Throughput Modeling of the IEEE MAC for Sensor Networks

Martijn Onderwater[1,3]
m.onderwater@cwi.nl

Gerard Hoekstra[2,1]
g.hoekstra@cwi.nl

Rob van der Mei[1,3]
r.d.van.der.mei@cwi.nl

[1]CWI
Stochastics Group
Science Park 123
1098XG, Amsterdam
The Netherlands

[2]Thales
Innovation Research & Tech.
Bestevaer 46
1271ZA, Huizen
The Netherlands

[3]VU University
Faculty of Sciences
De Boelelaan 1081a
1081HV, Amsterdam
The Netherlands

ABSTRACT

In this paper we provide a model for analyzing the saturation throughput of the IEEE 802.15.4 MAC protocol, which is the de-facto standard for wireless sensor networks, ensuring fair access to the channel. To this end, we introduce the concept of a *natural layer*, which reflects the time that a sensor node typically has to wait prior to sending a packet. The model is simple and provides new insight how the throughput depends on the protocol parameters and the number of nodes in the network. Validation experiments with simulations demonstrate that the model is highly accurate for a wide range of parameter settings of the MAC protocol, and applicable to both large and small networks. As a by-product, we discuss fundamental differences in the protocol stack and corresponding throughput models of the popular 802.11 standard.

Keywords

IEEE 802.15.4, performance, throughput, natural layer, sensor network

1. INTRODUCTION

Over the past few years, the use of sensor networks has been growing at an unprecedented rate, which is not likely to slow down in the near future. Consequently, sensor networks are expected to process highly increasing amounts of sensor data, stressing the processing capabilities of the network. This development has raised the need for sensor network providers to understand the speed limitations of the network. Motivated by this, in this paper we propose a model to analyze how the saturation throughput of a network depends on the MAC protocol parameters and the number of nodes in the network.

The most widely used standard for sensor networks is the IEEE 802.15.4 protocol, which is aimed at providing low-cost, low-power communications for resource-limited devices. It is particularly suitable for sensor networks, since sensor nodes are typically battery powered and have few computational resources available. Part of this standard is the MAC protocol, which is responsible for governing access to the wireless channel. In particular, it describes the collision avoidance (CSMA-CA) mechanism employed by nodes to limit loss of packets due to collisions. In essence, this mechanism instructs nodes to wait a random amount of time before attempting a transmission. This randomization of medium access does, however, come at a price, because it decreases the effective throughput that can be obtained compared to the maximum specified in the standard.

In the literature, much work has been done on analyzing throughput of the IEEE 802.15.4 MAC protocol. The authors of [1] analyze both throughput and delay of the unslotted IEEE 802.15.4 MAC for a simple network containing a single node. They formulate an expression for the throughput and the delay in terms of the protocol parameters, and verify these with results from a real sensor network. [2] looks at the unslotted MAC in more detail and formulates a three dimensional Markov Chain for the CSMA-CA process. From this, the authors find an expression for the stationary distribution of the Markov chain, and use a fixed-point iteration on three unkown variables to derive expressions for link reliability, packet delay, and energy consumption. The results are valid for both a network in star formation, as well as for a general multi-hop network. For large-scale networks the authors construct an approximate model, in order to keep computations numerically tractable. Simulations are used to validate their model. [3] analyzes throughput for the MAC by combining a renewal process for the physical layer with a semi-Markov process for the MAC layer. Their analysis results in a fixed-point iteration on the idle probabilities of each backoff stage in the CSMA-CA process. The throughput obtained from this model closely resembles values observed in a discrete event simulation.

For the slotted MAC, [4] takes an approach similar to [2]. The authors investigate throughput by constructing a two-dimensional Markov Chain, and derive an expression for the throughput using a fixed-point iteration on three unknown variables. They then compare the results of the model with the outcome of simulations, and demonstrate that their model accurately captures the throughput. In [5] the authors look at various performance metrics, including throughput and average service time for a transmission. Their method relies on viewing a cycle of a transmission and the subsequent waiting by a node as a renewal process. They derive a model that is solved via a fixed point iteration, and

demonstrate its accuracy by comparing it to results from a discrete event simulation.

Although the papers mentioned above provide insight into the throughput behavior of sensor networks, the models involved are typically rather complex. Motivated by this, the goal of this paper is to provide a simple yet accurate model for analyzing the throughput in sensor networks. To this end, we propose a new concept called the *natural layer* which reflects the time a sensor node typically has to wait as part of the CSMA-CA process prior to sending a packet (as detailed in Section 2.1). Using this concept, we develop a simple model for determining the throughput. Contrary to existing models, our model does not require calculating the stationary distribution of a Markov chain, and uses a fixed-point iteration with just one unknown variable (the natural layer). Additionally, our model provides new insights into the differences between IEEE 802.15.4 and the popular 802.11 standard. In particular, we highlight the aspect of 'freezing' in the 802.11 protocol, and discuss how the absence of freezing in IEEE 802.15.4 influences the saturation throughput.

The organization of the paper is as follows. In Section 2 we outline the IEEE 802.15.4 CSMA-CA protocol, list model assumptions and notation for our analysis. Then, in Section 3 we derive an expression for the throughput of sensor networks with a single backoff layer. Subsequently, in Section 4 we introduce the concept of a natural layer and use this to extend the model to a setting with multiple layers. In Section 5 we show simulation results to demonstrate that the model captures the throughput accurately for a wide range of parameter settings. Next, in Section 6 we discuss key differences between the IEEE 802.15.4 and 802.11 standards, and how these differences influence modeling of the throughput. We conclude in Section 7 with concluding remarks and ideas for future research.

