally, the derivation of the performance metrics is explained.

A. System Model

To model the performance of the MAC layer in a general TSCH network, we consider the situation in which N transmitting nodes send their data packets to a common receiver node, called n_r. This case may be a star topology wherein n_r is the PAN coordinator, or it may be a neighborhood in a multi-hop topology. In this work, we focus on applications wherein the nodes drop their data packets if they reach the end of the slotframe, and new data packets are generated in the next slotframe. Let $X = \{n_1, ..., n_N\}$ be the set of N nodes transmitting data to n_r in the neighborhood of n_r. Each slotframe contains N dedicated timeslots for transmission, and M shared timeslots for retransmission. We indicate all timeslots in a slotframe as $L = \{D_1, ..., D_N, S_1, S_2, ..., S_M\}$, where D_i is the dedicated timeslot assigned to n_i ($1 \leq i \leq N$) and S_k ($1 \leq k \leq M$) is a shared timeslot available for retransmission by all transmitting nodes. Each transmitting node generates a data packet at the beginning of each slotframe, and waits for its dedicated timeslot (D_i) to transmit to n_r. If the node did not receive an ACK of the transmitted data packet, it waits for the shared timeslots in the same slotframe for retransmission. The TSCH CSMA/CA back-off procedure is activated when the node fails to receive the ACK in the first shared timeslot.

The goal is to derive performance metrics, namely packet reception probability, packet latency, and energy consumption of a typical node, n_i. For this, we first analyze the communication status of n_i, in conjunction with $N-1$ other nodes in its neighborhood during a slotframe. The status of n_i in each timeslot is modeled by communication probabilities of the data and ACK packets. From all these probabilities for all timeslots, a stochastic model for performance evaluation is extracted.

In this paper, we use notations p^i_{phy-d} and p^i_{phy-a} as the PHY success probabilities of data packet transmission from n_i to n_r and its ACK, respectively, which are derived from PHY layer analysis e.g. as presented in [14], [15]. These probabilities depend on several parameters such as transmission power and receiver sensitivity of the wireless nodes, the distance between n_i and n_r, and the medium properties. Here, these probabilities are assumed to be constant over time or at least for each slotframe duration time.

Let $p^i_d(D_i)$ and $p^i_a(D_i)$ be the success probabilities of data and ACK packets delivery in D_i. Since there is no collision in dedicated timeslots, we have

$$p^i_d(D_i) = p^i_{phy-d} \ , \ p^i_a(D_i) = p^i_{phy-d} \times p^i_{phy-a} \qquad (1)$$

If n_i fails to receive the ACK in D_i, it waits for a shared timeslot to retransmit. Let $p^i_{tx}(S_k, RT)$ be the probability that n_i transmits its data packet in the shared timeslot S_k for its RT^{th} retransmission time (RT indicates the retransmission number of the data packet, $1 \leq RT \leq maxR$). Clearly, n_i experiences its first retransmission in S_1 if it did not receive the ACK in D_i. So, we have

$$p^i_{tx}(S_1, 1) = p^i_a(D_i) = 1 - p^i_a(D_i) = 1 - p^i_{phy-d} \times p^i_{phy-a} \ . \qquad (2)$$

The transmission of n_i in S_k may happen because of the first to $maxR^{th}$ retransmission time. $p^i_{tx}(S_k)$ is the probability that node n_i transmits its data packet in timeslot S_k, which is calculated as follows.

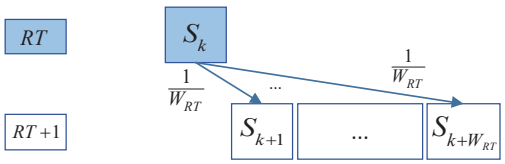


Fig. 1. n_i's back-off after transmission failure in RT^{th} retransmission

$$p^i_{tx}(S_k) = \sum_{RT=1}^{maxR} p^i_{tx}(S_k, RT) \tag{3}$$

Since only the first retransmission of n_i occurs in S_1, then

$$p^i_{tx}(S_1) = p^i_{tx}(S_1, 1) = 1 - p^i_{phy-d} \times p^i_{phy-a} \quad . \tag{4}$$

