Enhancing Channel Contention Efficiency in IEEE 802.15.4 Wireless Networks

Yi-Hua Zhu *, Luming Jia and Yufan Zhang

The School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China; jialuming953@163.com (L.J.); 1112012016@zjut.edu.cn (Y.Z.)
* Correspondence: yhzhu@ece.ubc.ca or yhzhu@zjut.edu.cn

Abstract: Numerous Internet of Things (IoT) devices adopt the IEEE 802.15.4 standard, which targets low data rate wireless networks. With the explosive growth in the use of IoT devices, it is essential to design effective and efficient channel access schemes for the 802.15.4 networks. In order to improve channel contention efficiency (CCE), which is defined as the number of times of successfully gaining the channel per unit of backoff time whereby throughput is improved, the scheme of enhancing channel contention efficiency (ECCE) has been proposed to jointly optimize the three key parameters of macMinBe, macMaxBe and macMaxCsmaBackoffs in the carrier sense multiple access with collision avoidance (CSMA-CA) mechanism in the 802.15.4 standard. A novel Markov chain was developed to model the CSMA-CA mechanism, which yielded the expected number of failures in gaining the channel, the expected number of backoff periods and the expected number of backoffs when a node intended to transmit a packet. These statistics resulted in CCE. An optimization problem that maximized the CCE with respect to the above-mentioned three key parameters was formulated. The solution to the optimization problem led to the optimal parameter values, which were applied in the ECCE scheme. The simulation results show that the proposed ECCE scheme outperformed the CSMA-CA mechanism in terms of CCE, delay and throughput.

Keywords: IEEE 802.15.4 standard; CSMA-CA; IoT; channel access efficiency

1. Introduction

The IEEE 802.15.4 standard, initiated in 2006, defines the physical (PHY) layer and medium access control (MAC) sublayer specifications for low data rate wireless connectivity for nodes with no battery or with very limited battery consumption [1]. The standard was updated in 2011 [2], 2015 [3] and 2020 [4], i.e., the last version of the IEEE 802.15.4 standard was released in 2020.

Contemporary Wireless Sensor Networks (WSNs) are main components of the Internet of Things (IoT) and as such, adopt the IEEE 802.15.4 standard in both MAC and PHY layers [5]. Nowadays, with the explosive growth of the number of devices injected into the IoT, IoT devices operating with the 802.15.4 standard face increasingly serious competition in gaining the channel. Therefore, it is important to enhance the channel contention efficiency for 802.15.4 wireless networks.

To date, there have been many studies on channel contention in 802.15.4-based networks in the literature. These works take channel access time, clear channel assessment (CCA) results, the state of busy or idle channels, packet collision, transmission failure rate and other factors into account in order to improve throughput, reduce delay, enhance energy efficiency, ensure high reliability or provide fair channel access.

In fact, in the IEEE 802.15.4 standard, the carrier sense multiple access with collision avoidance (CSMA-CA) channel access mechanism was introduced for the network nodes to contend for the channel. The CSMA-CA mechanism is classified into two categories: slotted CSMA-CA and unslotted CSMA-CA. This paper focuses on the former. There are three
key parameters in the slotted CSMA-CA mechanism that considerably impact channel contention efficiency, namely $\text{macMinBe}$, $\text{macMaxBe}$ and $\text{macMaxCsmaBackoffs}$. Unfortunately, the IEEE 802.15.4 standard only defines the ranges of these parameters, leaving their optimal values unresolved. As a result, the nodes in IEEE 802.15.4 standard-based networks are faced with the dilemma of how to set appropriate values for these parameters. Hence, it is desirable to design a scheme that enables a node to set the optimal values of these parameters so that channel contention efficiency is improved. This was the motivation for this paper. To the best of our knowledge, this is the first study that jointly optimizes the three key parameters to improve channel contention efficiency (CCE).