2. PRELIMINARIES

In this section we briefly outline the IEEE 802.15.4 CSMA-CA protocol, list the model assumptions and provide some preliminary remarks.

2.1 The IEEE 802.15.4 CSMA-CA protocol

The CSMA-CA mechanism prevents collisions between packets by enforcing a random waiting time, prior to sending a packet. This waiting time affects throughput, and a thorough understanding of the CSMA-CA mechanism is key to modeling throughput.

The CSMA-CA mechanism states that prior to sending a packet, a node should wait a random amount of time (known as the *backoff time*). This time is drawn uniformly from the interval $[0, W_0 - 1]$, where W_0 is the initial *backoff window* (controlled via parameter *macMinBE*). The resulting backoff time is discrete, and corresponds to the number of time slots that the node has to wait. The length (in seconds) of a time slot is defined in IEEE 802.15.4. After the required backoff time, the node assesses if the channel is idle and if it is, the node sends the packet. If, however, the channel is busy, the window W_0 is doubled and the backoff process starts again. This procedure is repeated until the packet is sent. The next packet goes through the same procedure, again starting with backoff window W_0.

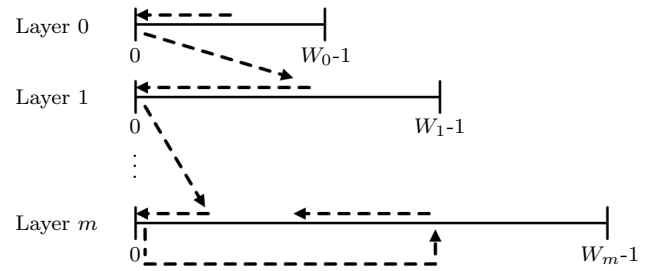Initially, the backoff window W_0 is set to $2^{macMinBE}$ and it

Figure 1: The csma-ca mechanism applied by each node in the network.

is repeatedly doubled during the CSMA-CA process. However, overly long backoff times cause unnecessary delays, so the CSMA-CA mechanism defines a maximum backoff exponent (*macMaxBE*). Once the backoff window reaches $2^{macMaxBE}$, the doubling is disabled.

Fig. 1 illustrates the CSMA-CA mechanism as described above. The node starts in layer 0 and draws a random backoff time from the interval $[0, W_0 - 1]$. Then it waits until this backoff time has passed, and does a channel assessment. If the channel is busy, the process moves to layer 1. The window in layer 1 is twice as large as that of layer 0 ($W_1 = 2 \cdot W_0$) because of the doubling of the window. The node now draws a backoff time from $[0, W_1 - 1]$, and again waits until this time has passed. The window is repeatedly doubled until the backoff exponent reaches *macMaxBE* (layer m), at which point the doubling is disabled. The CSMA-CA process then continues until the packet is sent successfully.

2.2 Assumptions

Before starting our throughput analysis we mention several assumptions we make in this paper.

1. All nodes in the network are within sensing range of eachother.

2. The network is saturated, meaning that nodes always have a packet ready for transmission. Consequently, there are no periods of inactivity on the channel caused by a lack of packets.

3. The network is in non-beacon mode, and uses the unslotted CSMA-CA mechanism.

4. Packets go through the CSMA-CA mechanism until they are sent.

5. Acknowledgements are disabled.

Assumptions 1-4 are made so that our analysis closely resembles the scenario analyzed in [6] for 802.11. Note that Assumption 2 may not be realistic for lightly loaded sensor networks, but provides a valuable throughput-performance guarantee for heavily loaded 802.15.4-based sensor networks. The last assumption is revisited in Section 6.

2.3 Preliminary remarks

Notation. The IEEE 802.15.4 MAC specifies several variables that influence its operation, for instance the *macMinBE* and *macMaxBE* that we saw previously. These are quite cumbersome in a mathematical analysis, so we use a different notation in the remainder of this paper. Table 1 provides

Notation	Description
W_0	Backoff window for layer 0 $\left(= 2^{macMinBE}\right)$
m	$macMaxBE\text{-}macMinBE$
W_m	Maximum backoff window $\left(= 2^m\right)$
$U_x^{(k)}$	Random variable for the backoff time at layer x for node k ($0 \leq k \leq n$)
$\bar{U}_x^{(k)}$	Random variable of the remainder of $U_x^{(k)}$
S_C	Throughput on the channel
S_N	Throughput achieved by a single node
T	Number of time slots (possibly non-integer) required to send one packet

Table 1: Notation used in this paper.

a translation from the names in IEEE 802.15.4 to the mathematical notation, and also contains several elements that we refer to later in this paper.

Continuous backoff time. For this paper, we assume that the random backoff times in the CSMA-CA mechanism are drawn from a *continuous* uniform distribution, even though IEEE 802.15.4 specifies a *discrete* uniform distribution. This is done purely for notational convenience, and our method works for discrete uniform distributions as well.

Channel speed. We set the channel speed (and thus the maximum throughput) to 250,000 b/s. The IEEE 802.15.4 standard specifies several options for the channel speed, depending on configuration and geographical location. Our choice for the channel speed is not essential to the model in this paper; it works for other channel speeds as well. To emphasize this, we always normalize the throughput to the interval $[0, 1]$ when reporting on it.

