# Performance Analysis of Slotted Carrier Sense IEEE 802.15.4 Medium Access Layer

Sofie Pollin, *Student Member, IEEE,* Mustafa Ergen, Sinem Coleri Ergen, Bruno Bougard
Liesbet Van der Perre, Ingrid Moerman, Ahmad Bahai, *Member, IEEE,*
Pravin Varaiya and Francky Catthoor, *Fellow, IEEE*

**Abstract**

Advances in low-power and low-cost sensor networks have led to solutions mature enough for use in a broad range of applications varying from health monitoring to building surveillance. The development of those applications has been stimulated by the finalization of the IEEE 802.15.4 standard, which defines the medium access control (MAC) and physical layer for sensor networks. One of the MAC schemes proposed is slotted Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), and this paper analyzes whether this scheme meets the design constraints of those low-power and low-cost sensor networks. The paper provides a detailed analytical evaluation of its performance in a star topology network, for uplink and acknowledged uplink traffic. Both saturated and unsaturated periodic traffic scenarios are considered. The form of the analysis is similar to that of Bianchi for IEEE 802.11 DCF only in the use of a per user Markov model to capture the state of each user at each moment in time. The key assumptions to enable this important simplification and the coupling of the per user Markov models are however different, as a result of the very different designs of the 802.15.4 and 802.11 carrier sensing mechanisms. The performance predicted by the analytical model is very close to that obtained by simulation. Throughput and energy consumption analysis is then performed by using the model for a range of scenarios. Some design guidelines are derived to set the 802.15.4 parameters as function of the network requirements.

Manuscript received March 1, 2006. Part of the material in this paper is submitted to IEEE Globecom 2006.
Sofie Pollin is with IMEC and K.U.Leuven, Belgium (phone: +3216288750; fax: +3216281515; e-mail: pollins@imec.be).
Mustafa Ergen is with UC Berkeley, USA (e-mail:ergen@cal.berkeley.edu).
Sinem Coleri Ergen is with UC Berkeley, USA (e-mail:csinem@eecs.berkeley.edu).
Bruno Bougard is with IMEC and K.U.Leuven, Belgium (e-mail: bougardb@imec.be).
Liesbet Van der Perre is with IMEC, Belgium (e-mail: vdperre@imec.be).
Francky Catthoor is with IMEC and K.U.Leuven, Belgium (e-mail: catthoor@imec.be).
Ingrid Moerman is with UGent, Belgium (e-mail: ingrid.moerman@intec.ugent.be).
Ahmad Bahai is with UC Berkeley, USA (e-mail:bahai@eecs.berkeley.edu).
Pravin Varaiya is with UC Berkeley, USA (e-mail:varaiya@eecs.berkeley.edu).
Research supported by IBBT WBA and National Semiconductor and ARO-MURI UCSC-W911NF-05-1-0246-VA-09/05.

# I. INTRODUCTION

Wireless sensor networks are autonomous networks for monitoring purpose, ranging from short-range, potentially *in vivo* health monitoring, to wide-range environmental surveillance. Despite the huge variety of their potential applications, all sensor networks are severely constrained in terms of power consumption. Sensor nodes are small form factor battery powered devices and size constraints limit the battery capacity. In most cases, the density of the network or the vaste environment where they are deployed prohibits a periodic replacement of the batteries. This makes energy efficiency a very important design requirement for those networks.

In a general way, the task of a sensor network consists of measuring a variable through the sensors, eventually (pre-)processing this information, and if opportune, transmitting the data to a data sink. It has been shown in varies design cases [1] that some of the most power hungry tasks of sensors are related to the communication: not only transmission and receive power, but the power needed while waiting (idle) and scanning the channel can be significant.

To address these requirements, the IEEE 802.15.4 standard which specifies the network's medium access control (MAC) and physical (PHY) layer, has been developed [2], [3], [4]. In IEEE 802.15.4 sensor networks a central controller, called the PAN (personal area network) coordinator, builds the network in its personal operating space. The standard supports three networking topologies relevant to sensor networking applications: star, peer-to-peer and cluster-tree. Since most sensor network applications involve monitoring tasks and reporting towards a central sink, and since the focus of this paper is on the 802.15.4. medium access control analysis, we focus on a one-hop star network. Instantiation of the model for other types of networks is also possible.

The channel access schemes are designed to save energy, so mechanisms are included to allow nodes to switch to low-power states and avoid expensive modes such as *transmission*, *reception* and *channel listening*. 802.15.4 compliant hardware has been designed, with very lower power *idle* and *sleep* modes (Table I) to take advantage of those mechanisms optimally. Beacon-enabled networks use a slotted carrier sense multiple access mechanism with collision avoidance (CSMA/CA), and the slot boundaries of each device are aligned with the slot boundaries of the PAN coordinator. To save energy, nodes do not have to listen to the channel continuously, so it is possible to switch to low-power when the wake-up delay fits within the timing constraints of the medium access control. While contending for the channel, nodes delay their carrier sensing by a random backoff delay during which they go to low power modes. Only after that random delay, the contending node wakes-up to listen to the channel during maximally two backoff slots. As a result, the power consumption during channel listening is minimized. To receive data, pending data reception is announced through the beacon, and the data is sent by the central coordinator only after receiving the data request message that informs the coordinator that the device will be listening for the data (Fig. 1). Finally, in the beacon-enabled channel access mode, periods can be announced in the beacon during which the network will be asleep to save even more energy. An optional time division multiple access (TDMA) scheme can be added, in addition to the mandatory CSMA/CA. It is clear that the beacon-enabled mechanism was developed mainly to allow for energy savings.

The 802.15.4 standard also specifies an access mechanism that does not rely on the availability of beacons. This simpler unslotted CSMA/CA also incorporates the delay line to minimize the channel listening. Energy savings through network shutdown and pending data announcement are however more difficult to implement without beacons. Hence in this paper, we will analytically verify the impact of the beacon-enabled slotted CSMA/CA mechanism on both throughput and energy consumption. We will mainly focus on the uplink scenario which is most relevant for sensor networks (Fig. 1).

The performance of the IEEE 802.15.4 protocol has been evaluated by simulation for small and low load networks in [5] and for dense networks in [6]. In contrast, this paper provides an analytical Markov model that predicts the performance and detailed behavior of the 802.15.4 slotted CSMA/CA mechanism. This is then verified for accuracy by detailed comparison to simulation. The model incorporates details of the exponential delay lines and double Clear Channel Assessment (CCA). The form of the analysis is similar to that of Bianchi for IEEE 802.11 DCF [10], [11] only in the use of a per user Markov model to capture the state of each user at each moment in time. The assumptions to enable this important simplification and the coupling of the per user models are however different, as a result of the very different design of the 802.11 carrier sensing mechanism where nodes monitor the channel continuously and are hence continuously aware of the channel state. This small difference results however in a key difference in the main approximation assumptions: Each device's carrier sensing probability, rather than its packet sending probability, is assumed independent. Also, unlike in 802.11, the slot duration is fixed since the channel is not constantly monitored by the stations and only a fixed slot duration model can keep the system synchronized. Finally, the fixed duration two-slot clear channel assessment leads to memory in the coupling of the per user individual Markov chains. The analytical model for IEEE 802.15.4 developed in [7], [8] fails to match the simulation results, since they used the same Markov formulation and assumptions as Bianchi in [10] for 802.11.

This problem has been reported in [9] as well, and a better model has been proposed there. However, after detailed analysis this model does not mimic the 802.15.4 behavior sufficiently either, since they do not correctly couple the per user Markov chains through the two-slot channel sensing. This will be explained and shown in more detail in this paper in Section III.

For the analysis, we further consider two traffic scenarios which are particularly relevant in the context of sensor networks. First, we assume a large sensor network that has been deployed to monitor events. Upon the detection of an event, we can assume all sensors will have data to send to a central data gathering device. This traffic condition is well modeled by assuming a large number of nodes where each node has a packet to send, which is the saturation condition. Secondly, sensor networks are typically deployed for periodic monitoring purposes. Measurements should be transmitted at regular time intervals, but the measurement update period varies depending on the application instance. This more periodic traffic case is also modeled.

In the remainder, section II briefly describes the slotted CSMA/CA mechanism in IEEE 802.15.4, which is analyzed in Section III for uplink, acknowledged uplink and downlink traffic under saturation and unsaturated periodic traffic. Section IV validates the accuracy of the model by comparing the analytical predictions and simulation results. Section V gives energy and throughput results for both saturated and unsaturated cases and gives some design guidelines that can be derived easily with the proposed model. Section VI concludes the paper.

## II. IEEE 802.15.4 slotted CSMA/CA Mechanism

We briefly explain the 802.15.4. medium access control mechanism. When considering beacon-enabled mode, the beacons are used to synchronize the attached devices, to identify the PAN, and to describe the structure of superframes. The superframes are bounded by network beacons and divided into 16 equally sized slots. The beacon frame is sent in the first slot of each superframe.

The superframe can have an active and an inactive portion. During the inactive portion the coordinator does not interact with its PAN and may enter a low-power mode. The active portion consists of a contention access period (CAP) and a contention free period (CFP). A device that wishes to communicate during the CAP competes with other devices using a slotted CSMA/CA mechanism. On the other hand, the CFP contains guaranteed time slots (GTSs). The GTSs appear at the end of the active portion starting at a slot boundary immediately following the CAP.

In the slotted CSMA/CA channel access mechanism, the backoff slot boundaries of every device in the PAN are aligned with the superframe slot boundaries of the PAN coordinator. Each time a device wishes to transmit data frames during the CAP, it must locate the boundary of the next slot period. Moreover, before accessing the channel, a random number of backoff slots should be waited. During this time, the device is idle but not scanning the channel to save energy. After the random delay, a two slot clear channel assessment is carried out.