The main contributions of the paper are as follows:

- We propose the scheme of enhancing channel contention efficiency (ECCE), aiming to improve throughput and reduce delay through maximizing the CCE, which is defined as the number of successful gaining the channels per unit of backoff period;
- We developed a novel Markov chain to model the CSMA-CA, which yielded the following statistics: the expected number of failures in gaining the channel per packet, the expected number of backoff periods and the expected number of backoffs that a node experiences for transmitting a packet. These statistics were used to calculate the CCE. In addition, we formulated an optimization problem that maximized the CCE with respect to the aforementioned three key parameters in the CSMA-CA mechanism. The solution to the optimization problem leads to the optimal values of the parameters, which are applied in the proposed ECCE scheme;
- The simulation results show that the proposed scheme outperformed the CSMA-CA mechanism in terms of CCE, throughput and delay.

The remainder of this paper is organized by follows. The related works are surveyed in Section 2. In Section 3, we introduce the background and present the definition of the problem. In Section 4, we propose the ECCE scheme with its Markov model and formulate the optimization problem. In Section 5, we evaluate the performance of the proposed scheme via simulation. We conclude the paper in Section 6.

2. Related Work

Channel access that is based on the CSMA-CA has been intensively investigated within the research community. Z. Tao et al. [6] used a Markov chain to model the CSMA-CA mechanism in IEEE 802.15.4 networks and pointed out that a slight modification to the protocol can considerably improve throughput and delay. J. Abegunde et al. [7] presented a game theory-based solution to 802.15.4 channel contention, which reduces energy consumption and increases fairness. Z. Zhang et al. [8] applied a priority mechanism and virtual carrier sensing to the traditional CSMA-CA algorithm to improve throughput and reduce energy consumption. F. Shu et al. [9] analyzed the throughput of the energy conserving CSMA-CA under saturated and periodic traffic conditions. Y. Zhang et al. [10] studied the idle period after the last collision and proposed a mechanism to save power and improve throughput. C. Wang et al. [11] showed that the optimization of channel access time can reduce energy consumption. In fact, channel access time, or the backoff periods in the CSMA-CA mechanism, also affects throughput, delay, etc. Some parameters in the CSMA-CA mechanism that affect channel access time have been investigated. S. Brienza et al. [12] optimized them in terms of reliability and energy efficiency. J. Y. Ha et al. [13] proposed an optimization scheme to adjust the backoff exponent that was based on the clear channel assessment (CCA) results, which improves throughput and energy efficiency of the network. Y. Zhang et al. [14] presented a blind adaptive access parameter optimization algorithm to improve energy efficiency. B. Khandish et al. [15] adjusted the minimum value of the random backoff by using CCA results to improve network throughput. J. He et al. [16] studied the impact of the parameters of the slotted CSMA-CA mechanism on throughput by using two Markov chains. I. Bouazzi et al. [17] dynamically adjusted MAC parameters, based on the length of each sensor queue, and analyzed delay and throughput. G. Boudour et al. [18] proposed an algorithm that adjusts the competition parameters according to the observed busy channel
and transmission failure rate. S. Moulik et al. [19] used a superframe duration–beacon interval ratio in an analysis model, which shows that delay can be reduced by 35 percent by adjusting some MAC parameters. Kanafer et al. [20] presented the adaptive backoff algorithm (ABA), in which the size of contention window is adaptively determined based on the collisions experienced by the nodes. Faridi et al. [21] scrutinized the assumptions made in the prevalent Markovian models and singled out the assumptions that influence performance metrics, including throughput, delay, power consumption, collision probability and packet discard probability. Azdad et al. [22] focused on understanding MAC parameters and manipulating the configuration of MAC parameters to support higher data rates in wireless body area networks (WBANs). Moulik et al. [23] presented a mechanism to ensure the reliable transmission of critical packets in WBANs that adopt the IEEE 802.15.4 standard. Alaparthi et al. [24] studied the suitability of CSMA-CA for WSN. Nabila et al. [25] enhanced the backoff strategy for the slotted CSMA-CA defined in the IEEE 802.15.4 standard in order to provide nodes with fair channel access. Singh et al. [26] proposed the “device registration” algorithm to support the applications that generate time-sensitive and sporadic data by modifying the parameters within the framework of the existing MAC. Aboubakar et al. [27] proposed an efficient approach for configuring the IEEE 802.15.4 MAC parameters, in which the predictive feature of machine learning algorithms is used to determine the IEEE 802.15.4 MAC parameters. Musaddiq et al. [28] applied a reinforcement learning (RL) mechanism to efficiently handle the channel access mechanisms. Alimorad et al. [29] presented a numerical MAC optimization algorithm for beacon-enabled IEEE 802.15.4-based WBANs to optimize delay, reliability and energy trade-offs, based on the existing traffic and QoS requirements.

