Networks of Sensors

Onderwater, M.

2016

4

Throughput Modeling of the IEEE MAC for Sensor Networks

In this chapter we focus on the 'network' aspect of a 'sensor network'. The increasing number of sensors in a network result in large number of measurements that have to be transmitted to, e.g., a middleware component for further processing, thereby stressing the processing capabilities of the network. We consider a specific performance indicator of a network, namely the *saturation throughput*. This property reflects at what speed the network is able to process measurements by sensors when a large number of these measurements is offered. 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 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.

This chapter is based on the results presented in [9].

4.1 Introduction

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. Without this mechanism, each node in the network would continuously attempt transmissions, causing massive loss of packets and large periods of inactivity on the network. However, the waiting time enforced by the CSMA-CA mechanism might decrease the throughput of the network significantly 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. This MAC protocol can be used in two different configurations: *slotted* and *unslotted*. The unslotted configuration is the simplest version, whereas the slotted protocol has richer features. Both configurations appear in the literature, and we review the state of the art below. The authors of [83] 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. [102] looks at the unslotted MAC in more detail and formulates a three dimensional Markov Chain for the CSMA-CA process. From this, expressions for link reliability, packet delay, and energy consumption are derived. The results are valid for both a network in star formation, as well as for a general multi-hop network. For large-scale networks an approximate model is constructed, in order to keep computations numerically tractable. Simulations are used to validate the model. [85] analyzes throughput for the unslotted MAC by combining a renewal process for the physical layer with a semi-Markov process for the MAC layer. The analysis results in equations that are solved via a fixed-point procedure, and the resulting throughput closely resembles values observed in a discrete event simulation.

For the slotted MAC, [126] is similar to [102]. The authors investigate throughput by constructing a two-dimensional Markov Chain, and derive an expression for the throughput. They then compare the results of the model with the outcome of simulations, and demonstrate that their model accurately captures the throughput. In [89], Lee et al 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 chapter is to provide a simple yet accurate model for analyzing the throughput. 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 4.2.1). Using this concept, we develop a simple model for the throughput, and use simulation results to demonstrate that it is accurate for a wide range of realistic parameter settings. In our model, we focus on the unslotted version of the MAC protocol. The model provides 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 chapter is as follows. In Section 4.2 we outline the IEEE 802.15.4 CSMA-CA protocol, list model assumptions and notation for our analysis. Then, in Section 4.3 we derive an expression for the throughput of sensor networks with a single backoff layer. Subsequently, in Section 4.4 we introduce the concept of a natural layer and use this to extend the model to a setting with multiple layers. In Section 4.5 we show simulation results to demonstrate that the model captures the throughput accurately for a wide range of parameter settings. Next, in Section 4.6 we discuss key differences between the IEEE 802.15.4 and 802.11 standards, and how these differences influence modeling of the throughput. Finally, Section 4.7 contains concluding remarks and ideas for future research.

4.2 Preliminaries

In this section we briefly outline the IEEE 802.15.4 CSMA-CA protocol, because a good understanding of this mechanism is essential when modeling throughput. Additionally, we list the model assumptions and provide some preliminary remarks.

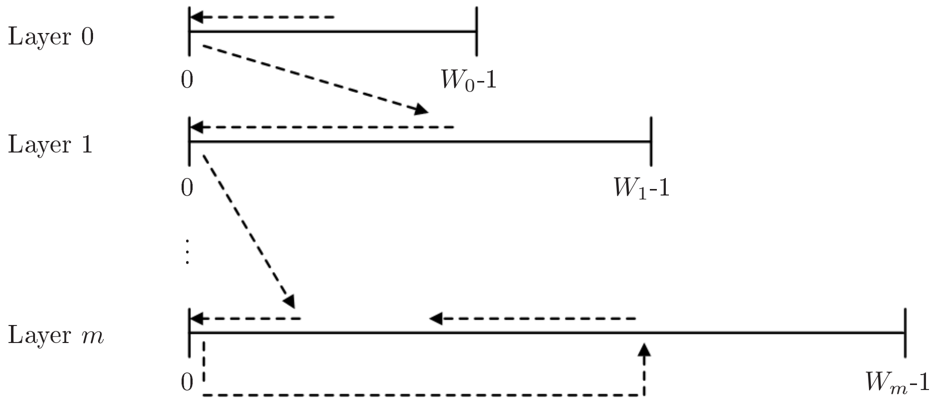

FIGURE 4.1: The CSMA-CA mechanism applied by each node in the network.