Note that ACK delivery to n_i in a shared timeslot can fail on account of two reasons. Firstly, the data packet may fail to reach the receiver, and then the receiver does not send ACK to n_i. Secondly, the data packet is delivered to the receiver node successfully, and the receiver sends the ACK to n_i, but the ACK fails to reach n_i. In the latter case, although data packet is delivered, n_i continues the retransmission procedure in the next shared timeslots because it is not aware of the successful data packet delivery. Therefore, it is required to differentiate these two cases since the packet reception probability and the packet latency are calculated from the receiver's point of view by considering the data packet delivery. However, energy consumption of n_i is calculated considering all transmissions. We use the notation $p^i_{tx|nd}(S_k, RT)$ for the probability with which n_i transmits its data packet in timeslot S_k as the RT^{th} retransmission time given that it could not deliver its data packet to n_r in the previous timeslots. So, for $p^i_{tx|nd}(S_1, 1)$ we have

$$p^i_{tx|nd}(S_1, 1) = p^i_{\bar{d}}(D_i) = 1 - p^i_d(D_i) = 1 - p^i_{phy-d} \quad . \tag{5}$$

Let $p^i_{tx|nd}(S_k)$ be the probability that node n_i transmits its data packet in timeslot S_k, given that it could not deliver to n_r in the previous timeslots, calculated as follows.

$$p^i_{tx|nd}(S_k) = \sum_{RT=1}^{maxR} p^i_{tx|nd}(S_k, RT) \tag{6}$$

Accordingly, $p^i_{tx|nd}(S_1)$ is obtained as follows.

$$p^i_{tx|nd}(S_1) = p^i_{tx|nd}(S_1, 1) = 1 - p^i_{phy-d} \tag{7}$$

If the ACK packet does not reach n_i in the first shared timeslot (S_1) of the slotframe, the CSMA/CA back-off algorithm is activated, and the MAC parameters are initialized. If so, n_i randomly selects one of S_2, S_3,...,S_{W_1+1} to transmit for its second retransmission, where $BE = macMinBE$, and $W_1 = 2^{BE}$. Fig.1 shows the CSMA/CA algorithm for n_i in the shared timeslots. To follow the transmission status in the next shared timeslots, it is required to calculate the success probability of data and ACK packets delivery in the shared timeslots. Let, $p^i_{d|tx}(S_k)$ and $p^i_{a|tx}(S_k)$ be the probabilities with which the data or ACK packets are successfully delivered in S_k, given that n_i transmits in S_k. From the probabilities $p^i_{d|tx}(S_1)$ and $p^i_{a|tx}(S_1)$, we derive

$$p^i_{tx}(S_2, 2) = \frac{p^i_{tx}(S_1, 1) \times p^i_{\bar{a}|tx}(S_1)}{W_1} \quad . \tag{8}$$

where the term $p^i_{tx}(S_1, 1) \times p^i_{\bar{a}|tx}(S_1)$ is the probability that n_i performs its first retransmission in S_1, and fails to receive ACK. The term $\frac{1}{W_1}$ is the probability that the node randomly selects one of the timeslots S_2 to S_{W_1+1}. Following the same approach for data delivery in S_1, we have

$$p^i_{tx|nd}(S_2, 2) = \frac{p^i_{tx|nd}(S_1, 1) \times p^i_{\bar{d}|tx}(S_1)}{W_1} \quad . \tag{9}$$

For a shared timeslot, non-ideal channel properties and also internal collisions lead to delivery failure of the data and ACK packets. The impact of the channel effects on the success probabilities in a shared timeslot are known from PHY layer analysis and are taken into account later. Here, we first focus on collision, and define $p^i_{\bar{c}|tx}(S_k)$ as the probability that n_i experiences no collision during its transmission in S_k. A data packet transmission of n_i experiences a collision if at least one of other (transmitting nodes transmits its data packet in S_k. Therefore, $p^i_{\bar{c}|tx}(S_k)$ is calculated as follows.

$$p^i_{\bar{c}|tx}(S_k) = p\Big(\bigcap_{q=1, q\neq i}^{N} \overline{tx_q} | tx_i \Big) \tag{10}$$

where tx_q and tx_i indicate transmission of n_q and n_i in S_k, respectively.

Calculating (10) for each shared timeslot requires heavy computations, which in turns leads to more complexity. The dependency of the nodes on each other decreases when the number of transmitting nodes increases. Therefore, for a large number of transmitting nodes in the network, we can assume that transmission by the nodes in S_k occurs independently, and we use an approximation for $p^i_{\bar{c}|tx}(S_k)$ as follows.

$$p^i_{\bar{c}|tx}(S_k) \simeq \prod_{q=1, q\neq i}^{N} (1 - p^q_{tx}(S_k)) \tag{11}$$

Now, we can calculate $p^i_{d|tx}(S_k)$ and $p^i_{a|tx}(S_k)$ using $p^i_{\bar{c}|tx}(S_k)$, p^i_{phy-d}, and p^i_{phy-a}, as

$$\begin{aligned} p^i_{d|tx}(S_k) &= p^i_{\bar{c}|tx}(S_k) \times p^i_{phy-d} \\ p^i_{a|tx}(S_k) &= p^i_{d|tx}(S_k) \times p^i_{phy-a} \end{aligned} \tag{12}$$

Following the communication status of n_i, if it fails to receive the ACK packet for its second retransmission, it updates BE to $BE = min(macMaxBE, BE+1)$, and sets $W_2 = 2^{BE}$. Then, it randomly selects one of S_3,..., S_{W_2+2} to transmit its data packet in the third retransmission time. This procedure continues until the number of retransmissions, RT, exceeds its maximum value ($maxR$), or k reaches the end of the slotframe.