Unit of time. At first, it seems natural to report on time in units of seconds, but this has several drawbacks. First, the time scales involved are small (in the order of fractions of milliseconds), and are thus somewhat laborious to work with. Second, the times depend on the speed of the channel, and this can vary per configuration and per geographical region. Even though we choose a certain channel speed in this paper, our analysis works for other choices as well. In order to preserve this neutrality, we use the time slots from the CSMA-CA process as the unit of time throughout this paper. These time slots are configuration- and region-neutral, and can easily be converted to seconds if needed (the method is described in the IEEE 802.15.4 standard). An additional benefit of using time slots is that we can quickly compare, e.g., the time needed to transmit a packet to the waiting times described by the CSMA-CA process. Finally, note that a non-integer number of time slots is also meaningful when using them as unit of time – for instance, a single packet transmission takes 12.7 time slots.

3. SINGLE-LAYER ANALYSIS

We start our throughput analysis by looking at a simplified version of the CSMA-CA mechanism. In this section we assume that $m = 0$, i.e., that the CSMA-CA uses just one layer
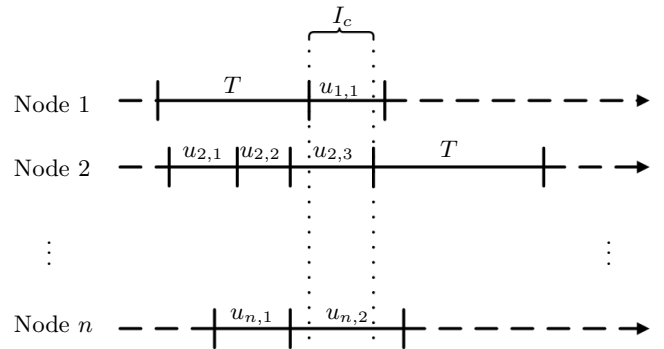

Figure 2: A timeline for the n nodes in the network, each going through the csma-ca process. The figure shows two transmissions (marked by T), several backoff periods $(u_{1,1}, \ldots, u_{n,2})$, and a period of time where the channel is idle (I_c).

(layer 0). This simplified scenario forms an introduction to the complete throughput analysis, later in this paper.

Fig. 2 shows n nodes going through the CSMA-CA process of transmitting packets (marked by T) and backing off $(u_{1,1}, \ldots, u_{n,2})$. The interval lengths $u_{1,1}, \ldots, u_{n,2}$ are backoff times drawn from the uniform distribution on interval $[0, W_0 - 1]$ (all nodes are at backoff layer 0 in our simplified scenario). Node 1 is the first to send a packet, and during the transmission at node 1, the other nodes are going through several backoff periods. In particular, node 2 starts three backoff periods (of length $u_{2,1}, u_{2,2}, u_{2,3}$, respectively), and node n starts two periods (of length $u_{n,1}$ and $u_{n,2}$). At the end of the transmission at node 1, this node also starts a backoff period (of length $u_{1,1}$). The first backoff period to end after the transmission at node 1 is the one of length $u_{2,3}$ at node 2, so the next transmission occurs at node 2. This process continues over time.

For determining throughput, we analyze the time that the channel is idle in between two transmissions. In Fig. 2, this idle time (denoted by I_c) is the time between the end of the transmission at node 1, and the start of the next transmission at node 2. Note that at the end of the packet transmission at node 1, the other nodes have already partly gone through a backoff period. Node 1, in contrast, starts a new backoff period. Our throughput analysis takes these two aspects into account: we determine the distribution of the channel idle time using the distribution of the length of the backoff period at node 1, and using the distribution of the residual of the backoff periods at the other nodes.

To start the analysis we introduce some notation, see also Table 1. First, $U_0^{(k)}$ is the random variable representing the backoff time at layer 0 for node k ($1 \leq k \leq n$). It is uniformly distributed (continuously, by assumption) on the interval $[0, W_0 - 1]$. The corresponding probability density function (pdf) and cumulative distribution function (cdf) are denoted by $f_{U_0^{(k)}}(t)$ and $F_{U_0^{(k)}}(t)$, respectively.

Next we formulate the distribution of the residual of $U_0^{(k)}$. To be precise, suppose a node starts a backoff period of length t and the node that is currently sending a packet finishes at time $s \in [0, t]$. We are then interested in the distribution of the residual backoff time $t - s$. We denote

this residual backoff time by $\bar{U}_0^{(k)}$, with pdf $f_{\bar{U}_0^{(k)}}(t)$ and cdf $F_{\bar{U}_0^{(k)}}(t)$. The density of $\bar{U}_0^{(k)}$ is given by

$$f_{\bar{U}_0^{(k)}}(t) = \frac{1 - F_{U_0^{(k)}}(t)}{\mathbb{E}U_0^{(k)}}, \quad t > 0, \quad 1 \le k \le n, \quad (1)$$

which is the well-known distribution of the residual backoff time [7][1].