4.2.1 The IEEE 802.15.4 CSMA-CA protocol

The CSMA-CA mechanism states that, in order to avoid collisions, a node should wait a random amount of time (known as the *backoff 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 backoff 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.

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

Figure 4.1 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.

4.2.2 Assumptions

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

- Sensor nodes are structured as a star network, and all nodes send packets to the sink node.
- 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.
- The network uses non-beacon mode, and the unslotted version of the CSMA-CA mechanism.
- Acknowledgements are disabled.
- If two nodes finish a backoff cycle simultaneously, only one of the packets is transmitted. The other packet moves to the next backoff layer. Hence, there are no packet collisions. In Section 4.6 we revisit this topic.
- Packets go through the CSMA-CA mechanism until they are sent.

4.2.3 Preliminary remarks

Notation. The IEEE 802.15.4 MAC variables *macMinBE* and *macMaxBE* are cumbersome in a mathematical analysis, so we use a different notation in the remainder of this chapter. By W_0 we denote the backoff window for layer 0, i.e., $W_0 = 2^{macMinBE}$. We use m instead of "*macMaxBE-macMinBE*" to indicate how often layer 0 is doubled in size. Finally, T is the number of time slots (see below) required to send one packet.

Continuous backoff time. For this chapter, 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 chapter; 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. 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 chapter, our analysis works for other choices as well. To preserve this neutrality, we use the time slots from the CSMA-CA process as the unit of time throughout this chapter. These time slots are configuration- and region-neutral, and can easily be converted to seconds if needed (this 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.

4.3 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 it uses just one layer, layer 0. This simplified scenario forms an introduction to the complete throughput analysis, later in this chapter. Figure 4.2 shows n nodes going through the CSMA-CA process of transmitting packets (marked by T) and backing off (denoted by $u_{1,1}, \ldots, u_{n,2}$). 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 cycles. In particular, node 2 starts three backoff cycles (of length $u_{2,1}, u_{2,2}, u_{2,3}$, respectively), and node n starts two cycles (of length $u_{n,1}$ and $u_{n,2}$). At the end of the transmission at node 1, this node also starts a backoff cycle (of length $u_{1,1}$). The first backoff cycle 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.

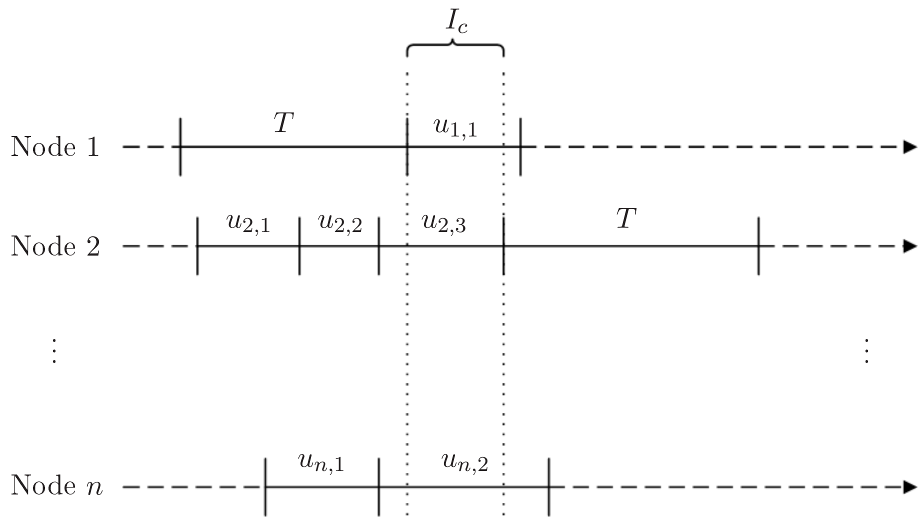

Figure 4.2: Events in the csma-ca process in between two transmissions (marked by T). The interval lengths $u_{1,1}, \ldots, u_{n,2}$ are backoff times drawn from the uniform distribution on interval $[0, W_0 - 1]$.