Given p^i_{phy-d} and p^i_{phy-a} from the PHY layer, the probabilities $p^i_{tx}(S_k, RT)$ and $p^i_{tx|nd}(S_k, RT)$ are required to calculate $p^i_{tx}(S_k)$ and $p^i_{tx|nd}(S_k)$, which are used to compute success probabilities of data and ACK packets as well as the performance metrics. On the other hand, $p^i_{tx}(S_k, RT)$ and $p^i_{tx|nd}(S_k, RT)$ depend on the success probabilities of data and ACK packets in the previous timeslots. Hence, we can calculate communication probabilities for all shared timeslots by using an iterative algorithm. In the next section, we describe our approach to derive the communication probabilities, and then derive the performance metrics from them.

B. Deriving Communication Probabilities

In this section, we propose an iterative algorithm to calculate the communication probabilities defined in the previous section for both dedicated and shared timeslots in a slotframe for all nodes. Let P^i_{tx} and $P^i_{tx|nd}$ be two matrices with size $maxR \times M$, that are defined as

$$P^i_{tx} = \left[p^i_{tx}(S_k, RT)\right]_{RT,k},\ P^i_{tx|nd} = \left[p^i_{tx|nd}(S_k, RT)\right]_{RT,k} \quad 1 \le RT \le maxR,\ \ 1 \le k \le M \tag{13}$$

where element (RT, k) of matrix P^i_{tx} (or $P^i_{tx|nd}$) is $p^i_{tx}(S_k, RT)$ (or $p^i_{tx|nd}(S_k, RT)$).

The goal is to calculate the elements of matrices P^i_{tx} and $P^i_{tx|nd}$ for $1 \le i \le N$. Algorithm 1 describes the iterative method to calculate these probabilities. It starts by configuring the MAC parameters and initializes matrices P^i_{tx} and $P^i_{tx|nd}$ for all nodes in a loop. First elements of P^i_{tx} and $P^i_{tx|nd}$ are initialized through (2) and (5) for each node in lines 3 and 4. The analysis starts from line 6, which is the main loop that traverses over all shared timeslots. In each shared timeslot, for each node, $p^i_{tx}(S_k)$ and $p^i_{tx|nd}(S_k)$ are calculated using elements of P^i_{tx} and $P^i_{tx|nd}$. Then, $p^i_{d|tx}(S_k)$ and $p^i_{a|tx}(S_k)$ are calculated through (12). Then, for all nodes, algorithm enters to an inner loop, which traverses over all retransmissions until $maxR - 1$. As shown in Fig.1, when n_i performs its RT^{th} retransmission in S_k and fails to receive an ACK, W_{RT}, indicating the Back-off Window (BW) size in the $(RT+1)^{th}$ retransmission, is generated, and n_i selects any of S_{k+1}, S_{k+2}, ..., and $S_{k+W_{RT}}$ timeslots to retransmit as its $(RT+1)^{th}$ retransmission. That is why the algorithm has its most inner loop in each iteration of which the probabilities $p^i_{tx}(S_{k+l}, RT+1)$ and $p^i_{tx|nd}(S_{k+l}, RT+1)$, are increased. Hence, in each iteration of k in the main loop for all nodes, and for each retransmission from $RT = 1$ to $RT = maxR - 1$, the values of the elements $(RT+1, k+1)$, ... , $(RT+1, k+W_{RT})$ in the matrices P^i_{tx} and $P^i_{tx|nd}$ are updated as lines 16 and 17.

Note that the PHY layer success probabilities for different transmitting nodes may be different. In the next section, we derive performance metrics of a transmitting node in a neighborhood of n_r from the communication probabilities that are calculated in Algorithm 1.