For the throughput analysis we are interested in the idle time of the channel, i.e., the time in between the end of a transmission, and the start of the next one. If we assume, without loss of generality, that node 1 is the one finishing a transmission, then the idle time involves random variable $U_0^{(1)}$ (for node 1), and $\bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}$ for the remaining nodes. We are looking for the first backoff period to finish, i.e., the expectation of the minimum of these n random variables. The idle time of the channel, I_c, is then given by $I_c = \min\{U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}\}$. First we determine $F_{I_c}(t)$, the cdf of I_c, for $t > 0$:

$$\begin{aligned}
F_{I_c}(t) &= \mathbb{P}(I_c \le t) \\
&= \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}\} \le t) \\
&= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}\} \ge t) \\
&= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_0^{(1)}, \ldots, \bar{U}_0^{(1)}\} \ge t) \\
&= 1 - \mathbb{P}(U_0^{(1)} \ge t) \cdot \left(\mathbb{P}(\bar{U}_0^{(1)} \ge t)\right)^{n-1} \quad (2) \\
&= 1 - (1 - F_{U_0^{(1)}}(t)) \cdot (1 - F_{\bar{U}_0^{(1)}}(t))^{n-1} \\
&= 1 - \mathbb{E}U_0^{(1)} \cdot f_{\bar{U}_0^{(1)}}(t) \cdot (1 - F_{\bar{U}_0^{(1)}}(t))^{n-1}.
\end{aligned}$$

In the fifth equality in Eq. (2) we used independence of the random variables $U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}$, and in the last equality we substituted Eq. (1). The expectation of I_c can now be obtained by integrating the tail probabilities:

$$\begin{aligned}
\mathbb{E}I_c &= \int_0^{W_0-1} \mathbb{P}(I_c \ge t)dt \\
&= \int_0^{W_0-1} \left(1 - F_{I_c}(t)\right)dt \\
&= \int_0^{W_0-1} \mathbb{E}U_0^{(1)} f_{\bar{U}_0^{(1)}}(t) \cdot (1 - F_{\bar{U}_0^{(1)}}(t))^{n-1}dt \quad (3) \\
&= \frac{\mathbb{E}U_0^{(1)}}{n} \\
&= \frac{W_0 - 1}{2n},
\end{aligned}$$

where we used in that last equality that $U_0^{(1)}$ is uniform on $[0, W_0 - 1]$. Conform our expectation, with $n = 1$ the waiting time is half the initial backoff window W_0, and as $n \to \infty$ the waiting time tends to 0. The throughput S_c is now computed using

$$S_c = \frac{T}{T + \mathbb{E}I_c} = \frac{T}{T + \frac{W_0-1}{2n}}, \quad (4)$$

with T the number of time slots needed to transmit a single packet. Numerical experiments in Section 5 demonstrate

---

[1]For example, with $U_0^{(k)}$ uniformly distributed on $[0, W_0-1]$, this density becomes $f_{\bar{U}_0^{(k)}}(t) = \frac{1-t/(W_0-1)}{W_0-1} = \frac{W_0-1-t}{2}$.

that this expression does indeed capture the throughput accurately.

4. MULTI-LAYER ANALYSIS

When the assumption of a single layer is dropped, the situation becomes considerably more complex. At the end of the packet transmission we now no longer know the distribution of the remaining $n - 1$ nodes. For instance, after the packet transmission at node 1 in Fig. 2, node 2 has been through two backoff periods and is busy with at least its third backoff period. It may even be more than that, since the figure does not show what happened at node 2 prior to the backoff period of length $u_{2,1}$. Potentially, a throughput analysis of the multi-layer scenario involves a large and complex model including the behavior of individual nodes. Clearly, such models are intractable for larger networks with many nodes. In this section we provide a simple model for the throughput that allows us to overcome this issue.

Before continuing, we remind the reader that in a multi-layer scenario, the backoff window depends on the layer. To be precise, in layer x the window W_x is

$$W_x = W_0 \cdot 2^{\min(x,m)}. \quad (5)$$

So each time a node moves to the next backoff layer, the window is doubled, until it reaches layer m. At layer 0, the window is W_0, and at layer $x \ge m$ the window is $W_0 2^m$. See also the description of the CSMA-CA protocol in Section 2.1.

For now, suppose that at the end of a packet transmission, the $n - 1$ other nodes are all at the same layer, and denote this layer by x. Node 1, which just finished the transmission, is at layer 0. Following the notation of the previous section, we denote the backoff time at layer x for node k ($1 \le k \le n$) by $U_x^{(k)}$, and the corresponding remainder by $\bar{U}_x^{(k)}$. Then, we have $I_c(x) = \min\{U_x^{(1)}, \bar{U}_x^{(2)}, \ldots, \bar{U}_x^{(n)}\}$. Note that we changed notation from I_c to $I_c(x)$, reflecting the dependency on layer x. Repeating the steps of the previous section gives $F_{I_c(x)}(t)$ for $t > 0$:

$$\begin{aligned}
F_{I_c(x)}(t) &= \mathbb{P}(I_c(x) \le t) \\
&= \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_x^{(2)}, \ldots, \bar{U}_x^{(n)}\} \le t) \\
&= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_x^{(2)}, \ldots, \bar{U}_x^{(n)}\} \ge t) \\
&= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_x^{(1)}, \ldots, \bar{U}_x^{(1)}\} \ge t) \quad (6) \\
&= 1 - \mathbb{P}(U_0^{(1)} \ge t) \cdot \left(\mathbb{P}(\bar{U}_x^{(1)} \ge t)\right)^{n-1} \\
&= 1 - (1 - F_{U_0^{(1)}}(t)) \cdot (1 - F_{\bar{U}_x^{(1)}}(t))^{n-1} \\
&= 1 - \mathbb{E}U_0^{(1)} \cdot f_{\bar{U}_0^{(1)}}(t) \cdot (1 - F_{\bar{U}_x^{(1)}}(t))^{n-1}.
\end{aligned}$$