For determining throughput, we analyze the time that the channel is idle in between two transmissions. In Figure 4.2, 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 cycle. Node 1, in contrast, starts a new backoff cycle. 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 cycle at node 1, and using the distribution of the residual of the backoff cycles at the other nodes.

To start the analysis we introduce some notation. $U_0^{(k)}$ 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 cycle of length t and the node that is currently sending a packet finishes at time $s \in [0, t]$. We are 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

$$f_{\bar{U}_0^{(k)}}(t) = \frac{1 - F_{U_0^{(k)}}(t)}{\mathbb{E}U_0^{(k)}}, \quad t > 0, \quad 1 \leq k \leq n, \tag{4.1}$$

which is the well-known distribution of the residual backoff time [13]. 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 cycle 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 $I_c = \min\{U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}\}$. We determine $F_{I_c}(t)$, the cdf of I_c, for $t > 0$ via:

$$\begin{aligned} F_{I_c}(t) &= \mathbb{P}(I_c \leq t) \\ &= \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}\} \leq t) \\ &= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_0^{(2)}, \ldots, \bar{U}_0^{(n)}\} \geq t) \\ &= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_0^{(1)}, \ldots, \bar{U}_0^{(1)}\} \geq t) \\ &= 1 - \mathbb{P}(U_0^{(1)} \geq t) \cdot \big(\mathbb{P}(\bar{U}_0^{(1)} \geq t)\big)^{n-1} \\ &= 1 - (1 - F_{U_0^{(1)}}(t)) \cdot (1 - F_{\bar{U}_0^{(1)}}(t))^{n-1} \\ &= 1 - \mathbb{E}U_0^{(1)} \cdot f_{\bar{U}_0^{(1)}}(t) \cdot (1 - F_{\bar{U}_0^{(1)}}(t))^{n-1}. \end{aligned} \tag{4.2}$$

In the fifth equality in Eq. (4.2) 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. (4.1). The expectation of I_c can now be obtained by integrating the tail probabilities:

$$\begin{aligned} \mathbb{E}I_c &= \int_0^{W_0-1} \mathbb{P}(I_c \geq t)dt \\ &= \int_0^{W_0-1} \big(1 - F_{I_c}(t)\big)dt \\ &= \int_0^{W_0-1} \mathbb{E}U_0^{(1)} f_{\bar{U}_0^{(1)}}(t) \cdot (1 - F_{\bar{U}_0^{(1)}}(t))^{n-1}dt \\ &= \frac{\mathbb{E}U_0^{(1)}}{n} \\ &= \frac{W_0 - 1}{2n}, \end{aligned} \tag{4.3}$$

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

$$S_c = \frac{T}{T + \mathbb{E}I_c} = \frac{T}{T + \frac{W_0-1}{2n}}, \tag{4.4}$$

with T the number of time slots needed to transmit a single packet. Numerical experiments in Section 4.5 demonstrate that this expression does indeed capture the throughput accurately.