The exact mechanism that has to be followed before accessing the channel is depicted in Figure 2 and explained below. Each device in the network has three variables: NB, CW and BE. NB is the number of times the CSMA/CA algorithm was required to delay while attempting the current transmission. It is initialized to 0 before every new transmission. CW is the contention window length, which defines the number of slot periods that need to be clear of activity before the transmission can start. It is initialized to 2 before each transmission attempt and reset to 2 each time the channel is assessed to be busy. BE is the backoff exponent, which is related to how many slot periods a device must wait before attempting to assess the channel. The initial value differs when `Battery Extension` is enabled, since in that case the number of slots to wait before sensing the channel is decreased to minimize energy spent while waiting. Although the receiver of the device is enabled during the channel assessment portion of this algorithm, the device must discard any frames received during this time.

The slotted CSMA/CA mechanism works as follows. NB, CW and BE are initialized and the boundary of the next slot period is located (step1). The MAC layer delays for a random number of complete slot periods in the range 0 to $2^{BE} - 1$ (step 2) and then requests PHY to perform a CCA (clear channel assessment) (step 3). The MAC sublayer then proceeds provided that the remaining CSMA/CA algorithm steps—frame transmission, and any acknowledgment—can be completed before the end of the CAP. If the MAC sublayer cannot proceed, it must wait until the start of the CAP in the next superframe and then repeat the evaluation.

If the channel is assessed to be busy (step 4), the MAC sublayer increments both NB and BE by one, ensuring that BE is not more than *aMaxBE*, and CW is reset to 2. If the value of NB is less than or equal to *macMaxCSMABackoffs*, the CSMA/CA must return to step 2, else the CSMA/CA must terminate with a Channel-Access-Failure status. The parameters used are listed in Table I.

If the channel is assessed to be idle (step 5), the MAC sublayer must ensure that the contention window is expired before starting transmission. For this, the MAC sublayer first decrements CW by one. If CW is not equal to 0, it must go to step 3, else start transmission on the boundary of the next slot period.

## III. Formulation

The core contribution of this paper is the analytical modeling of the slotted CSMA/CA mechanism of the IEEE 802.15.4 standard. We start with the formulation for unacknowledged uplink data transmission. Next, we will extend the formulation for acknowledged uplink and unsaturated traffic conditions.

### A. Uplink saturation

We assume a network of a fixed number $N$ of devices, and each device always has a packet available for transmission. The parameters used in this section are summarized in Table II.

The analysis is in two steps, and the goal is to find a set of equations that uniquely define the network operating point. We first study the behavior of a single device using a Markov model (Fig.4). From this model, we obtain the stationary probability $\phi$ that the device attempts it's carrier channel assessment (CCA) for the first time within a slot. ($\phi$ is the counterpart of the probability $\tau$ that the device transmits a packet in a virtual slot in the analysis of 802.11 in [10].) For 802.11 in saturated traffic conditions, a device is always listening to the channel when not transmitting: $\phi = 1 - \tau$. In 802.15.4, this $\phi$ is a function of the exponential delay line as explained in Section II. Secondly, we couple the per user Markov chains, to obtain an additional set of equations to be able to solve the system. An important assumption in this coupling is that the probability to start sensing the channel is independent across the nodes. This coupling is very different than for 802.11. Indeed, in 802.11 networks, users constantly monitor the channel and are hence aware of the medium state. For 802.15.4, this is not the case. It will be shown later that two additional equations are required to fully determine the system.

We first develop the Markov model to determine $\phi$, see Fig. 4. Let $c(t)$ be the stochastic process representing the delay line and transmission duration counters of the device. The integer time $t$ corresponds to the beginning of the slot times. In contrast to the model in [10], $t$ corresponds directly to system time. After the delay counter is decremented to zero, $c = 0$, the values $c = -1$ and $c = -2$ correspond to the first CCA ($CCA^1$) and second CCA ($CCA^2$), respectively.

Let $\alpha$ be the probability of assessing channel busy during $CCA^1$, and let $\beta$ be the probability of assessing it busy during $CCA^2$, given that it was idle in $CCA^1$. Next, when entering the transmission state, $L$ slots should be counted, where $L$ denotes the packet transmission duration measured in slots[1].

Let $s(t)$ be the stochastic process representing the delay line stages ($s(t) \in \{0, \cdots, NB\}$), or the transmission stage ($s(t) = -1$) at time $t$. The stages ($s(t) = -2$) in Fig. 4 will be introduced in Section III-C to model

[1]We assume that this duration is an integer number of slots in the remainder.

unsaturated periodic traffic and are not relevant at this point. Here, they can be assumed to have zero delay or impact. We assume that the probability to start sensing is constant and independent of all other devices and of the number of retransmissions suffered. With this important assumption, $\{s(t), c(t)\}$ is the two-dimensional Markov chain of Fig. 4 with the following transition probabilities:

$$\begin{aligned} P\{i,k|i,k+1\} &= 1,\ k \geq 0 && (1) \\ P\{0,k|i,0\} &= (1-\alpha)(1-\beta)/W_0,\ i < NB && (2) \\ P\{i,k|i-1,0\} &= (\alpha + (1-\alpha)\beta)/W_i, \\ & \qquad i \leq NB, k \leq W_i - 1 && (3) \\ P\{0,k|NB,0\} &= (1-\alpha)(1-\beta)/W_0 + P_f/W_0. && (4) \end{aligned}$$

The delay window $W_i$ is initially $W_0 = 2^{aMinBE}$ and doubled any stage until $W_i = W_{max} = 2^{aMaxBE}$, $(aMaxBE - aMinBE) \leq i \leq NB$.

Equation 1 is the condition to decrement the delay line counter per slot. Equation 2 states that it is only possible to enter the first delay line from a stage that is not the last one ($i < NB$) after sensing the channel idle two consecutive times and hence transmitting a packet. Equation 3 gives the probability that there is a failure on both channel assessments or sensing slots (CCA) and the station selects a state in the next delay level. Equation 4 gives the probability of starting a new transmission attempt when leaving the last delay line, following a successful or failed packet transmission attempt. Note that the number of transmission attempts is limited and either ends with a packet transmission or failure $P_f$.

Denote the Markov chain's steady-state probabilities by $b_{i,k} = P\{(s(t), c(t)) = (i,k)\}$, for $i \in \{-1, NB\}$ and $k \in \{-2, \max(L-1, W_i - 1)\}$. Using Equation 3 we get

$$b_{i-1,0}(\alpha + (1-\alpha)\beta) = b_{i,0}\ , 0 < i \leq NB, \qquad (5)$$

which leads to

$$b_{i,0} = [(\alpha + (1-\alpha)\beta)]^i b_{0,0}\ , 0 < i \leq NB. \qquad (6)$$

From Equations 1- 4 we obtain

$$b_{i,k} = \frac{W_i - k}{W_i}\{(1-\alpha)(1-\beta)\sum_{j=0}^{NB} b_{j,0} + P_f\}, \qquad (7)$$

for $i = 0$ and

$$b_{i,k} = \frac{W_i - k}{W_i} b_{i,0}, \qquad (8)$$

for $i > 0$.

Since the probabilities must sum to 1,

$$\begin{aligned} 1 &= \sum_{i=0}^{NB}\sum_{k=1}^{W_i-1} b_{i,k} + \sum_{i=0}^{NB} b_{i,-1} + \sum_{i=0}^{NB} b_{i,-2} + \sum_{i=0}^{L-1} b_{-1,i} \\ &= \sum_{i=0}^{NB} b_{i,0}\Big[\frac{W_i-1}{2} + 1 + (1-\alpha) + (1-\alpha)(1-\beta)L\Big]. \end{aligned} \tag{9}$$

Substituting the expression for $W_i$ this leads to

$$\begin{aligned} 1 &= \frac{b_{0,0}}{2}\{[1+2(1-\alpha)+2(1-\alpha)(1-\beta)L] \times [\frac{1-(\alpha+\beta-\alpha\beta)^{NB+1}}{1-(\alpha+\beta-\alpha\beta)}] \\ &+ 2^{aMaxBE-aMinBE} W_0[\frac{(\alpha+\beta-\alpha\beta)^{aMaxBE-aMinBE+1}-(\alpha+\beta-\alpha\beta)^{NB}}{1-(\alpha+\beta-\alpha\beta)}] \\ &+ W_0[\frac{1-[2(\alpha+\beta-\alpha\beta)]^{aMaxBE-aMinBE+1}}{1-2(\alpha+\beta-\alpha\beta)}]\}. \end{aligned} \tag{10}$$

The transmission failure probability $P_f$ is

$$P_f = b_{NB,0}(\alpha - \beta\alpha + \beta), \tag{11}$$

and the probability that a node starts to transmit is (this corresponds to the transmission probability $\tau$ in Bianchi's model)

$$\tau = P_s = \phi(1-\alpha)(1-\beta), \tag{12}$$

in which

$$\phi = \phi_1 = \sum_{i=0}^{N} B b_{i,0}. \tag{13}$$

We have now derived one expression for $\phi$ from the per user Markov models. By determining the interactions between users on the medium, we will now derive expressions for the remaining unknowns $\alpha$ and $\beta$.