We can also compute $\mathbb{E}I_c(x)$ as before by integrating the tail probabilities via

$$\mathbb{E}I_c(x) = \int_0^{W_0-1} \mathbb{P}(I_c(x) \ge t)dt. \quad (7)$$

Here, we used that $U_0^{(1)}$ is uniform on $[0, W_0-1]$ to establish the interval over which to integrate. The resulting expression is cumbersome, so we omit the result here. The expression for throughput in Eq. (4) still holds, but we repeat it here with adapted notation to emphasize the dependence on x:

$$S_c(x) = \frac{T}{T + \mathbb{E}I_c(x)}. \quad (8)$$

We now focus our attention on the throughput analysis of a single node. Without loss of generality, we assume that this is node 1. Prior to a packet transmission, node 1 spent some time waiting as part of the CSMA-CA process. By assumption, we know that it is currently at layer x and thus we also know how much time node 1 spent waiting: the sum of the expected backoff time at layers $0, \ldots, x$. However, for reasons that become apparent later, we need a sensible interpretation of a layer number x that is non-integer. Therefore, suppose that $x = \lfloor x \rfloor + \alpha$, with $\alpha \in [0, 1)$ and $\lfloor x \rfloor$ the largest integer smaller than x. When a node is at a decimal layer x, we interpret this as it having to wait at all integer layers $0, \ldots, \lfloor x \rfloor$, plus a fraction α at the layer with backoff window $W_{\lfloor x \rfloor + \alpha}$. This interpretation is consistent with the integer view of layers when $\alpha = 0$ and when α tends to 1.

With this interpretation, we denote the waiting time on a node by $I_N(x)$ and calculate its expectation from

$$\mathbb{E}I_N(x) = \sum_{j=0}^{\lfloor x \rfloor} \mathbb{E}U_j^{(1)} + \alpha \mathbb{E}U_x^{(1)}. \tag{9}$$

We can expand the sum in Eq. (9) further, taking care that the doubling is stopped after layer m and that we do not know whether x is larger or smaller than m. With $U_x^{(1)}$ uniformly distributed on $[0, W_x - 1]$, some careful calculations yield

$$\begin{aligned}
\mathbb{E}I_N(x) &= -\frac{\lfloor x \rfloor + 1}{2} + W_0 \frac{2^{\min(\lfloor x \rfloor, m) + 1} - 1}{2} \\
&\quad + \frac{W_0 2^m}{2}(x - m)^+ + \alpha W_0 \frac{2^{\min(x, m)} - 1}{2},
\end{aligned} \tag{10}$$

where $(x)^+ = \max(0, x)$. Similar to Eq. (4), we can derive the throughput of one particular node as

$$S_N(x) = \frac{T}{T + \mathbb{E}I_N(x)}. \tag{11}$$

We now have an expression for the throughput on the channel from $S_c(x)$ in Eq. (8), and for the throughput provided by each node ($S_N(x)$ in Eq. (11)). Since all nodes are identical, each node will contribute an equal share to the throughput on the channel. Therefore, the following consistency relation should hold:

$$S_c(x) = n \cdot S_N(x). \tag{12}$$

Analyzing the saturation throughput is now done by calculating a value x such that Eq. (12) holds. The following lemma and definition formalize the process.

LEMMA 1. *The consistency relation in Eq. (12) has a unique solution x^*.*

PROOF. We begin the proof by inspecting Eq. (12) with $n = 1$, for which it reduces to $\mathbb{E}I_N(x) = \mathbb{E}I_c(x)$. For $\mathbb{E}I_c(x)$, Eq. (7) is the same as Eq. (3) so that $\mathbb{E}I_c(x) = (W_0 - 1)/2$. The same expression results from Eq. (10) if we calculate $\mathbb{E}I_N(0)$, and thus for $n = 1$ the natural layer is $x^* = 0$. This corresponds to intuition, since with a network containing 1 node, the channel is always free at the end of the backoff time at layer 0, and there is no need to go to higher layers.

To show uniqueness for the case with $n > 1$ we inspect the behavior at $x = 0$ and as $x \to \infty$. At $x = 0$ we have $\mathbb{E}I_c(0) = \frac{W_0 - 1}{2n}$ (again from Eq. (3)), and thus that

$$S_c(0) = \frac{T}{T + \frac{W_0 - 1}{2n}} = \frac{n \cdot T}{nT + \frac{W_0 - 1}{2}}.$$

From Eq. (10) we get $\mathbb{E}I_N(0) = \frac{W_0 - 1}{2}$ and thus

$$n \cdot S_N(0) = \frac{n \cdot T}{T + \frac{W_0 - 1}{2}}.$$

So at $x = 0$ we have $n \cdot S_N(0) > S_c(0)$ (since $n > 1$).