4.4 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 Figure 4.2, node 2 has been through two backoff cycles and is busy with at least its third backoff cycle. It might even be more than that, since the figure does not show what happened at node 2 prior to the cycle 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 recall that in a multi-layer scenario, the backoff window depends on the layer. To be precise, in layer x the window W_x is

$$W_x = W_0 \cdot 2^{\min(x,m)}. \tag{4.5}$$

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 $x \geq m$ the window is $W_0 2^m$. See also the description of the CSMA-CA protocol in Section 4.2.1. 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 ($1 \leq k \leq n$) by $U_x^{(k)}$, and the corresponding remainder by $\bar{U}_x^{(k)}$. Then, we have $I_c(x) = \min\{U_0^{(1)}, \bar{U}_x^{(2)}, \ldots, \bar{U}_x^{(n)}\}$. 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 $t > 0$:

$$
\begin{aligned}
F_{I_c(x)}(t) &= \mathbb{P}(I_c(x) \leq t) \\
&= \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_x^{(2)}, \ldots, \bar{U}_x^{(n)}\} \leq t) \\
&= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_x^{(2)}, \ldots, \bar{U}_x^{(n)}\} \geq t) \\
&= 1 - \mathbb{P}(\min\{U_0^{(1)}, \bar{U}_x^{(1)}, \ldots, \bar{U}_x^{(1)}\} \geq t) \\
&= 1 - \mathbb{P}(U_0^{(1)} \geq t) \cdot \left(\mathbb{P}(\bar{U}_x^{(1)} \geq t)\right)^{n-1} \\
&= 1 - (1 - F_{U_0^{(1)}}(t)) \cdot (1 - F_{\bar{U}_x^{(1)}}(t))^{n-1} \\
&= 1 - \mathbb{E}U_0^{(1)} \cdot f_{\bar{U}_0^{(1)}}(t) \cdot (1 - F_{\bar{U}_x^{(1)}}(t))^{n-1}.
\end{aligned}
\tag{4.6}
$$

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

$$\mathbb{E}I_c(x) = \int_0^{W_0 - 1} \mathbb{P}(I_c(x) \geq t) dt. \tag{4.7}$$

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 for $\mathbb{E}I_c(x)$ requires several pages to display, so we omit it here for compactness. The expression for throughput in Eq. (4.4) still holds, but we repeat it here with adapted notation to emphasize the dependence on x:

$$S_c(x) = \frac{T}{T + \mathbb{E}I_c(x)}. \tag{4.8}$$

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. To this end, 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

$$\mathbb{E}I_N(x) = \sum_{j=0}^{\lfloor x \rfloor} \mathbb{E}U_j^{(1)} + \alpha \mathbb{E}U_x^{(1)}. \tag{4.9}$$

We can expand the sum in Eq. (4.9) 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

$$\begin{aligned} \mathbb{E}I_N(x) &= -\tfrac{\lfloor x \rfloor + 1}{2} + W_0 \tfrac{2^{\min(\lfloor x \rfloor, m)+1} - 1}{2} \\ &+ \tfrac{W_0 2^m}{2}(\lfloor x \rfloor - m)^+ + \alpha W_0 \tfrac{2^{\min(x,m)} - 1}{2}, \end{aligned} \tag{4.10}$$

where $(x)^+ = \max(0, x)$. Similar to Eq. (4.4), we can find the throughput of one particular node using

$$S_N(x) = \frac{T}{T + \mathbb{E}I_N(x)}. \tag{4.11}$$

We now have an expression for the throughput on the channel from $S_c(x)$ in Eq. (4.8), and for the throughput provided by each node ($S_N(x)$ in Eq. (4.11)). In a fair star network, all nodes are identical and each contributes an equal share to the throughput on the channel. Therefore, the following consistency relation should hold:

$$S_c(x) = n \cdot S_N(x). \tag{4.12}$$

Analyzing the saturation throughput is now done by calculating a value x such that Eq. (4.12) holds.

In Figure 4.3, the throughput expressions for $n \cdot S_N(x)$ and $S_c(x)$ are plotted for a network with $n = 10$ nodes. We see that for $x = 0$, $n \cdot S_N(x) > S_c(x)$ and that $n \cdot S_N(x)$ decreases to 0 as x increases. $S_c(x)$, on the other hand, becomes constant as x increases, and the two lines intersect at the dotted line. We are looking for the value of x for which this intersection occurs (denoted by x^*). The following lemma shows that x^* exists and is unique.