Algorithm 1: Deriving the communications probabilities

Input: $macMinBE$, $macMaxBE$, $maxR$, and p^i_{phy-d}, p^i_{phy-a} for all transmitting nodes $(1 \le i \le N)$
Output: $p^i_{tx}(S_k)$, $p^i_{tx|nd}(S_k)$, $p^i_{d|tx}(S_k)$,$p^i_{a|tx}(S_k)$ for all transmitting nodes ($1 \le i \le N$)

```
   /* Initialization
1  for i = 1 to N do
2      P^i_tx and P^i_tx|nd set to zero matrices.
3      P^i_tx(1,1) ← 1 − p^i_phy−d × p^i_phy−a
4      P^i_tx|nd(1,1) ← 1 − p^i_phy−d
5  end
6  for k = 1 to M do
7      for i = 1 to N do
8          p^i_tx(S_k) = Σ_{RT=1}^{min(k,maxR)} P^i_tx(RT, k)
9          p^i_tx|nd(S_k) = Σ_{RT=1}^{min(k,maxR)} P^i_tx|nd(RT, k)
10     end
11     for i = 1 to N do
12         Calculate p^i_d|tx(S_k), and p^i_a|tx(S_k)
13         for RT = 1 to maxR − 1 do
14             W_RT ← 2^{min(macMinBE+RT−1, macMaxBE)}
15             for l = 1 to W_RT do
16
                   P^i_tx(RT+1, k+l) ← P^i_tx(RT+1, k+l) +
                       P^i_tx(RT, k) × p^i_ā|tx(S_k) / W_RT
17
                   P^i_tx|nd(RT+1, k+l) ← P^i_tx|nd(RT+1, k+l) +
                       P^i_tx|nd(RT, k) × p^i_d̄|tx(S_k) / W_RT
18             end
19         end
20     end
21 end
```

C. Performance Estimation Models

In the previous sections, the communication status of a transmitting node, n_i, in a neighborhood of n_r as a common receiver during a slotframe is modeled with communication probabilities. In the following, we use these probabilities to derive performance metrics, namely packet reception probability, packet latency, and energy consumption.

Packet Reception Probability (PRP): gives the probability that a transmitting node successfully delivers its data packet to the receiver during a slotframe regardless of whether the node receives the ACK (it is defined from the receiver's perspective). Thus, for a typical node n_i, PRP is calculated as

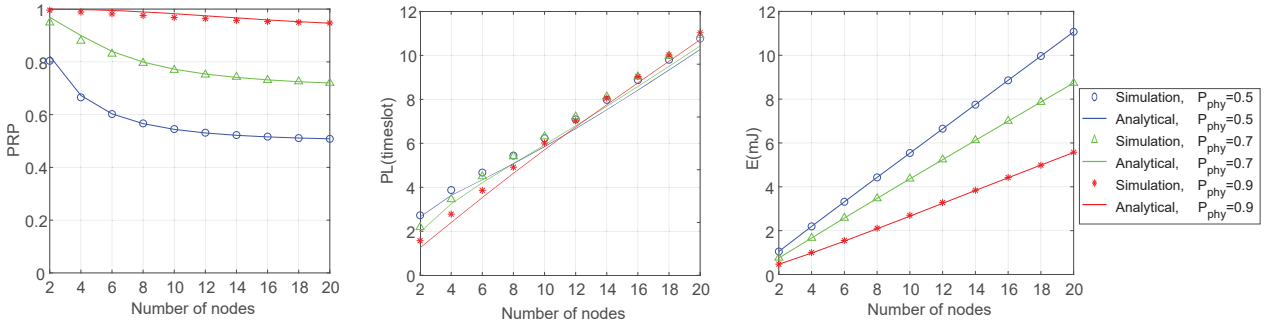
$$PRP_i = p^i_d(D_i) + \sum_{k=1}^{M} p^i_{tx|nd}(S_k) \times p^i_{d|tx}(S_k) \quad . \tag{14}$$

Packet Latency (PL): is defined as the time between packet generation by the transmitting node, and its first delivery to the receiver, averaged over all delivered data packets from that node. We assume that n_i generates a new data packet at the beginning of each slotframe. Note that latency is calculated only for data packets that are successfully delivered to the receiver before the end of the slotframe. Hence, PL_i, indicating latency of n_i, can be expressed in terms of timeslots using weighted mean as follows.

$$PL_i = \frac{i \times p^i_d(D_i) + \sum_{k=1}^{M}[(N+k) \times p^i_{tx|nd}(S_k) \times p^i_{d|tx}(S_k)]}{p^i_d(D_i) + \sum_{k=1}^{M}[p^i_{tx|nd}(S_k) \times p^i_{d|tx}(S_k)]} = \frac{i \times p^i_{phy-d} + \sum_{k=1}^{M}[(N+k) \times p^i_{tx|nd}(S_k) \times p^i_{d|tx}(S_k)]}{PRP_i} \tag{15}$$

Note that PL_i is equal to i timeslots if the n_i's transmission in its dedicated timeslot (D_i) succeeds. Otherwise, PL_i is equal to $N+k$ timeslots if the data packet gets delivered in the shared timeslot S_k in the slotframe.