Denote by $M(s) = -1$ the event that there is at least one transmission in the medium by another node and assume that, without loss of generality, that the node sensing is $i_N$, which is denoted as $S^{i_N}(c) = -1$ if $S^i(s) = -1$ is the event that node $i$ is transmitting. Then, the probability $\alpha$ which is the probability that a node sensing the channel finds it to be occupied is $P(M(s) = -1|S^{i_N}(c) = -1)$. Hence, $\alpha$ is computed as follows:

$$\begin{aligned} \alpha &= P(M(s) = -1|S^{i_N}(c) = -1) \\ &= \textstyle\sum_{n=0}^{N-2} \binom{N-1}{n+1} P(\bigcup_{k=1}^{n+1} S^{i_k}(s) = -1|S^{i_N}(c) = -1) \\ &= \textstyle\sum_{n=0}^{N-2} \binom{N-1}{n+1} P(S^{i_1}(s) = -1) P(\bigcup_{k=2}^{n+1} S^{i_k}(s) = -1|S^{i_1}(s) = -1, S^{i_N}(c) = -1). \end{aligned} \tag{14}$$

Let $E_c$ denote the event that node $i_1$ is in state $(-1, c)$. The probability that node $i_1$ is transmitting is

$$P(S^{i_1}(s) = -1) = \sum_{c=0}^{L-1} P(E_c) = LP(E_0) = LP_s \tag{15}$$

$$= L\phi(1-\alpha)(1-\beta), \tag{16}$$

which requires the node to sense (with probability $\phi$) two slots before transmission and the following two slots to be empty (with probability $(1-\alpha)(1-\beta)$).

To express the conditional probability in terms of $\phi$, the transmission pattern needs to be understood: If there are two or more transmissions in a particular slot, the transmissions must start at the same slot, because devices that transmit later would detect earlier transmissions and would not start transmitting. Starting transmission at the same time slot moreover requires being in the sense state at the same time. Thus the conditional probability is hence equivalent to

$$P(\bigcup_{k=2}^{n+1} S^{i_k}(s) = -1 \mid S^{i_1}(s) = -1, S^{i_N}(c) = -1) = P(\bigcup_{k=2}^{n+1} S^{i_k}(c) = -1 \mid S^{i_1}(c) = -1, S^{i_N}(c) = -1). \quad (17)$$

Since a main assumption in this paper is that the probability $\phi$ to sense in a given slot is independent across nodes, we can easily see that this is

$$P(\bigcup_{k=2}^{n+1} S^{i_k}(c) = -1 \mid S^{i_1}(c) = -1, S^{i_N}(c) = -1) = \phi^n (1-\phi)^{N-2-n}, \quad (18)$$

which requires nodes $i_2, ..., i_{n+1}$ to sense and the remaining $N-2-n$ nodes not to sense in the sensing slot of $i_1$. Resultingly, we can see that the probability $\alpha$ that a node $i_N$ sensing the channel the first time finds it busy is equal to the probability that one of the remaining $N-1$ nodes sense the channel successfully and send, independently of the node $i_N$. We note this as

$$\alpha = P(M(s) = -1 | S^{i_N}(c) = -1) = P(M(s) = -1), \quad (19)$$

where we defined $M$ to be the medium without the current node $N$.

Thus

$$\alpha = L[1 - (1-\phi)^{N-1}](1-\alpha)(1-\beta). \quad (20)$$

From this, we can derive a second expression for $\phi$:

$$\phi_2 = 1 - [(1 - \frac{\alpha}{L(1-\alpha)(1-\beta)})]^{\frac{1}{N-1}}. \quad (21)$$

A third relation is needed to be able to solve the system and determine the operating point. For this purpose, we derive an expression for $\beta$. A packet transmission starts when the channel has been sensed idle two consecutive times. Although the probability to sense (busy) the first time can be considered to be independent across users, the probability to sense busy the second attempt can not. Indeed, shared knowledge about the first slot being empty for all users strongly impacts the probabilities in the second sensing slot. The same reasoning leads to the packet send probabilities to be correlated across users, i.e. after the shared knowledge of two empty slots. Although the authors in [9] correctly assess the correlated sending probability, they do not correctly take into account the effect for the second sensing probability. As a result, a detailed analysis of the sensing probabilities in their model does not match simulation results.

We define $\beta$ as the probability that there is a transmission in the medium when the considered device does its second sense in slot $CCA^2$, given that the medium was idle during its first sense in slot $CCA^1$,

$$\beta = Pr(M_{CCA^2}(s) = -1 | M_{CCA^1}(s) \geq 0 \cap S^{i_N}_{CCA^1}(c) = -1), \tag{22}$$

where $M(s) \geq 0$ denotes the probability that no station in the medium is transmitting (other than the station $i_N$). The superscript denotes the local time of the node doing its second sense as shown in Fig. 3(a). We note that $M_{CCA^1}(s) \geq 0 \cap S^{i_N}_{CCA^1}(c) = -1$ is equivalent to $S^{i_N}_{CCA^2}(c) = -1$, i.e., a node can only be sensing during the second slot $CCA^2$ when it was sensing during the first slot and the medium was empty then.

In the unacknowledged uplink case, the device will sense busy only if some other node in the medium was sensing its second time during our device's first sense and started a new transmission in slot $CCA^2$. This can only happen if this node started sensing in slot $1$ ($M_1(c) = -1$) and the channel was then idle ($M_1(s) \geq 0$).

That is

$$\begin{aligned} \beta \quad &= P(M_1(s) \geq 0 \mid M_1(c) = -1, M_{CCA^1}(s) \geq 0, S^{i_N}_{CCA^1}(c) = -1) \\ &\times P(M_1(c) = -1 \mid M_{CCA^1}(s) \geq 0, S^{i_N}_{CCA^1}(c) = -1) \\ &= P(M_1(s) \geq 0 \mid M_1(c) = -1, M_{CCA^1}(s) \geq 0, S^{i_N}_{CCA^1}(c) = -1) \times P(M_1(c) = -1) \end{aligned} \tag{23}$$

Here $P(M_1(s) \geq 0 \mid M_1(c) = -1, M_{CCA^1}(s) \geq 0, S^{i_N}_{CCA^1}(c) = -1)$ is the probability that a given idle slot is preceded by another idle slot when a node is sensing. Also, the last simplification to $P(M_1(c) = -1)$ is possible since the probability to start sensing is independent of the medium status during that slot or the slot following. We can see that

$$\begin{aligned} &P(M_1(s) \geq 0 \mid M_1(c) = -1, M_{CCA^1}(s) \geq 0, S^{i_N}_{CCA^1}(c) = -1) \\ &= 1 - \frac{P(M_1(s)=-1 \cap M_{CCA^1}(s) \geq 0 | M_1(c)=-1, S^{i_N}_{CCA^1}(c)=-1)}{Pr(M_{CCA^1}(s) \geq 0 | M_1(c)=-1, S^{i_N}_{CCA^1}(c)=-1)}. \end{aligned} \tag{24}$$

This means that to compute the nominator of this probability we have to list all cases that result in an idle slot $CCA^1$ when a node is sensing $(M_{CCA^1}(s) \geq 0 | M_1(c) = -1, S^{i_N}_{CCA^1}(c) = -1)$, and see which of those have a busy slot $1$ before ($M_1(s) = -1$). Indeed, all cases that result in an idle slot $CCA^1$ are, obviously, the sum of the ones with a busy slot $1$ and an idle slot $1$:

$$\begin{aligned} &P(M_{CCA^1}(s) \geq 0 | M_1(c) = -1, S^{i_N}_{CCA^1}(c) = -1) \\ &= P(M_1(s) = -1 \cap M_{CCA^1}(s) \geq 0 | M_1(c) = -1, S^{i_N}_{CCA^1}(c) = -1) \\ &+ P(M_1(s) \geq 0 \cap M_{CCA^1}(s) \geq 0 | M_1(c) = -1, S^{i_N}_{CCA^1}(c) = -1), \end{aligned} \tag{25}$$

We list now the different cases that result in an idle slot $CCA^1$. Those cases that are preceded by an busy slot $1$ are denoted as **Case 1** while those preceded by an idle slot $1$ are **Case 2**:

- **Case 1:** A busy slot $1$ right before before the idle slot $CCA^1$ is counted in $P(M_1(s) = -1 \cap M_{CCA^1}(s) \geq 0 | M_1(c) = -1, S^{i_N}_{CCA^1}(c) = -1)$. This is the case when there is a start of a transmission exactly $L$ slots before the slot $CCA^1$, which is denoted by a transmission start in slot $L$ in Fig. 3. In that case both CCA$^1$ and CCA$^2$ are guaranteed to be idle, since no node could have sensed successfully during slot $1$. The probability to have a transmission starting at slot $L$ (in Fig. 3), given the node $i_N$ is sensing its second time, is written as

$$\begin{aligned}&P(M_L(s,c)=(-1,1)\cap M_{CCA^1}(s)\geq 0|M_1(c)=-1,S^{i_N}_{CCA^1}(c)=-1)\\&=P(M_L(s,c)=(-1,1)|M_1(c)=-1,S^{i_N}_{CCA^1}(c)=-1)\end{aligned} \tag{26}$$

This step is based on the fact that when a transmission starts at slot $L$, slot $CCA^1$ is guaranteed to be idle. From here on, the derivation is very similar to the computation of $\alpha = P(M(s)=-1|S^{i_N}(c)=-1)$ which also computes the probability that the medium is busy conditioned on sensing events. We have to replace $M(s)=-1$ by $M(s,c)=(-1,1)$ and $S(s)=-1$ by $S(s,c)=(-1,1)$ in Eq.(14). This will result in considering only $E_0$ in Eq.(15), hence $P(S(s,c)=(-1,1))=\phi(1-\alpha)(1-\beta)=P_s$. Next, the conditioning here depends on node $i_N$ starting the sensing during slot $CCA^1$ and at least one other of the $N-1$ nodes starting sensing during slot $1$. This limits the number of nodes that are free to start a transmission at slot $L$. Although it is possible for $i_N$ to start a transmission at slot $L$ and start sensing at slot $CCA^1$, clearly the nodes starting their sense at slot $1$ cannot start a transmission at slot $L$. Assume that $x$ nodes are starting a transmission at slot $1$. Resultingly, it is possible to show that

$$\begin{aligned}P(M_L(s,c)=(-1,1)|S^{i_N}_{CCA^1}(c)=-1)\quad&=(1-(1-\phi)^{N-x})(1-\alpha)(1-\beta)\\&=P^{N-x}_{\text{send}}.\end{aligned} \tag{27}$$

We call this $P^{N-x}_{\text{send}}$ since it denotes the probability that a node (different from the $x$ nodes sensing) are starting a transmission at a particular slot. Indeed, this requires that one of the $N-x$ nodes starts sensing, provided those sensing events are successful.