Lemma 4.4.1. *The consistency relation in Eq. (4.12) has a unique solution x^*.*

Proof. We begin the proof by inspecting Eq. (4.12) 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. (4.7) is the same as Eq. (4.3) so that $\mathbb{E}I_c(x) = (W_0 - 1)/2$. The same expression results from Eq. (4.10) 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.

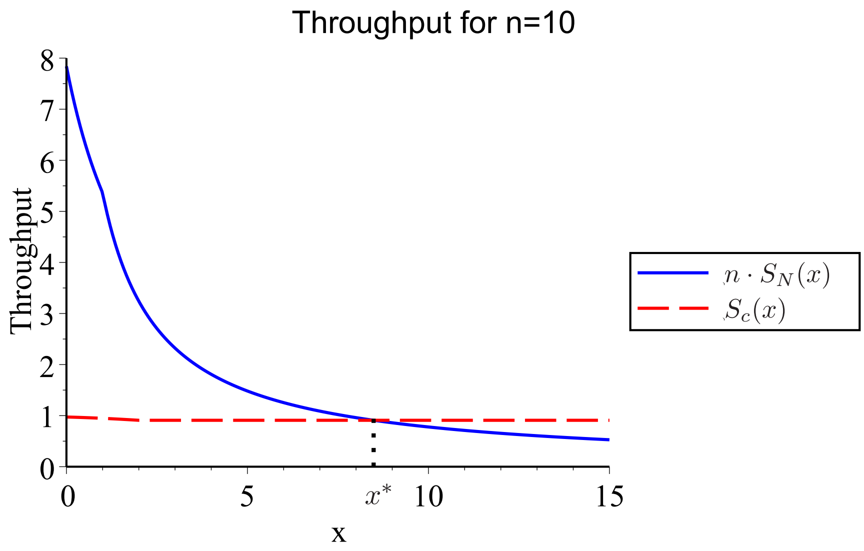

FIGURE 4.3: Throughput $n \cdot S_N(x)$ and $S_c(x)$ for a network with $n = 10$ nodes. $S_N(x)$ decreases as x increases, whereas $S_c(x)$ becomes constant. The x value at which the two lines intersect is the natural layer x^*.

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. (4.3)), and thus

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

From Eq. (4.10) 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, by the Intermediate Value Theorem [123], somewhere in the interval $(0, \infty)$, there is a unique $x = x^*$ where the monotonously decreasing $n \cdot S_N(x)$ crosses the constant $S_c(x)$, so that we have $S_c(x^*) = n \cdot S_N(x^*)$.

□

Definition We call the unique solution x^* to Eq. (4.12) the *natural layer*. Based on Eq. (4.10), 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. In the next section we demonstrate that the throughput $S_c(x^*)$ closely resembles the results of simulations.

4.5 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. (4.12) holds. Once x^* is found, we use Eq. (4.8) to calculate $S_c(x^*)$.

In Figure 4.4 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. This is as we expected, since we assumed in Section 4.2.2 that there are no collisions between packets that simultaneously finish a backoff cycle. We revisit the topic of collisions in the next section.

Note that with $n = 1$ the natural layer is always $x^* = 0$ and the multi-layer analysis in Section 4.4 should match the result of the single-layer analysis in Section 4.3. Substituting $n = 1$ in Eq. (4.4) yields the values in Table 4.1, which nicely match the left-most markers in Figure 4.4. Next, we inspect the natural layer numbers corresponding to the graphs in Figure 4.4. These are plotted in Figure 4.5, 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 of the backoff window after layer m. We expect that with a deeper analysis we are able to explain the effects in detail.