Energy consumption (E): is the average energy consumed by a node to transmit a data packet to the receiver and receive its ACK. Each time that n_i transmits its data packet regardless of its success or failure, it is considered in energy consumption. So, E_i indicating energy consumption by n_i is obtained as



(a) Average PRP (b) Average PL (c) Average network energy consumption

Fig. 2. Analytical and simulation results vs different physical layer reliabilities and different number of nodes.

$$E_i = E_s \times \left(p_a^i(D_i) + \sum_{k=1}^{M} p_{tx}^i(S_k) \times p_{a|tx}^i(S_k) \right) + E_f \times \left(p_{\bar{a}}^i(D_i) + \sum_{k=1}^{M} p_{tx}^i(S_k) \times p_{\bar{a}|tx}^i(S_k) \right) \quad , \tag{16}$$

where $p_{\bar{a}|tx}^i(S_k) = 1 - p_{a|tx}^i(S_k)$. E_s denotes the energy consumption to transmit a data packet successfully and receive its ACK, while E_f is the energy consumption to transmit a data packet and not receive its ACK. This can occur due to a collision or non-ideal channel effects. E_s and E_f are computed as

$$E_s = P_{TX} \times T_{TX} + P_{RX} \times T_{ACK} \tag{17}$$

$$E_f = P_{TX} \times T_{TX} + P_{RX} \times T_{to} \tag{18}$$

where P_{TX} and P_{RX} are the radio power consumption of the node in transmitting and receiving states, respectively. T_{TX}, and T_{ACK} are the time durations of data packet transmission and receiving the ACK, respectively. In case that the ACK packet is lost, the standard specifies a time-out (T_{to}) until which the node continues listening to the channel for the ACK. Afterward, the data packet transmission is considered as failed by the transmitter.

IV. Performance Evaluation

We implemented our analytical model in Matlab. To verify the model, we use simulations, real-world experiments, and a state-of-the-art analytical model. For evaluations, we use a network of N transmitting nodes with different TSCH schedules for communications towards a common receiver node. This studies the communications inside a cluster within a larger network. The worst-case contention in such a cluster happens when all children nodes generate data packets at the same time. Accordingly, we study the case that all nodes generate data packets of 100 bytes at the beginning of slotframes. This also enables repeatable evaluations with the same traffic load. The ACK packets are 11 bytes in length.

A. Model Verification

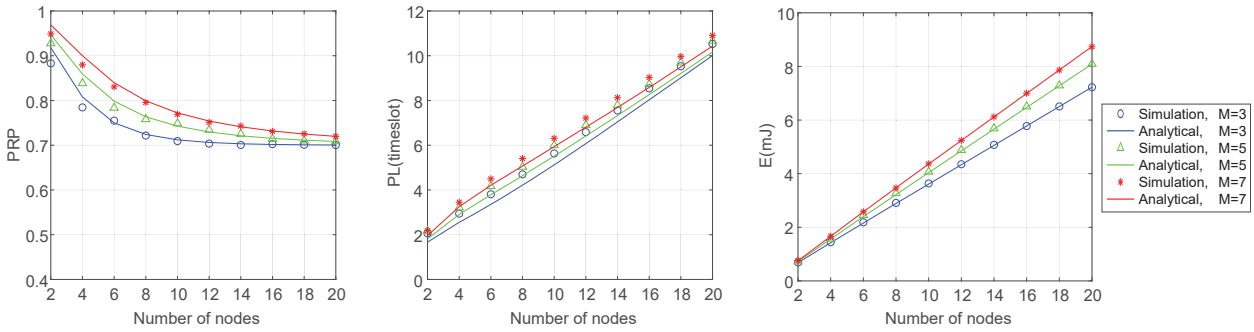
To verify the accuracy of our model under non-ideal channels, we compare the results against simulations and experiments. For this, we use slotframes that consist of one dedicated timeslot per link, followed by a number of shared timeslots. The Cooja network simulator [8] is used for simulations. This simulator uses the implementation of the TSCH protocol stack on top of the Contiki [16] operating system. Sky motes that emulate the behavior of the TelosB/Tmote Sky platform [17] are used for simulations. Each simulation consists of 10000 packet transmissions. The radio power consumption of these motes is $P_{TX} = 37.5mW$ in transmission mode, and $P_{RX} = 56.4mW$ in the receive mode.