- **Case 2:** An idle slot $1$ right before the slot $CCA^1$ that was sensed by the node $i_N$ is counted in $P(M_1(s)\geq 0\cap M_{CCA^1}(s)\geq 0|S^{i_N}_{CCA^1}(c)=-1)$. If a node starts sensing during that idle slot $1$, it will sense the channel idle both during that slot $1$ and the next slot $CCA^1$, since we already know that that slot was sensed idle by node $i_N$. Hence, if a node starts sensing during that idle slot, we know for sure this node will start transmitting during slot $CCA^2$ and cause a $busy$ event.

  Such an idle slot $1$ can be the result of a transmission that ended the slot before (i.e., slot $2$ in Fig. 3), leading to at least two idle slots after the transmission (i.e., slots $1$ and $CCA^1$ in this case). In addition, slot $1$ can be idle given that $CCA^1$ is also idle when a transmission ended at any slot $i, 3\leq i<\infty$, and no one started sensing in the slots $2,\ldots,i-1$. For instance, when a transmission ended at slot $3$, and a node would be sensing in the following idle slot $2$, that node would also sense slot $i$ to be idle, resulting in a transmission in slot $CCA^1$ by that node. Since this contradicts our information that $CCA^1$ is idle, we hence have to add the constraint that no node can be sensing in the idle slots following the end of the last transmission. We can sum up all the possible cases that result in an idle slot $1$ before an idle $CCA^1$:

$$P(M_1(s)\geq 0\cap M_{CCA^1}(s)\geq 0|S^{i_N}_{CCA^1}(c)=-1)=\sum_{i=2}^{\infty}P^{N}_{\text{send}}((1-\phi)^N)^{(i-2)}. \tag{28}$$

  Here $P^N_{\text{send}}$ is the probability that a node is starting a transmission at a particular slot, as derived before. We note that all nodes, including those sensing, should be considered here.

Finally, to complete the computation of $\beta$ in Eq.(23) the probability that some node starts sensing during slot 1 is

$$P(M_1(c) = -1) = (1 - (1 - \phi)^{N-1}). \tag{29}$$

Then $\beta$ is given by

$$\beta = \left[1 - \frac{P_{\text{send}}^{N-x}}{P_{\text{send}}^{N-x} + \frac{P_{\text{send}}^{N-x}}{1-(1-\phi)^N})}\right](1 - (1 - \phi)^{N-1}) \tag{30}$$

which for large $N$ can be simplified to;

$$\beta = [1 - \frac{P_{\text{send}}^{N}}{P_{\text{send}}^{N}(1 + \frac{1}{1-(1-\phi)^N})}](1 - (1 - \phi)^{N}) \tag{31}$$

from which one obtains

$$\phi_3 = 1 - (1 - \frac{\beta}{1-\beta})^{1/N} \tag{32}$$

The network operating point as determined by $\phi$, $\alpha$ and $\beta$ is determined by solving the three non-linear Equations 13, 20, 32. These three values are sufficient to determine the network throughput and energy consumption achieved during operation, as we will show in Section V. First, we however extend the model to cover also acknowledged uplink traffic. Also, it is shown how unsaturated traffic scenarios that are relevant in the context of sensor networks can be modeled.

### B. *Extension to acknowledged uplink traffic*

In the case acknowledgments are expected by the node transmitting its data, each packet of size $L$ is followed by an idle period equal to $t_{ACK}$ slot times, and an acknowledgment of size $L_{ACK}$. According to the 802.15.4 standard, the $t_{ACK}$ should be less than or equal to $aTurnaroundTime + aUnitBackoffPeriod = 12 + 20\ symbols$. As a result, this $t_{ACK}$ can be larger than a backoff slot, and a station sensing the channel in between the data and it's ACK can sense the channel to be idle. In fact, this is the main motivation for the double CCA. To take into account the effect of this idle time, we have to take the integer number of slots that fit into this time and can hence be sensed idle: $\lfloor t_{ACK} \rfloor = 1slot$. The acknowledgment for 802.15.4 is $11Bytes$ long, which is slightly more than a slot. We model the time as $\lceil L_{ACK} \rceil = 2slots$. Given the timing of the 802.15.4 acknowledgment scheme, we show how the three expressions for $\phi$ have to be changed to consider this acknowledged case.

First, the per user Markov model expressions have to be updated to capture the extra time spent waiting for the ACK. We assume that, in case no ACK is received, the time a node waits for its ACK equals $t_{ACK} + L_{ACK}$. In that case, $L$ in Equation 10 should be replaced by $L' = L + t_{ACK} + L_{ACK}$ for $\phi_1$.

Next, a user doing its first clear channel assessment can also find the channel busy because of the ACK transmitted. Define $L^* = L + L_{ACK} \times (1 - P_{\text{netcol}})$, where $P_{\text{netcol}}$ is the probability that a collision is seen on the channel on the condition that a transmission was going on, and as a result no acknowledgment is sent. It can be defined as:

$$P_{\text{netcol}} = 1 - \frac{P_{\text{success}}}{P_{\text{send}}} = 1 - \frac{N\phi(1-\phi)^{N-1}}{1-(1-\phi)^N} \tag{33}$$

Since the probability $P_{\text{success}}$ to have a successful transmission is the probability that a node senses two times successfully, and the others not:

$$P_{\text{success}} = N\phi(1-\phi)^{N-1}(1-\alpha)(1-\beta) \tag{34}$$

and the probability $P_{\text{success}}$ that a node in the network sends a packet is the probability that at least one node is sensing successfully

$$P_{\text{send}} = P_{\text{send}}^{N} = 1-(1-\phi)^{N}(1-\alpha)(1-\beta). \tag{35}$$

To determine the new expression for $\alpha$, we have to replace $L$ in Eq. 20 with $L^{*}$ for $\phi_2$.

Finally, we have to take into account the fact that a user can find the channel clear during the first clear channel assessment $CCA^1$ when this slot coincides with the time between data and acknowledgment $t_{ACK}$. When the data transmission was successful, an acknowledgment will follow and the user doing its second clear channel assessment $CCA^2$ will find the channel busy. This is different compared to the unacknowledged case, where a second busy sense can only occur with a new transmission start, which led to the condition that some node had to be sensing before in slot 1 ($M_1(c)=-1$). We hence update Eq. 23

$$\begin{aligned}\beta_{ACK} &= P(M_1(s)\geq 0 \mid M_1(c)=-1, M_{CCA^1}(s)\geq 0, S^{i_N}_{CCA^1}(c)=-1)\times P(M_1(c)=-1)\\ &+P_{\text{betaACK}}\end{aligned} \tag{36}$$

Here $P_{\text{betaACK}}$ is the probability to have a collision with the acknowledgment, and is given by

$$P_{\text{betaACK}} = \frac{P_{\text{send}}\times(1-P_{\text{netcol}})}{P(M_{CCA^1}(s)\geq 0|S^{i_N}_{CCA^1}(c)=-1)} \tag{37}$$

where the denominator expresses the fact that we know that the first sense succeeded, i.e. there was an idle slot and a node was sensing. Note that to compute $P_{\text{betaACK}}$ we do not need to assume that a node is sensing during slot 1, i.e., the busy slot $CCA^2$ is in this case not caused by a new transmission (which required sensing in slot 1).

The probability to sense a new transmission during the slot $CCA^2$ is computed similarly to the unacknowledged case. Therefore, we list all cases that result in an idle slot $CCA^1$ when node $N$ is sensing $P(M_{CCA^1}(s)\geq 0|M_1(c)=-1, S^{i_N}_{CCA^1}(c)=-1)$. From those cases, the ones with idle slot 1 will result in a new transmission during slot $CCA^2$ (since we condition on the fact that a node starts sensing during slot 1: $M_1(c)=-1$). We can see in Figure 3(b) that following cases result in an idle slot $CCA^1$ ($M_{CCA^1}(s)\geq 0$):

- **Case 1:** A busy slot before the idle slot is counted in $P(M_1(s)=-1\cap M_{CCA^1}(s)\geq 0|M_1(c)=-1, S^{i_N}_{CCA^1}(c)=-1)$. Two sub-cases can be considered:
  - **Case 1.1:** In the first sub-case, the idle slot follows a data packet transmission, which can be a successful or a collided data transmission. This is similar to **Case 1** in the unacknowledged case, which occured with probability $P_{\text{send}}^{N-x}$.
  - **Case 1.2:** In the second sub-case, the busy slot just before the idle slot is in between a data packet and the acknowledgment following a successful data transmission. The probability to have a successful data transmission is $P_{\text{success}} = N\phi(1-\phi)^{N-1} = P_{\text{send}}\times(1-P_{\text{netcol}})$. This approximates the probability

to have Case 1.2, following similar derivations as we did before. Indeed, in order to derive the correct probability for this subcase, we shoud substitute $N$ by $N-x$ where $x$ is the number of nodes sensing during slot 1. However, since we will neglect this $x$ to derive a compact expression for $\beta$ later, we do not compute it exactly here.

- **Case 2:** An idle slot 1 right before slot $CCA^1$ is counted in $P(M_1(s) \geq 0 \cap M_{CCA^1}(s) \geq 0 | M_1(c) = -1, S^{i_N}_{CCA^1}(c) = -1)$. This can happen when an acknowledged successful transmission or an unacknowledged unsuccessful transmission ended in slot 2. The sum of both events is $P_{\text{send}}$. Alternatively, the transmission ended before slot 2, but no node started sensing after the end of the transmission, until slot 1. The resulting probability for this case is hence the same as in the unacknowledged Case 2: $P(M_1(s) \geq 0 \cap M_{CCA^1}(s) \geq 0) = \sum_{i=1}^{\infty} P_{\text{send}}((1-\phi)^N)^{(i-1)}$.