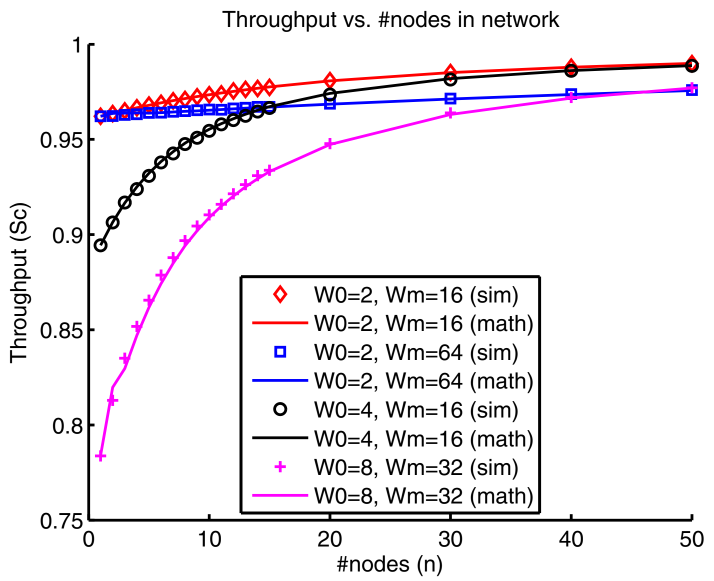

FIGURE 4.4: Throughput S_c as computed via the natural layer (solid line), and as obtained from simulations (markers), for varying number of nodes in the network (n).

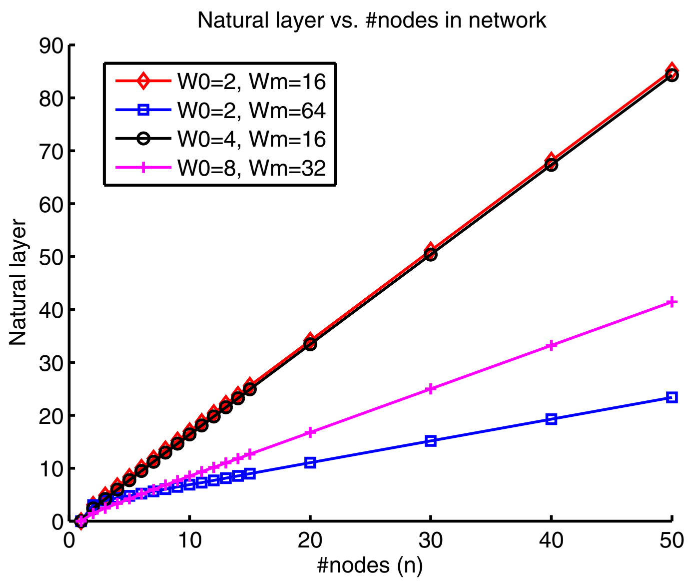

FIGURE 4.5: The natural layer x^* as computed in our throughput analysis for varying number of nodes in the network (n).

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

TABLE 4.1: Throughput according to the single-layer analysis from Eq. (4.4) for the parameter values used in Figure 4.4.

Figures. 4.4-4.5 demonstrate that despite its simplicity the model leads to an accurate prediction of the throughput for a wide range of protocol parameter settings. In the next section we make several remarks relevant to the throughput model discussed in this chapter.

4.6 Discussion

Comparison to 802.11. Many papers investigating saturation throughput are based on the paper by Bianchi [29], 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 are highly 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 idle and 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 [29]) that describes the evolution of the backoff state (i, k), 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 the Markov chain has to be extended to include the duration of transmissions (as done in, e.g., [126]). As an alternative, in the present chapter we propose a different approach by introducing the concept of a natural layer, allowing us to consider a much simpler, single-layered model for the throughput.