As $x \to \infty$, $\mathbb{E}I_N(x)$ tends to infinity linearly, and thus $S_N(x)$ tends to 0. However, $\mathbb{E}I_c(x)$ becomes constant as $x \to \infty$, because the doubling of W_x is stopped when $x > m$. Hence, $S_c(x)$ also tends to a constant and as $x \to \infty$ we have $n \cdot S_N(x) < S_c(x)$. Consequently, somewhere in the interval $[0, \infty)$, there is a unique $x = x^*$ where the monotonously decreasing $S_N(x)$ crosses the constant $S_c(x)$, so that $S_c(x^*) = n \cdot S_N(x^*)$. □

Definition We call the unique solution x^* to Eq. (12) the *natural layer*. Based on Eq. (10), the natural layer is interpreted as the expected amount of time that a node typically has to wait as part of the CSMA-CA process, prior to sending a packet.

Observe that there is no guarantee that the natural layer x^* is an integer, which is why we extended the interpretation of a layer to non-integer values. Also, in the proof of Lemma 1 we showed that for a network with $n = 1$ node our technique yields the correct throughput. When we look at large networks and let $n \to \infty$, we see that $F_{I_c(x)}(t) \to 1$, that $\mathbb{E}I_c(x) \to 0$, and thus that $S_c(x) \to 1$ (from Eqs. (6), (7), and (8), respectively). In other words, in large networks the expected idle time of the channel becomes 0, and the throughput rises to its maximum value of 1. This is indeed what we expect to see happening in our model for large networks.

In the next section we demonstrate that the throughput $S_c(x^*)$ closely resembles the results of simulations.

5. EXPERIMENTS

We validate the model described in the previous section by comparing it to the results obtained from a discrete event simulation of the CSMA-CA process. Finding the throughput using our model is done by numerically finding the natural layer x^* for which Eq. (12) holds. Once x^* is found, we use Eq. (8) to calculate $S_c(x^*)$.

In Fig. 3 we compare the throughput S_c obtained from our model (lines), to the results of the discrete event simulations (markers). The figure shows the throughput for varying number of nodes n, and several IEEE 802.15.4 parameter settings (for easy notation we report W_0 and W_m instead of the corresponding parameter values for $macMinBE$ and $macMaxBE$). The values from the analysis closely match those of the simulations, demonstrating that our analysis accurately captures the throughput. Also, as n increases the throughput tends to 1 for all parameter settings.

Note that with $n = 1$ the natural layer is always $x^* = 0$ and the multi-layer analysis in Section 4 should match the result of the single-layer analysis in Section 3. Substituting $n = 1$ in Eq. (4) yields the values in Table 2, which nicely match the left-most markers in Fig. 3.

Next, we inspect the natural layer numbers corresponding to the graphs in Fig. 3. These are plotted in Fig. 4, where for small n the lines show a slight curvature, and as n increases they suggest a linear increase in the natural layer. These effects are due to the MAC protocol stopping the doubling
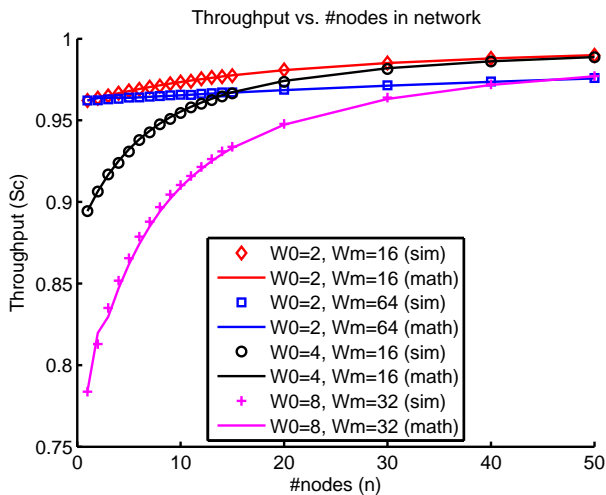
Figure 3: Throughput S_c as computed via the natural layer (solid line), and as obtained from simulations (markers), for varying number of nodes in the network (n).

W_0	W_m	S_c	W_0	W_m	S_c
2	16	0.96	4	16	0.89
2	64	0.96	8	32	0.78

Table 2: Throughput according to the single-layer analysis from Eq. (4) for the parameter values used in Fig. 3.

of the backoff window after layer m. We expect that with a deeper analysis we are able to explain the effects in detail.

Figs. 3-4 demonstrate that despite its simplicity the model leads to an accurate prediction of the throughput for a wide range of protocol parameter settings.

6. DISCUSSION

In this section we make several remarks relevant to the throughput model discussed in this paper.

Comparison to 802.11. Many papers investigating saturation throughput are based on the seminal paper by Bianchi [6], who formulates a model for the throughput of a WLAN network as specified in the IEEE 802.11 standard. The MAC protocols of IEEE 802.11 and IEEE 802.15.4 (which is studied in the present paper) are quite similar, except for a property called *freezing*. In 802.11, a node that is backing off does a channel assessment at the end of each time slot, to see if the channel is busy. If the channel is busy, the backoff process is paused until the channel is free again. So during a transmission, all non-sending nodes are completely idle and are not backing off. This 'freezing' feature, which is absent in IEEE 802.15.4, has significant impact on the throughput performance of 802.11. Specifically, consider the three event types on the channel identified by Bianchi: a successful transmission, a collision between two or more packages, and a backoff event. The probability of these events is easy to derive from the two-dimensional discrete-time Markov chain (defined in [6]) that describes the evolution of the backoff state (i, k),
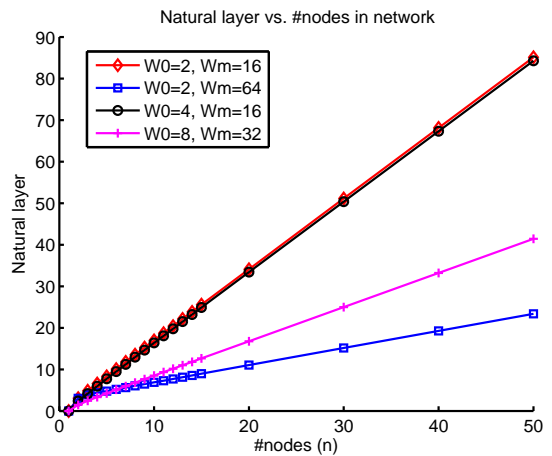