The expression for $\beta$ in the acknowledged case is hence:

$$\beta_{ACK} = \left[1 - \frac{P_{\text{send}}[(1-P_{\text{netcol}})+1]}{P_{\text{send}}(2-P_{\text{netcol}}+\frac{1}{1-(1-\phi)^N})}\right](1-(1-\phi)^{N-1}) + \frac{P_{\text{send}}(1-P_{\text{netcol}})}{P_{\text{send}}(2-P_{\text{netcol}}+\frac{1}{1-(1-\phi)^N})} \quad (38)$$

which, for $N$ large, can be simplified to;

$$\beta_{ACK} = \frac{2-P_{\text{netcol}}}{(2-P_{\text{netcol}}+\frac{1}{1-(1-\phi)^N})}. \quad (39)$$

And hence the updated expression for $\phi_3$ in the acknowledged case is:

$$\phi_{3,ACK} = 1 - (1 - \frac{\beta_{ACK}}{(1-\beta_{ACK})(2-P_{\text{netcol}})})^{1/N} \quad (40)$$

### *C. Extension to unsaturated traffic conditions*

So far we have modeled saturated traffic conditions, in which case all nodes constantly have packets to send. This case can be used to model the performance in sensor networks in which sensors want to send data at the same moment, e.g. when an event is detected. In reality, however, this is not always the case since often sensor networks are designed to continuously monitor a variable and send the measurement to a central sink. The variable can be monitored periodically where the period depends on the application. In this subsection, we propose a model for those unsaturated traffic conditions. The model assumes periodic traffic scenarios where the average traffic generation rate is identical for all nodes in the network and constant over time.

The idea behind this extension is to introduce additional states following each periodic transmission or transmission attempt, as depicted by the states ($s(t) = -2$) in Fig. 4. The number of those states is fixed and tuned based the packet generation period. Each of the states represents one slot delay, where a slot is $0.32\ ms$. As shown in Fig. 4, we consider three different types of traffic delay slots, which can each be tuned depending on the application requirements.

Delaying the next transmission attempt after a successful transmission is obtained by tuning $X_3$. This models an application where a variable should be reported reliably to a central sink approximately every $T$ slots, where $T = X_3 + W_3$ with $W_3$ the expected waiting time until the end of the successful transmission. However, when the transmissions are not acknowledged, it is not possible to know if the transmission was successful, and this application scenario is not possible. Moreover, it could be the sensor network application policy to not report a measurement more than once, e.g. to save energy, when reliability is not the main design goal. If the transmission failed, a more recent measurement will be transmitted later, at a moment where there is (hopefully) less network activity hence less collision probability. In those cases, $X_2$ is the additional delay that should be added before the next periodic packet transmission. Finally, in some energy sensitive applications, it could be the policy to only contend once to send one unit of information and resume transmission when that single contention cycle ends. In this case, contention energy consumption is decreased as much as possible by delaying transmissions whenever the channel is detected to be very busy. In that case, the delay $X_1$ is added after each transmission attempt. We will focus on this last case since it is particularly relevant for energy critical sensor network applications.

The formulations developed for the saturated case can then be used for the unsaturated case except the following term that should be added to Eq. 10:

$$b_{0,0}\{[X_1 + (1-\alpha)(1-\beta)\{X_2 + (1-P_c)X_3\}] \times [\frac{1-(\alpha+\beta-\alpha\beta)^{m+1}}{1-(\alpha+\beta-\alpha\beta)}] \tag{41}$$

where $P_c$ is the probability that a node's transmission collides with another transmission: $P_c = 1-(1-\phi)^{N-1}$. The parameters $X_1$, $X_2$ and $X_3$ need to be tuned, as discussed before. It should be clear that putting those parameters to $0$ models the above discussed saturated case. Also, in this model, it is possible to add have a non-zero number of slots for each $X$. Typically however, we assume that only one of them is non-zero, as function of the application requirements.

## IV. Model Validation

### A. Simulation Model

To validate the proposed analytical model, a Monte-Carlo simulation of the 802.15.4 contention procedure is considered. We developed therefore a vector-based simulator in Matlab. The pseudo-code of the simulation loop is depicted in Algorithm 1. It can easily be verified that it matches the 802.15.4 CSMA/CA algorithm as depicted in Fig. 2. We only depict the code for unacknowledged and saturated uplink traffic. Extension to the other traffic scenarios is straightforward. Since the 802.15.4 CSMA/CA scheme can be captured with a fixed duration backoff slot, it is possible to consider each simulation step a new slot. The system state during each slot is tracked using state vectors of dimension $N$, which is the number of nodes. Three vectors represent the values of *NB*, *CW* and *BE*. A fourth vector *delay* represents both the state of the node ($-1$ if idle, $0$ if transmitting and contention, $> 0$ in the delay link). When a node is in transmit state, the remaining transmission time in slots in maintained in the corresponding element of the fifth vector (*busyFor*). The number of transmissions (*nbTransmission*), collisions (*nbCollision*), access failures (*nbFailure*) and CCA's (*nbCCA*) are tracked for a simulation run of $T = 10^8$ slots. Based on that, the collision probability, failure probability and average number of CCA per channel access can be computed and compared to the prediction of the analytical model.

```
%State vectors
NB(1:.N) = 0;
CW(1:.N) = 2;
BE(1:.N) = aMinBE;
delay(1:.N) = random number in [1 ... 2^BE(i)] ;
busyFor(1:.N) = 0;

%Tracking variables
nbTransmission;
nbCollision;
nbFailure;
nbCCA;

for T slots do
    check nodes i ready to for CCA (delay == 0) and update nbCCA(i);

    if channel idle then
        decrement CW for nodes i;
        check nodes j from i with CW == 0;

        if more than one node j then
            increment nbCollision(j);

        else
            increment nbTransmission(j);

            set busyFor(j) to the transaction length in slots;

        end
    else
        % Channel busy

        check nodes j transmitting (busyFor(j) > 0);

        for node transmitting j do
            decrement BusyFor(j);

            if BusyFor(j) == 0 then
                reset NB(j), BE(j), CW(j) and delay(j) to initial values for new transmission;

            end
        end

        for nodes i doing CCA do
            update CW(i), NB(i), BE(i) according to CSMA/CA algorithm;
            check nodes k from i with NB(k) == macMaxCSMABackoffs;

            if maxBackoff reached for node k then
                increment nbFailure(k);
                reset NB(k), BE(k), CW(k) and delay(k) to initial values for new transmission;

            end
        end
    end

    for nodes i in delay line do
        delay(i) = delay(i) − 1;

    end
end
```

**Algorithm 1**: Pseudocode of the 802.15.4 CSMA/CA simulator.

### B. Model Validation

We evaluate the above derived expressions for $\phi$, $\alpha$ and $\beta$. In Figure 5 it can be seen that the value for $\phi$ as predicted by the model is very accurate for all scenarios. As expected, saturated traffic with a lower initial $aMinBE$ results in the highest sensing probability, since less time is spent in the delay line in this case (Section II). Also, we can see a small difference predicted by the model for acknowledged and unacknowledged traffic, which is due to the extra time spent waiting for or receiving the acknowledgment. When a periodic traffic scenario is modeled, the probability to start sensing in a given slot is decreased. In this scenario, we delay the next transmission attempt by $100$ slots which is equivalent to $32ms$.

We note that the model is very accurate for large $N$. For smaller $N$, the model is slightly different compared to the simulation since the approximations we used to compute $\beta$ as function of $P_{\text{send}}$, $P_{\text{netcol}}$ and $P_{\text{success}}$ are only valid for larger $N$. Next, we note that our basic assumption that the probabilities to start sensing of the different nodes is independent is more accurate for larger N. Indeed, in [12] it has been shown that, when the number of stations increases – i.e., large $N$ – stations tend to become independent.

The difference between the proposed model and the model in [9] is illustrated in Fig. 6 by comparing the results for $\beta$, which is the probability to sense busy during the second $CCA^2$. As explained above, the model in [9] fails to correctly model this $\beta$ since it does not correctly capture the fact that only a successful $CCA^1$ can result in a second $CCA^2$. As a result, their expression for $\beta$, and also the network equilibrium that depends on that expression, is not exact. It is clear in Figure 6 that the model proposed here matches the simulation results much more accurately and saturates for higher $N$. It can be seen that, when acknowledged uplink traffic is used, the value of $\beta$ is increased significantly which is the result of the probability $P_{\text{betaACK}}$ (Eq. 37). This contribution is however decreasing with the number of users, because the number of collisions increases significantly with $N$, and no acknowledgment is sent for collided packets.

This increased collision probability with $N$ is shown in Fig. 7. It is clear that the conditional network collision probability $P_{\text{netcol}}$ can be decreased by increasing the initial backoff exponent $aMinBE$ or by decreasing the traffic rate. The model matches simulation results for all scenarios. In that same figure, it can be seen that $\alpha$ is very large for saturated traffic conditions. As a result, it can be expected that a lot of energy is lost by sensing the channel busy which we will discuss in the next section. Obviously, both $\alpha$ and $P_{\text{netcol}}$ are increasing with $N$.

Finally, the probability to send a packet $P_s$ is plotted in Figure 8. The probability to send a packet is largest in the case no acknowledgment is used, and for saturated traffic. However, is should be clear that due to the large collision probability, the actual throughput per user is very low. For periodic unsaturated traffic, less packets are sent, but more throughput is achieved, as illustrated in Figure 8.