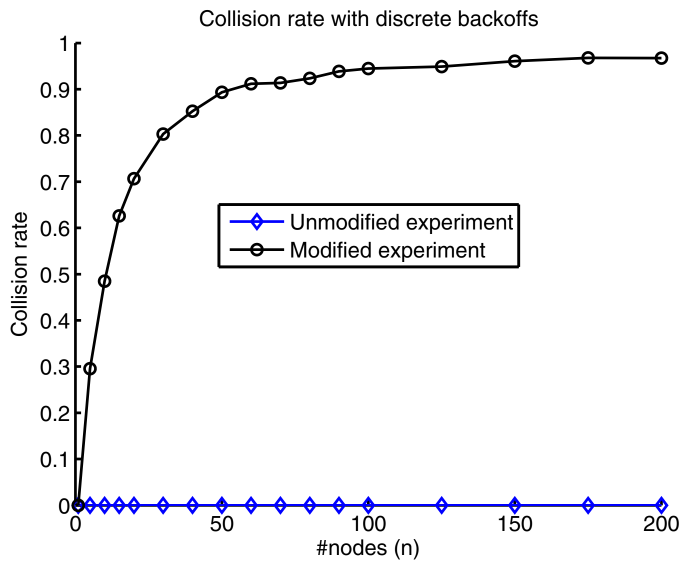

FIGURE 4.6: Collision rate when a discrete backoff time distribution is used in simulations. The experiment shows a zero collision rate (diamonds) because (1) a random waiting time is used prior to sending the first packet, and (2) a packet transmission is equivalent to a non-integer number of backoff 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 (circles).

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 chapter), it is highly unlikely that two or more nodes finish backing off at the same time and cause a collision. At first glance, this event seems more likely in the scenario with a discrete distribution for the backoff times (as used in IEEE 802.15.4). If two nodes draw the same (discrete) backoff time and start the backoff process at the same time, then they potentially cause a collision.

There are, however, two factors 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 this observation about collisions, we change the distribution of the backoff time to a discrete uniform distribution, run simulations again, and record the collision rate. The result is plotted in Figure 4.6 (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. Figure 4.6 shows a collision rate that approaches 1 as the number of nodes n in the network increases (black circles). Hence, with the disabled waiting time and the modified packet size, most transmissions cause a collisions as expected. 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. Figure 4.4 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 Figure 4.7, where we decrease the packet size from 1,250 to 250 bits and fix $W_0 = 8, W_m = 32$. For small packet sizes, our model only captures throughput well for large n.

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, since then the minimum of the residual backoff times at the other $n-1$ nodes decreases.

Our model assumes, in Eq. (4.6), 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

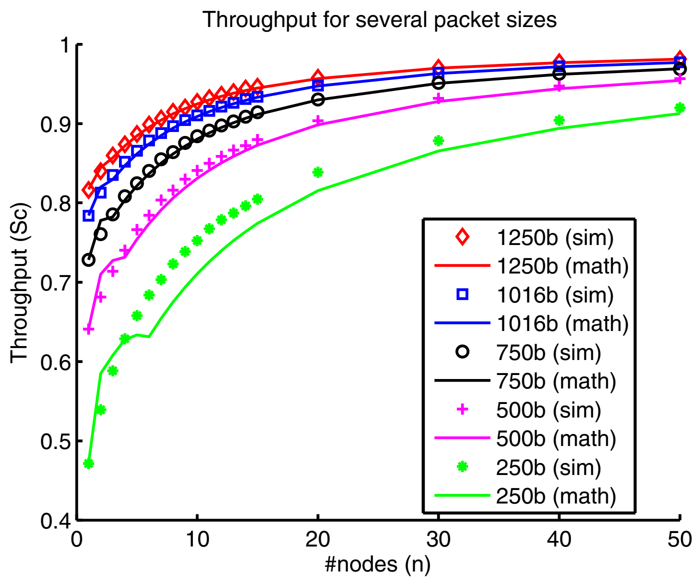

FIGURE 4.7: Throughput S_c via the natural layer (solid line), and simulations (markers), for decreasing packet size.

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 4.4 is written for general $U_x^{(k)}$, and only requires a change to the lower and upper bound of the integral in Eq. (4.7) if the distribution has a domain different from $[0, W_0]$. On the node level, Eqs. (4.9) and (4.11) remain valid, as does the consistency relation Eq. (4.12).

4.7 Conclusion

In this chapter 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. We introduced the concept of a *natural layer* which 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.

Future work includes adding more features of the MAC protocol, particularly acknowledgements and a maximum number of layers in the CSMA-CA process. Central to this research will be analyzing how much extra idle time these aspects cause in the wireless channel, and how they influence the natural layer.