Fig. 2 shows the results for different number of nodes and different PHY success probabilities. In this setup, the slotframe has one dedicated timeslot per node and $M = 7$ shared timeslots. The TSCH CSMA/CA parameters are set to $macMinBE = 1$, $macMaxBE = 2$, and $maxR = 3$ to use all the shared timeslots. All the results for PRP and PL are the average values over all nodes. The results of simulations and the model are very close to each other. However, for small networks, there is a bit higher gap between the average PRP and PL results. This is due to the approximation that is used in the model to ignore the dependency between nodes when they access the shared timeslots. This dependency decreases when the number of competing nodes increases, which leads to lower deviation from the simulation results. The PL gap for larger networks is less than half a timeslot due to the buffering delay between the MAC and the application layers of the nodes in simulations.

To measure the energy consumption in simulations, we count the number of successful and failed communications, and using the same E_s and E_f as the ones used for the model, we compute the energy consumptions. In any case, the sum of average energy consumption of all nodes is reported. Fig. 2(c) shows that the results of our model follow the simulations results for different network sizes and PHY success probabilities.

Fig. 3 presents the results for different number of shared timeslots ($M = 3,\ 5$, and 7), when $p_{phy} = 0.7$. This setup also shows that the results of our model are very close to the simulation results. The error between the model and simulation results follow almost the same trend as the results of Fig. 2. Both model and simulations results show that increasing the number of shared timeslots in a TSCH schedule can improve the PRP, but causes longer packet latencies as well as higher energy consumption.

Computer simulations or analytical models may deviate from real-world behavior to some extend due to unpredictable behavior of the channel or real hardware. To explore the performance of our model in this regards, we complement our evaluations with real-world experiments. We deploy TSCH networks with the same setup as simulations, using a different number of NXP JN5168 motes [18]. These motes have a 32-bit RISC processor



(a) Average PRP (b) Average PL (c) Average network energy consumption

Fig. 3. Analytical and simulation results vs different number of shared timeslots (M) and different number of nodes, under $p_{phy} = 0.7$.

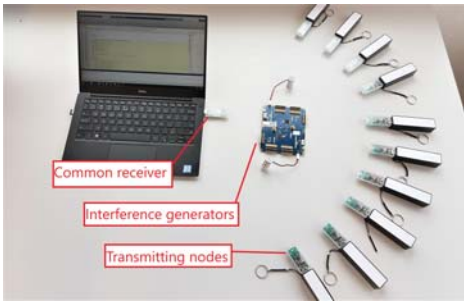
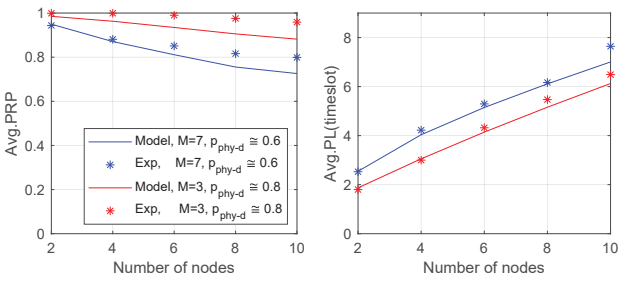


Fig. 4. Experimental setup for 10 transmitting nodes.



(a) Average PRP (b) Average PL

Fig. 5. Analytical vs experimental results for different number of nodes and shared timeslots (M).

and a 2.4GHz IEEE 802.15.4 compliant transceiver.

The experiments are conducted in an electromagnetically isolated room. To imitate the non-ideal channel conditions in a repeatable way, controlled interference is generated in one or two frequency channels out of the five channels used in the hopping sequence list. Fig. 4 shows one of the experimental setups. We run each setup for 10000 independent slotframes and measure the packet and ACK reception ratio in the dedicated timeslots for each transmitting node. These values are then fed to the analytical model as p_{phy-d} and p_{phy-a} probabilities for each individual node. Since the generated interference in our experiments is channel-based, success of ACKs depend only on the packet delivery status. Accordingly, experimental results show $p_{phy-a} = 1$ for all nodes. In the experiments, we set $macMinBE = 1$, $macMaxBE = 3$, and $maxR = 3$.

Fig. 5 depicts the average PRP and average PL for different number of nodes and setups. Fig. 5(a) shows that PRP values measured in experiments is greater than that of the model. Moreover, by increasing the number of nodes in the network, the error between the PRP of our model and experiments gets higher. This is due to the capture effect that occurs when some nodes concurrently transmit their data packets in a shared timeslot. In this situation, the signal of one of the transmitters at the receiver side may be stronger than others, and be successfully received by the receiver node despite the collision with the other nodes. However, the model assumes that all concurrent packet transmissions fail. Accordingly, as the number of shared timeslots with contention increases due to the increase in the number of nodes, the capture effect is more pronounced. This effect is very hard to model, as it depends on many temporal and spatial parameters of the WSNs (such as antenna radiation pattern of each sensor node). However, the results of the model still show the worst-case PRP, when capture effect is zero.