## V. Throughput and Energy Analysis

In this section we use the analytical model to study the energy and throughput behavior of both saturated and unsaturated 802.15.4 networks, and derive some design guidelines. The saturated case models a sensor network scenario in which an event is detected by many sensors that want to transmit the gathered information at the same time. The unsaturated case reflects a scenario with periodic monitoring intervals. In the saturated case, packets keep being generated so the system is never in one of the extra states $X_i$ of the Markov model depicted in Fig. 4. To model the unsaturated case, we introduce a fixed delay of $100$ slots before going into the first delay stage, representing a delay of $32ms$ after sending the previous packet (or failure to send it with probability $P_f$). This corresponds to $X1 = 100$ in the model.

We assume that collided packets, or packets that failed to be sent (with probability $P_f$), are not retransmitted since this simplifies the throughput expressions. It is clear that in this scenario, the network throughput for both saturated and unsaturated traffic conditions is given by the simple expression

$$S = LN\phi(1-\phi)^{N-1}(1-\alpha)(1-\beta) \times A = ALP_{\text{success}}. \tag{42}$$

where $A = \frac{80bit}{0.32ms}$ is a normalization constant to convert to $bps$. The throughput corresponds to the probability that a node in the network starts sensing alone, and has success during its channel assessments.

We now look how this net throughput is achieved. In Figure 8 it was already clear that in 802.15.4 networks a lot of packets get lost due to collisions, and the actual throughput can be very low. We compare the probability to start a transmission attempt (entering delay line), the probability to start sending, and the probability that a packet was sent successfully. We determine $P_{\text{attempt}}$, $P_{\text{send}}$ and $P_{\text{success}}$ and plot the breakdown for a set of scenarios in Figure 9. Only $P_{\text{attempt}}$ has not been determined yet in the model section:

$$P_{\text{attempt}} = P_{\text{send}} + N \times P_f \tag{43}$$

where the transmission failure $P_f$ for one node is given in Eq. 11. To compare the efficiency of the protocol, we convert the packet probabilities per slot to $bps$. According to the 802.15.4 physical layer in the $2.4GHz$ band, a maximum bitrate of $250kbps$ is possible.

Since a lot of packets are lost due to collisions, a lower initial Backoff Exponent $aMinBE$ is not optimal in this case. It can be seen in Fig. 9(b) that with $aMinBE = 2$ more packets are sent, but less net throughput is achieved and a lot of energy is wasted in collided transmissions. For sensor networks that are expected to generate a lot of traffic across the nodes simultaneously, $aMinBE$ should be as large as possible. It can be seen that, when a delay of $100$ slots or $32ms$ is introduced, the probability to start a transmission attempt is reduced significantly (Fig. 9). However, although the number of packet transmissions is lower, a higher throughput is achieved for high $N$ in the unsaturated traffic case compared to the saturated case (Fig. 9).

Next, we look at the energy consumption breakdown of 802.15.4 networks. We compute the percentage of time the transceiver is in each of its five states $Tx$, $Rx$, $CCA$, $Idle$ or $sleep$ (Table I), and multiply this time percentage by the power consumption of that state. We assume the transceiver is in $Tx$ mode when transmitting, in $Rx$ mode when waiting (interframe spaces) or receiving, in $CCA$ mode when performing CCA, in $idle$ mode in the exponential delay line, and in $sleep$ mode when not in the aforementioned cases. Typically in sensor networks, the transmit power is limited since the transmission range is limited. As a result, the power consumed during $Rx$ and $CCA$ is larger than the $Tx$ power consumption [6]. Following the power consumption characteristics of some of the 802.15.4 compliant products, we model the power consumption during the *sleep* state a factor $250000$ lower than $Rx$ state, which is significant [6]. Indeed, most 802.15.4 hardware has been optimized to meet this extremely low *sleep* power state. The $idle$ state is a second low power state, but transition to and from this state is possible in less than a backoff slot time. As a result, the power consumption is small but a lot higher than in deep $sleep$, more specifically a factor $50$ below $Rx$ [6]. We note that it could hence make sense to have an additional intermediate state in between the current $sleep$ and $idle$ states.

First, we are interested in the time spent scanning the channel. A delay line has been introduced in the 802.15.4 protocol to minimize the number of channel senses or the number of slots actually listening to the channel, as these are typically energy expensive. The probability of scanning the channel given a slot is (for each node):

$$P_{\text{sensing}} = \phi + \phi(1-\alpha). \tag{44}$$

Next, the time spent in $Rx$ and $Tx$ mode is known when we know $L$, $(t_{ACK} + L_{ACK})$ and $p_{\text{send}}$. The time in deep sleep depends in $X1$, $X2$ and $X3$ as expressed in Eq. 41. The idle time in the exponential delay line is then the remaining time. The resulting energy breakdown is plot in Fig. 10 where it can be noted that the energy needed to send a bit is very high in saturation, because of increased sensing and (collided) transmission probabilities. The energy spent in the $CCA$ mode is relatively high, despite the efforts of the 802.15.4 MAC design to minimize this energy. The energy expenditure due to the idle state during the exponential delay line is, as shown in Fig. 10, small. This is achieved thanks to the low power consumption of the *idle* state. 802.15.4 hardware design should hence, next to a very low power *sleep* and *idle* power mode, still optimize the power consumption of the $CCA$ mode. The energy spent while transmitting is however still quite high, because of the high (unsuccessful) packet transmission probability. In the model derived in [9], because of the higher $\beta$ in that model (Fig. 6), this transmission energy is however modeled to be decreasing for higher $N$ (Fig. 10).

In sensor networks, we are mostly interested in the average power needed to deliver packets at a certain rate. For that, we plot the energy per bit versus net throughput per node for a range of MAC parameter settings such as $aMinBE$, $NB$ and $L$ (for $aMinBE$ and $NB$ see Table I for their standardized range). Although the maximum bitrate is $250$ $kbps$, it can be seen in Fig. 11 that the actual achieved bitrate is much lower in reality. As expected, larger packet sizes ($120B$ in this case) perform the best, both in terms of power consumption and achieved throughput. For a lower load the energy per useful bit steadily decreases (Fig. 11). Lowering the initial $aMinBE$ results in a very small power consumption improvement for a given packet size. Since the power consumption in the idle state is typically very low for 802.15.4 transceivers, the gains are very small. Further, we note that the average power consumption increase with rate is mainly because of the increased *Tx* power but also because of increased *CCA* power consumption. Efforts to decrease that *CCA* power consumption are hence very useful.

## VI. Conclusion

In this paper, we have presented an analytical model for the medium access control layer in IEEE 802.15.4 standard. The model assumes a finite number of terminals and ideal channel conditions. The validity of the analytical model is demonstrated by the fact that its predictions closely match the simulation results. We then use the analytical model to predict energy consumption and achieved throughput of saturated and unsaturated 802.15.4 networks, based on which some design guidelines can be derived. It is shown that for saturated networks, it is best to choose a larger exponential delay backoff. For unsaturated networks, smaller backoff values improve the energy consumption but these energy savings are very small. It is also shown that although the new CSMA/CA mechanism significantly decreases the time and hence energy spent to listen to the channel, this $CCA$ energy consumption still represents a major part of the total system energy consumption breakdown.

## VII. Acknowledgments

The first author would like to thank Yuuichi Aoki for writing the simulation code for acknowledged uplink traffic and showing the need to extend the model to this special but non-trivial case.

## References


[1] J.M. Rabaey et al, *PicoRadio supports ad hoc ultra low power wireless networking*, IEEE Computer, vol.33, pp. 42-48, 2000.

[2] LAN-MAN Standards Committee of the IEEE Computer Society, *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE, 2003

[3] S. C. Ergen, *ZigBee/IEEE 802.15.4 Summary*, http://www.eecs.berkeley.edu/ csinem/academic/publications/zigbee.pdf

[4] J. Zheng and M. J. Lee, *Will IEEE 802.15.4 Make Ubiquitous Networking a Reality?: A Discussion on a Potential Low Power, Low Bit Rate Standard*, IEEE Communications Magazine, June 2004.

[5] G. Lu, B. Krishnamachari and C. S. Raghavendra, *Performance Evaluation of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Networks*, Workshop on Energy-Efficient Wireless Communications and Networks (EWCN '04), April 2004.

[6] B. Bougard, F. Catthoor, D. C. Daly, A. Chandrakasan and W. Dehaene, *Energy Efficiency of the IEEE 802.15.4 Standard in Dense Wireless Microsensor Networks: Modeling and Improvement Perspectives*, Proc. Design Automation and Test in Europe Conference and Exhibition, pp.196-201, March 2005.

[7] J. Misic, V. B. Misic and S. Shafi, *Performance of IEEE 802.15.4 Beacon-enabled PAN with Uplink Transmissions in Non-saturation Mode - Access Delay for Finite Buffers*, Proc. First International Conference on Broadband Networks, pp. 416-425, October 2004.

[8] J. Misic, S. Shafi and V. B. Misic, *The Impact of MAC Parameters on the Performance of 802.15.4 PAN*, Ad hoc Networks, vol. 3(5), pp. 509-528, September 2005.

[9] T.R. Park, T.H. Kim, J.Y. Choi, S. Choi, and W.H. Kwon, *Throughput and energy consumption analysis ofIEEE 802.15.4 slotted CSMA/CA*, Electronics Letters, September, 2005.

[10] G. Bianchi. *Performance Analysis of the IEEE 802.11 Distributed Coordination Function.* IEEE Journal on Selected Areas in Communications, vol.18, March 2000.

[11] LAN-MAN Standards Committee of the IEEE Computer Society, *Wireless LAN medium access control(MAC) and physical layer(PHY) specification*, IEEE, New York, NY, USA, IEEE Std 802.11-1997 edition, 1997

[12] Mustafa Ergen, *I-WLAN: Intelligent-Wireless Local Area Networking*, December 2004, UC Berkeley, PhD Thesis.