Figure 4: The natural layer x^* as computed in our throughput analysis for varying number of nodes in the network (n).

where i is the retransmission counter and k is the backoff counter. For the IEEE 802.15.4 protocol the absence of freezing implies that three event types used in Bianchi can no longer be defined in the same way. A solution for this is to extend the Markov chain to include the duration of transmissions, as done in [2] and [4]. This solution, however, requires assuming that T is an integer (which it is not by default) and adds further complexity to the Markov chain. As an alternative, in the present paper we propose a different approach by introducing the concept of a natural layer, allowing us to formulate a much simpler model for the throughput. Our model does not use a Markov chain, and is valid for non-integer values of T.

Collisions. A collision between packets occurs when two or more nodes finish backing off at the same time slot, see the channel idle, and consequently transmit a packet simultaneously. In the scenario with continuously distributed backoff times (as we assume in this paper), the probability of two or more nodes finishing a backoff period at the same time is 0. However, when we use a discrete distribution for the backoff times (as used in IEEE 802.15.4), collisions do seem possible: if two nodes draw the same (discrete) backoff time and start the backoff process at the same time, then they cause a collision.

There are, however, two factors in our experiments that make it unlikely that the two backoff processes start simultaneously. First, our discrete event simulation waits a random amount of time before processing the first packet. This amount of time is drawn from a continuous uniform distribution, thereby preventing the backoff process at the nodes to start simultaneously. In practice, such a precaution is advised as well. Second, a packet transmission is equivalent to a non-integer number of backoff steps.

To verify these observations, we change the distribution of the backoff time to a discrete uniform distribution, run simulations again, and record the number of packets that are involved in a collision during the simulation. We then compute the *collision rate* by dividing the total number of collided packets by the number of packets for which trans-
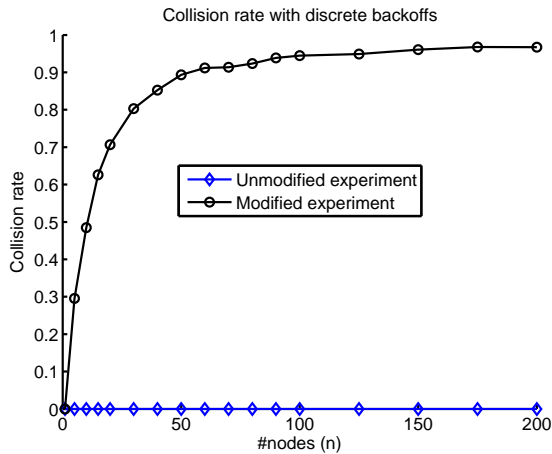
Figure 5: Collision rate when a discrete backoff time distribution is used in simulations. The experiment shows a zero collision rate (blue diamonds) because (1) a random waiting time is used prior to sending the first packet, and (2) a packet transmission is equivalent to a non-integer number of backoff steps. When the first is disabled, and the second modified to an integer, the collision rate is high and steadily approaches 1 as the number of nodes increases (black circles). Both experiments were done using $W_0 = 8, W_m = 32$ (the 802.15.4 default values).

Figure 6: Throughput S_c via the natural layer (solid line), and simulations (markers), for decreasing packet size.

mission was attempted. The result is plotted in Fig. 5 (blue diamonds), and demonstrates that the collision rate is equal to 0, even for a large number of nodes. Next, we disable the random waiting time that is used before processing the first packet, and modify the packet size such that a transmission takes an integer number of backoff steps. Fig. 5 shows a collision rate that approaches 1 as the number of nodes n in the network increases (black circles). This experiment demonstrates that the inclusion of a random waiting time prior to the first packet transmission and of the non-integer number of backoff steps needed for a packet transmission, effectively prevent collisions.

Avoiding long waiting times. Fig. 3 demonstrates that our model for the throughput accurately captures the throughput recorded in a discrete event simulation. However, a close look at the line corresponding to parameter values $W_0 = 8, W_m = 32$ (the IEEE 802.15.4 defaults) suggests a slight irregularity for our model at $n = 2$. This irregularity is exaggerated in Fig. 6, where we decrease the packet size from 1,250 to 250 bits and fix $W_0 = 8, W_m = 32$. For a small packets size, our model only captures throughput well for a large number of nodes.