As shown in Fig. 5(b), the value of PL in experiments is very close to that of the model, but always a bit higher. The packet latency is calculated only for the packets that are successfully delivered. As discussed, the capture effect leads to higher packet delivery probability in the shared timeslots that are placed at the end of the slotframe. This leads to a bit higher average of PL in the experiments compared to the results of the model.

B. Model performance vs state-of-the-art

This set of evaluations targets the model performance in terms of accuracy and speed, compared to the state-of-the-art analytical model presented in [5]. This model is considered to be very accurate, but computation intensive. We implemented this model in Matlab to be used for our comparison. This model does not cover dedicated timeslots in the TSCH schedule, neither the non-ideal channels. To enable this comparison, we use a schedule with only shared timeslots and ideal channels in this set of evaluations. Different combinations of $BE = macMinBE = macMaxBE$, and $maxR$ are tested to study different network configurations. A time limit of 1000 seconds is used to run each model for each setup (setups with longer execution times are skipped). Table I shows the percentage of error between the results of our model and the model in [5] for setups with a various number of nodes. For very small networks, our model deviates a little from the model in [5]. This is because of the approximation that we used in our model regarding the dependency of nodes in the shared timeslots. However, for larger networks, the results of the two models are

TABLE I
ERROR OF THE PROPOSED MODEL IN COMPARISON WITH MODEL IN [5]

Error %		Number of nodes							
		(BE,maxR)	2	4	6	8	10	12	14
	Avg.PRP	(1,2)	9.11	3.85	0.49	0.05	$5E^{-3}$	$6E^{-4}$	$6E^{-5}$
		(1,3)	10.03	4.73	0.80	0.19	0.05	0.01	$2E^{-3}$
		(2,2)	2.67	0.58	0.9	-	-	-	-
		(2,3)	1.08	2.57	-	-	-	-	-
	Avg.PL	(1,2)	8.79	0.09	0.01	$1E^{-3}$	$8E^{-5}$	$6E^{-6}$	$3E^{-7}$
		(1,3)	7.48	0.07	0.24	0.05	$7E^{-3}$	$5E^{-4}$	$8E^{-6}$
		(2,2)	3.90	1.48	0.22	-	-	-	-
		(2,3)	1.25	2.9	-	-	-	-	-

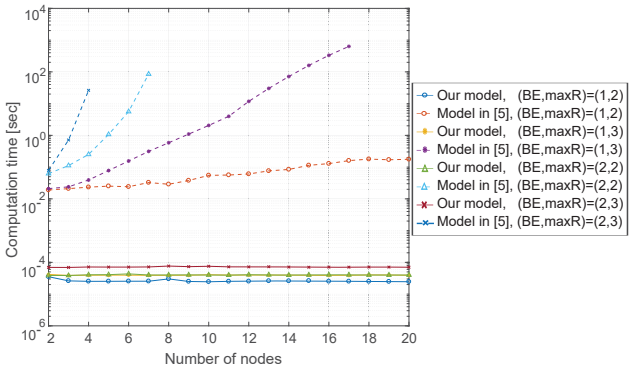


Fig. 6. Computation time vs network sizes, for different network setups.

very close. This confirms the assumption made in (11) about the lower dependency between transmissions, when the number of nodes is increased.

Speed or execution time is a key factor for performance models, as a faster model enables more evaluations to be done in a given time to explore different setups for a network. Fig. 6 shows the Matlab execution time for the two models on a logarithmic scale, for different network configurations. Results show that the computation time of the model in [5] exponentially increases with increasing the number of nodes, and higher values of MAC parameters. As our model iterates over timeslots to compute the network performance, these settings hardly affect the computation time of our model. Our model provides not only an accurate enough but also a very fast evaluation technique that can be used for optimization of TSCH networks.

V. CONCLUSION

We propose a scalable statistical model for performance analysis of the IEEE 802.15.4 TSCH networks. This model takes into account both dedicated and shared timeslots in a TSCH schedule. Moreover, the proposed model considers the transmission failures in the physical layer as input. An iterative algorithm extracts communication success probabilities of transmitting nodes for a given TSCH schedule. Performance metrics in terms of packet reception probability, packet latency, and energy consumption are derived using the proposed model. Simulations and real-world experiments verify this model in terms of accuracy. Simulation results show that this model is able to provide more than 98% accuracy for the networks with more than 8 nodes. Experimental results show the importance of the capture effect on the performance of the network, which is not considered in simulations and the model. A comparison with a state-of-the-art analytical model shows that our model runs orders of magnitude faster, while it provides comparable accuracy. As future work, we plan to extend our model to include the capture effect and packet buffering effect on the performance of TSCH networks.

VI. ACKNOWLEDGMENT

This work was supported by the SCOTT European project (www.scott-project.eu) that has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737422.