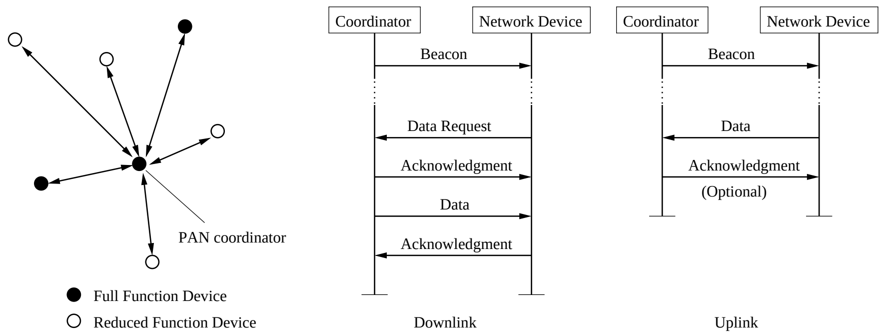



Fig. 1. The considered scenario for uplink and acknowledged uplink communication using beacon-enabled CSMA/CA.

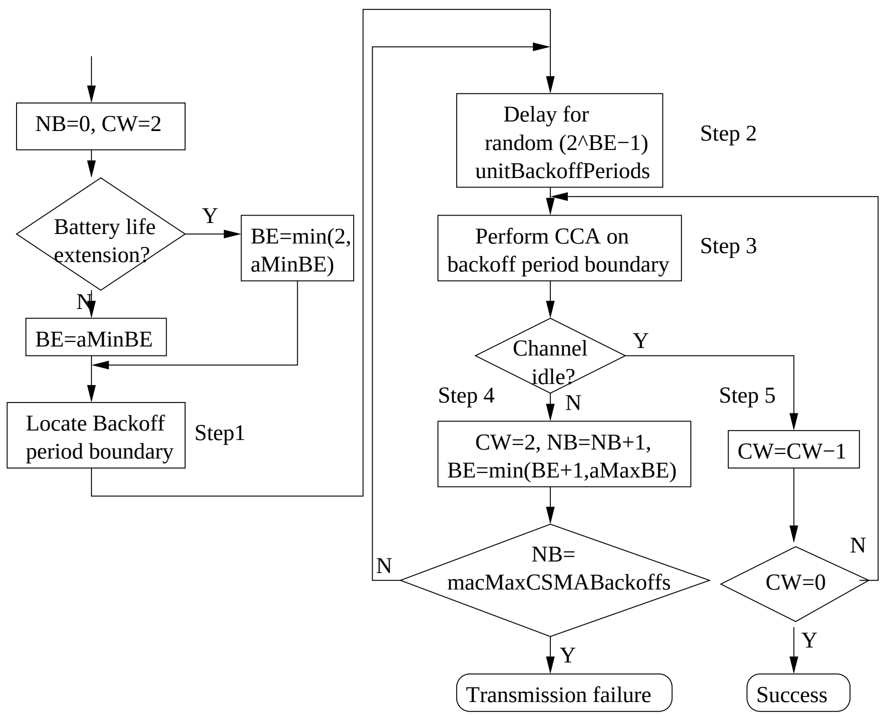
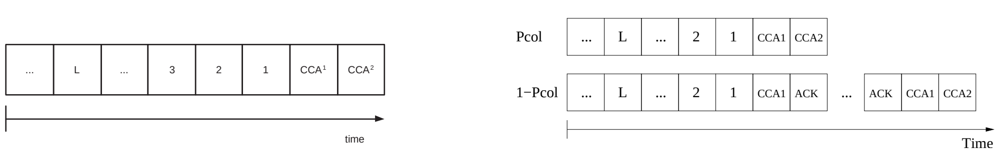



Fig. 2. Backoff mechanism for 802.15.4 CSMA




(a) Unacknowledged uplink.

(b) Acknowledged uplink

Fig. 3. Slot timing for the derivation of $\beta$, in case of an acknowledged or unacknowledged transmission.

TABLE I

THE DIFFERENT PARAMETERS USED FOR THE MODEL.

| | | |
|---|---|---|
| **Packet timings:** | $L_{\text{Header}} = 2slots$ | $L_{\text{data-bytes}} = 120Bytes$ |
| | $A = \frac{80bit}{0.32ms} = \frac{80bit}{slot}$ | $L_{ACK-Byte} = 11Bytes$ |
| | $\lceil L_{ACK} \rceil = 2slots$ | $\lfloor t_{ACK} \rfloor = 1slot$ |
| | $L = L_{\text{Header}} + \frac{L_{\text{data-bytes}}}{A} = 14slots$ | |
| **Variable parameters:** | **default:** | **range:** |
| | $NB = macMaxCSMABackoffs = 5$ | [0..5] |
| | $aMinBE = 3$ | [0..3] |
| **Fixed parameters:** | $aMaxBE = 5$ | $CW = 2$ |
| **Power states:** | *Rx* = $40mW$ | *CCA*=$40mW$ |
| | *Tx*= $30mW$ | *Idle* =$0.8mW$ |
| | *Sleep* = $0.16\mu W$ | |

TABLE II

THE DIFFERENT SYMBOLS USED FOR THE MODEL.

| | |
|---|---|
| $\phi$ | probability to start sensing |
| $\alpha$ | probability to sense busy first time |
| $\beta$ | probability to sense busy second time |
| $L, \dots, 1, CCA^1, CCA^2$ | time with respect to the sensing node $N$ |
| $M_t(s) = x$ | probability that at least one node of the medium has $s(t) = x$ |
| $M_t(c) = x$ | probability that at least one node of the medium has $c(t) = x$ |
| $S_t^n(s) = x$ | probability that node $n$ has $s(t) = x$ |
| $S_t^n(c) = x$ | probability that node $n$ has $c(t) = x$ |
| $P_{\text{send}}$ | probability that at least one node is sending |
| $P_{\text{netcol}}$ | conditional probability that at least one node is experiencing a collision |
| $P_{\text{loss}}$ | probability that at least one node is experiencing a collision |
| $P_{\text{success}}$ | probability that at least one node is sending successfully |
| $P_{\text{attempt}}$ | probability that at least one node is starting a transmission attempt |
| $P_{\text{f}}$ | probability that a node experiences a transmission failure |
| $P_{\text{s}}$ | probability that a node starts sending |
| $P_{\text{c}}$ | conditional probability that a node is experiencing a collision |
| $P_{\text{sensing}}$ | probability that a node is sensing |
| $X_1, X_2, X_3$ | Slots introduced to model periodic traffic |

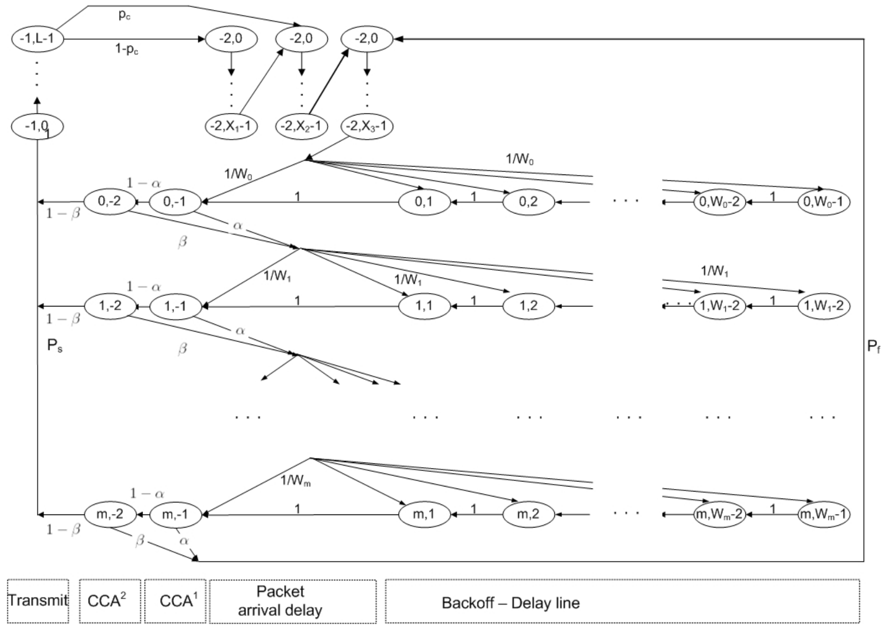



Fig. 4. Markov Model for IEEE 802.15.4

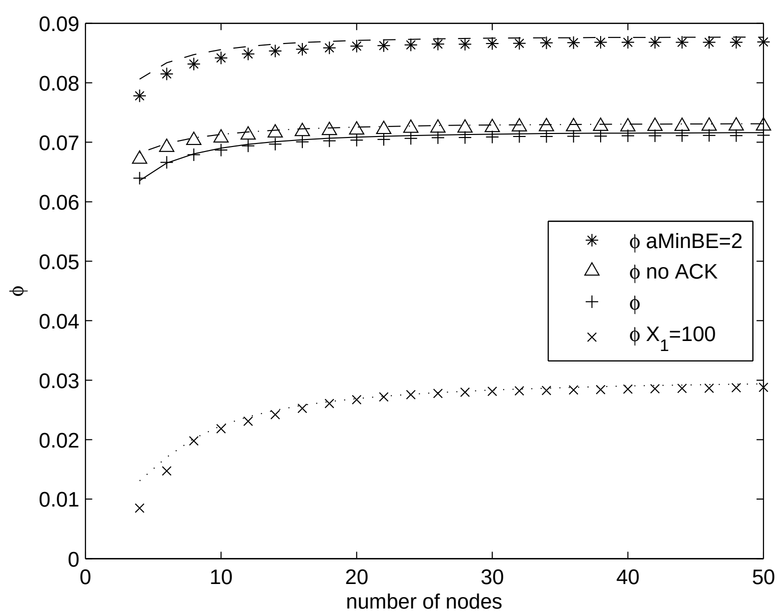



Fig. 5. Probability to start sensing $\phi$ for some saturated networks and for periodic traffic.