The irregularity is due to a small packet size as compared to the backoff times. For example, suppose node 0 starts a packet transmission and the other $n-1$ nodes are backing off. Node 0 then transmits the packet, and draws a new backoff time from interval $[0, W_0 - 1]$. If the sum of this transmission time and backoff time is smaller than the residual backoff times at the other $n-1$ nodes, node 0 also transmits the next packet. Hence, in a scenario where the packet size is small and the residual backoff times are large, it is likely
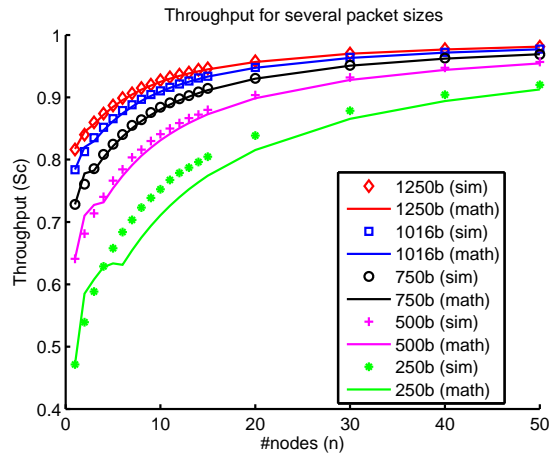
that several consecutive transmissions occur at node 0. We observed the tendency for consecutive transmissions in the discrete event simulation as well. The irregularity vanishes for increasing n (the number of nodes), since then the minimum of the residual backoff times at the other $n-1$ nodes decreases.

Our model assumes, in Eq. (6), that the random variables for the residual backoff time $\bar{U}_{x*}^{(k)}$ of the $n-1$ waiting nodes are independent. In the situation described above, this assumption fails and our model no longer captures the throughput well. This is, however, not a severe restriction on our model: if a network operator expects mainly small packets, he has the option to choose appropriately small values for the window sizes, thereby avoiding situations with long waiting times. Our model can be used to find values for the protocol parameters such that waiting times are acceptable.

Near insensitivity to the backoff time distribution. Our throughput model is also valid for non-uniform backoff time distribution. Section 4 is written for general $U_x^{(k)}$, and only requires a change to the lower and upper bound of the integral in Eq. (7) if the distribution has a domain different from $[0, W_0]$. On the node level, Eqs. (9) and (11) remain valid, as does the consistency relation Eq. (12).

Enabling acknowledgements. In Section 2.2, we assume for modeling purposes that acknowledgements are disabled. To enable these, we need to extend our model with additional CSMA-CA process parameters. In particular, this includes the time needed by the sink to switch from reception to transmission, and the time a node needs to do a channel assessment. Interestingly, there is some disagreement in the literature concerning the latter parameter. In [4] and [5] the authors model two channel assessments for the sensor nodes, whereas in [2] and [3] the authors model only one. The section in the IEEE 802.15.4 standard describing the CSMA-CA process also does not say explicitly how many channel assessments should be done. In future work, we will investigate these issues in order to include

acknowledgements in our model.

7. CONCLUSION AND FUTURE WORK

In this paper we presented a simple yet powerful method for analyzing the throughput of a network of sensor nodes running the IEEE 802.15.4 MAC protocol. After a description of the protocol, we introduced the concept of a *natural layer*. This concept allowed us to analyze the waiting time involved in the MAC protocol. Then, we formulated a model for the throughput on the channel, and a model for the contribution to the throughput of a single node. Combining these two resulted in an equation from which we numerically computed the natural layer, which in turn gave the throughput. The model was validated with experiments from a discrete event simulation, and demonstrated that our model accurately captures the throughput from the simulations.

In future work, we plan to extend the model by relaxing the assumptions made in Section 2.2. We intend to include non-saturated conditions (assumption 2), a maximum number of layers (assumption 4), and acknowledgements (assumption 5). In these steps we will continue to take potential collisions into account, via, e.g., explicit modeling or suitable changes to the experiment. Additionally, having an accurate model for the throughput allows us to analyze other quality of service aspects of the sensor network. For instance, the natural layer can be used to inspect the delay of a packet prior to being sent.

Acknowledgements

8. REFERENCES

[1] B. Latré, P. Mil, I. Moerman, B. Dhoedt, P. Demeester, and N. v. Dierdonck, "Throughput and delay analysis of unslotted IEEE 802.15. 4," *Journal of Networks*, vol. 1, no. 1, pp. 20–28, 2006.

[2] P. D. Marco, P. Park, C. Fischione, and K. H. Johansson, "Analytical modeling of multi-hop IEEE 802.15.4 networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3191–3208, 2012.

[3] B. Lauwens, B. Scheers, and A. v. d. Capelle, "Performance analysis of unslotted CSMA/CA in wireless networks," *Telecommunication Systems*, vol. 44, no. 1-2, pp. 109–123, 2010.

[4] S. Pollin, M. Ergen, S. Ergen, B. Bougard, L. 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 Transactions on Wireless Communications*, vol. 7, no. 9, pp. 3359–3371, September 2008.

[5] C. Y. Lee, H. I. Cho, G. U. Hwang, Y. Doh, and N. Park, "Performance modeling and analysis of IEEE 802.15. 4 slotted CSMA/CA protocol with ACK mode," *AEU-International Journal of Electronics and Communications*, vol. 65, no. 2, pp. 123–131, 2011.

[6] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[7] I. Adan and J. Resing, "Queueing theory," http://wwwhome.math.utwente.nl/~scheinhardtwrw/queueingdictaat.pdf, 2001.

[8] SURFsara, "http://www.surfsara.nl," 2013.