REFERENCES

[1] "IEEE Standard for Low-Rate Wireless Networks," *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)*, pp. 1–709, April 2016.

[2] T. Watteyne, J. Weiss, L. Doherty, and J. Simon, "Industrial IEEE 802.15.4e networks: Performance and trade-offs," in *2015 IEEE Int'l Conf. on Communications (ICC)*, June 2015, pp. 604–609.

[3] R. C. A. Alves and C. B. Margi, "IEEE 802.15.4e TSCH Mode Performance Analysis," in *2016 IEEE 13th Int'l Conf. on Mobile Ad Hoc and Sensor Systems (MASS)*, Oct 2016, pp. 361–362.

[4] S. Chen, T. Sun, J. Yuan, X. Geng, C. Li, S. Ullah, and M. Abdullah Alnuem, "Performance Analysis of IEEE 802.15.4e Time Slotted Channel Hopping for Low-Rate Wireless Networks." *KSII Trans. on Internet & Information Systems*, vol. 7, no. 1, 2013.

[5] D. De Guglielmo, B. Al Nahas, S. Duquennoy, T. Voigt, and G. Anastasi, "Analysis and experimental evaluation of IEEE 802.15.4e TSCH CSMA-CA algorithm," *IEEE Trans. on Vehicular Technology*, vol. 66, no. 2, pp. 1573–1588, 2017.

[6] C. Ouanteur, L. Bouallouche-Medjkoune, and D. Aïssani, "An Enhanced Analytical Model and Performance Evaluation of the IEEE 802.15.4e TSCH CA," *Wireless Personal Communications*, vol. 96, no. 1, pp. 1355–1376, 2017.

[7] S. B. Yaala, F. Tholeyre, and R. Bouallegue, "Performance Modeling of IEEE 802.15.4-TSCH with Shared Access and ON-OFF traffic," in *14th Int'l Wireless Communications Mobile Computing Conf. (IWCMC)*, June 2018, pp. 352–357.

[8] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with cooja," in *Local computer networks, proceedings 2006 31st IEEE Conf. on*. IEEE, 2006, pp. 641–648.

[9] C. Buratti and R. Verdone, "Performance analysis of IEEE 802.15.4 non beacon-enabled mode," *IEEE Trans. on Vehicular Technology*, vol. 58, no. 7, pp. 3480–3493, 2009.

[10] C. Buratti, "Performance analysis of IEEE 802.15.4 beacon-enabled mode," *IEEE Trans. on Vehicular Technology*, vol. 59, no. 4, pp. 2031–2045, 2010.

[11] S. Pollin, M. Ergen, S. C. Ergen, B. Bougard, L. Van der Perre, I. Moerman, A. Bahai, P. Varaiya, and F. Catthoor, "Performance analysis of slotted carrier sense IEEE 802.15.4 medium access layer," *IEEE Trans. on wireless communications*, vol. 7, no. 9, 2008.

[12] P. Park, P. Di Marco, P. Soldati, C. Fischione, and K. H. Johansson, "A generalized Markov chain model for effective analysis of slotted IEEE 802.15.4," in *Mobile Adhoc and Sensor Systems, 2009. MASS'09. IEEE 6th Int'l Conf. on*. IEEE, 2009, pp. 130–139.

[13] D. De Guglielmo, S. Brienza, and G. Anastasi, "IEEE 802.15.4e: A survey," *Computer Communications*, vol. 88, pp. 1–24, 2016.

[14] S. Y. Shin, H. S. Park, and W. H. Kwon, "Packet error rate analysis of IEEE 802.15.4 under saturated IEEE 802.11 b network interference," *IEICE Trans. on communications*, vol. 90, no. 10, pp. 2961–2963, 2007.

[15] M. B. Mantri, P. Velagapudi, B. C. Eravatri, and V. Mani, "Performance analysis of 2.4 GHz IEEE 802.15.4 PHY under various fading channels," in *Emerging Trends in Communication, Control, Signal Processing & Computing Applications (C2SPCA), 2013 Int'l Conf. on*. IEEE, 2013, pp. 1–4.

[16] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *Local Computer Networks, 2004. 29th Annual IEEE Int'l Conf. on*. IEEE, 2004, pp. 455–462.

[17] MEMSIC Inc., "TelosB mote platform," accessed: March. 2018. [Online]. Available: http://www.memsic.com/userfiles/files/Datasheets/WSN/telosb_datasheet.pdf

[18] NXP Semiconductors, "JN516x IEEE802.15.4 Wireless Microcontroller," accessed: Sept. 2017. [Online]. Available: http://www.nxp.com/docs/en/data-sheet/JN516X.pdf