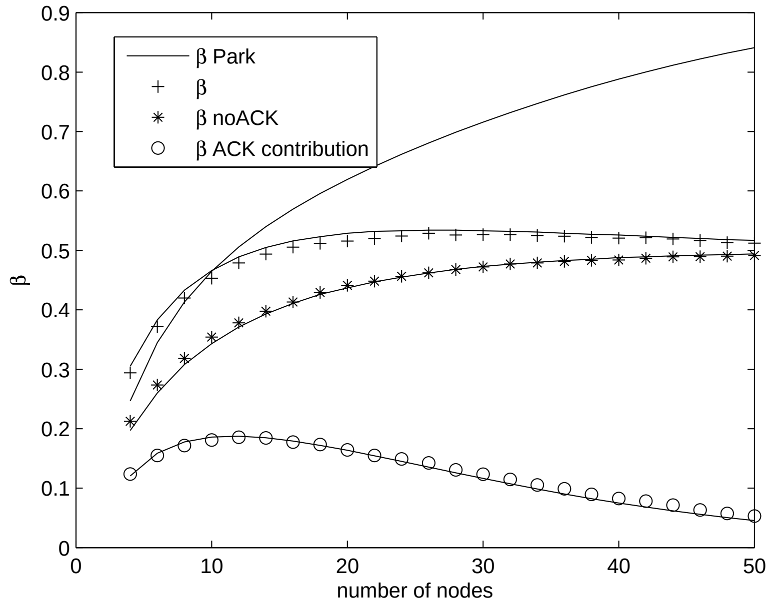



Fig. 6. Conditional probability $\beta$ to sense busy when sensing the second slot.

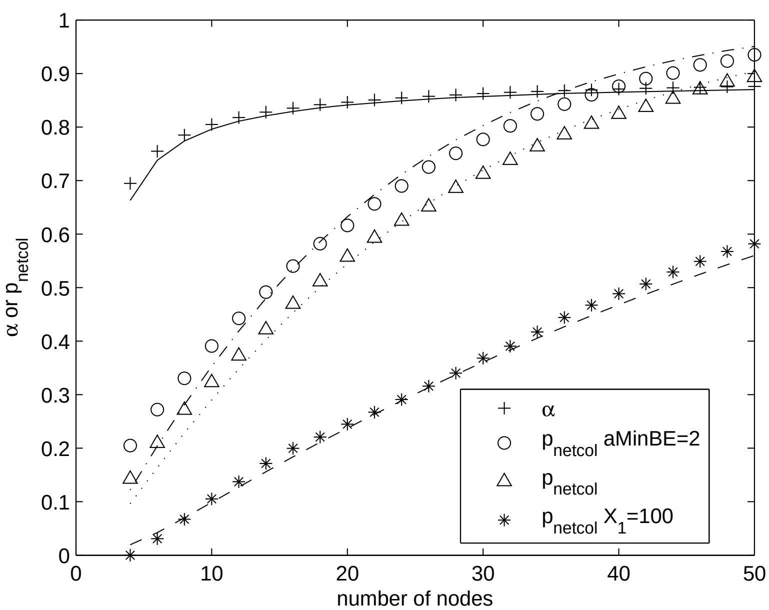



Fig. 7. Conditional probability $\alpha$ to sense busy when sensing the first slot slot. Conditional network collision probability: probability $P_{\text{netcol}}$ that a transmission going on in the network is a collision.

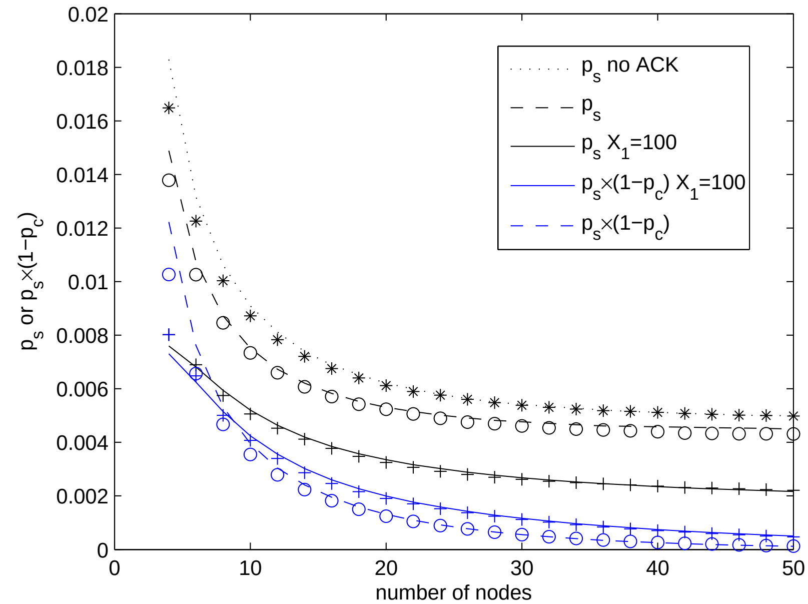
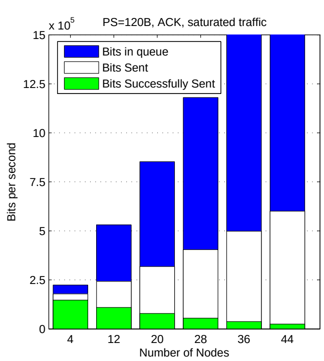
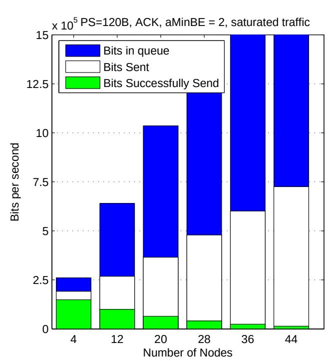


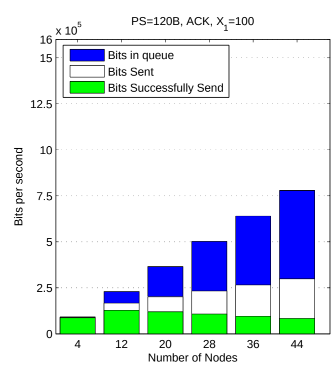

Fig. 8. Probability to start sending, and the probability to send successfully.




(a) No delay between transmit attempts




(b) No delay between transmit attempts, lower initial Backoff Exponent




(c) $32ms$ delay between transmit attempts

Fig. 9. Transmit rate breakdown: number of packets generated - number of packets sent - number of packets successfully sent

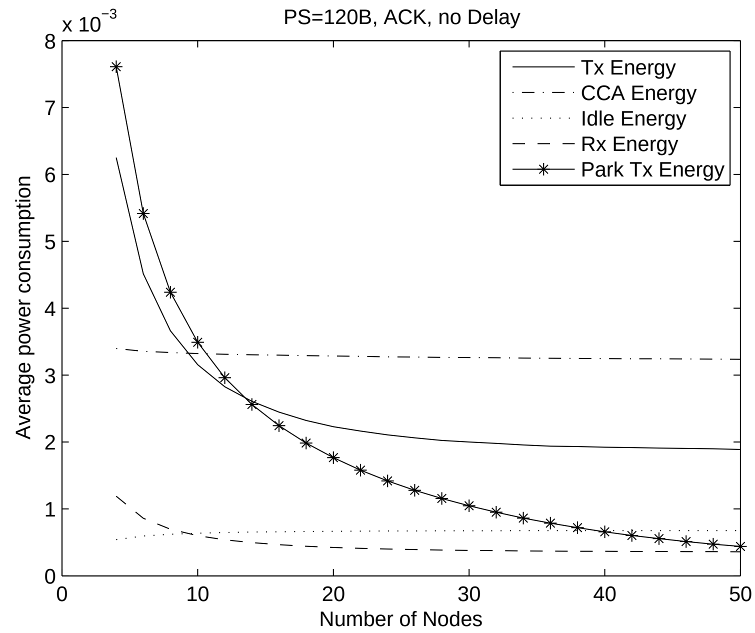



Fig. 10. Energy consumption during the different power states as function of the number of saturated users in the network.

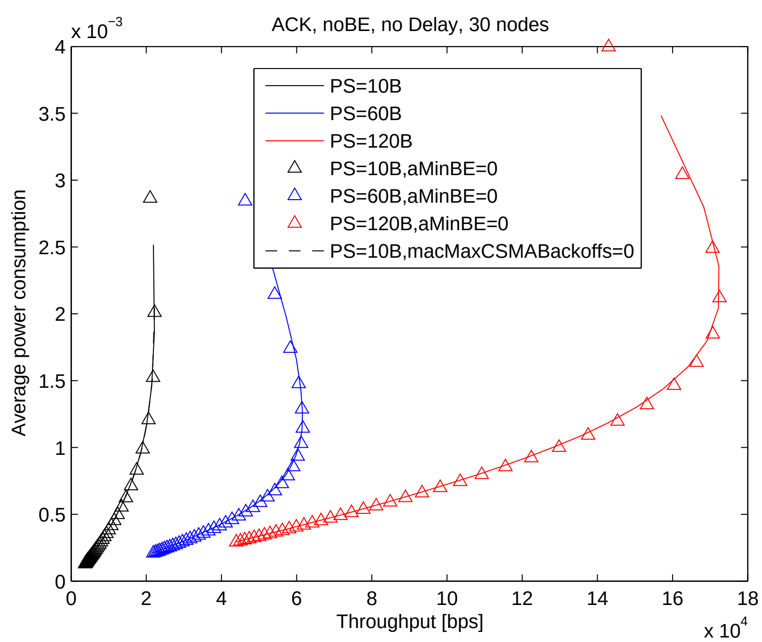



Fig. 11. Power consumption versus net throughput for different $aMinBE$ parameter settings.