In contrast to the existing schemes, our ECCE targets the maximum CCE by optimizing the three key parameters of $macMaxCsmaBackoffs$, $macMinBe$ and $macMaxBe$ [4] in the CSMA-CA mechanism to reduce delay and improve throughput.

### 3. Background and Problem Definition

In a slotted 802.15.4 network, each node abides by the CSMA-CA mechanism in each data transmission trial. With the CSMA-CA mechanism, a node maintains the three key variables of $NB$, $CW$ and $BE$, which stand for the number of backoffs, contention window and backoff exponent, respectively. The variable $NB$ holds the number of CSMA-CA algorithms being conducted to back off the current transmission. The variable $CW$ defines the number of successive backoff periods in which the channel is clear before the transmission can initiate. The variable $BE$ sets $2^{BE} - 1$ backoff periods, from which the node randomly picks one [3].

The procedure of the slotted CSMA-CA channel access mechanism is shown in Figure 1. In the figure, $macMaxBe$, $macMinBe$ and $macMaxCsmaBackoffs$ are the three key parameters defined in the MAC layer, which represent the maximum of $BE$, the minimum of $BE$ and the upper bound on the number of CSMA-CA algorithms being conducted, respectively. In the procedure, variables $NB$ and $CW$ are initialized to zero and two, respectively. Then, variable $BE$ is set to the lesser of $macMinBe$ and two, depending on battery life extension information. After locating or aligning the backoff period boundary, the node randomly picks a number, say $x$, from the set $\{1, 2, \ldots, 2^{BE} - 1\}$. Then, the node keeps silent until the boundary of the $x$-th and the $(x + 1)$-th backoff periods, when the node performs a clear channel assessment (CCA) to see whether the channel is idle or not. If it is, the node checks whether the beginning of the next backoff period is still idle. Only when $CW$ (i.e., two) successive backoff periods are idle can the node transmit its data. Otherwise, the node prepares the next random backoff period after resetting $CW$ to two and increasing $NB$ by one, except when $BE$ reaches its limitation $macMaxBe$.

Obviously, an increase in $BE$ may increase the node’s waiting time before gaining the channel because the node picks a random number from a set with nearly twice as many backoff periods as the previous one. It should be noted that $NB$, $CW$ and $BE$ are reset when-
ever the node succeeds in gaining the channel or fails after reaching the \textit{macMaxCsmaBackoffs} number of random backoffs.

Although the ranges of the \textit{macMinBe}, \textit{macMaxBe} and \textit{macMaxCsmaBackoffs} parameters are defined in the IEEE 802.15.4 standard, the nodes operating within this standard are faced with the problem of lacking the knowledge of how to set the optimal values for these parameters. This problem will be addressed in the next section.

![Figure 1. The procedure of the CSMA-CA algorithm [2,3].](image)

4. The Proposed Scheme with Mathematical Model

4.1. The ECCE Scheme

When a node intends to deliver a packet, it contends for the channel via a CSMA-CA mechanism that incorporates the three key parameters of \textit{macMaxCsmaBackoffs}, \textit{macMinBe} and \textit{macMaxBe}.

In the proposed ECCE scheme, when the node performs a CCA, the node calculates the probability of the CCA sensing a busy channel, denoted by $h$, according to the following procedure. The node uses $\mathcal{N}$ to hold the total number of times that the node performs a CCA and $\mathcal{N}_b$ to hold the number of times that the node has sensed a busy channel with the CCA. The probability $h$ is initialized to zero. Each time the node conducts a CCA, $\mathcal{N}$ is always increased by one and $\mathcal{N}_b$ is only increased by one when the CCA senses a busy channel. Then, the node updates $h = \mathcal{N}_b / \mathcal{N}$.

In the ECCE scheme, when a node fails in gaining the channel, it replaces the three key parameters in the CSMA-CA with the solution of the optimization problem in (32) in Section 4.4, which aims to maximize the CCE as defined in Section 4.3.

4.2. Markov Chain for the CSMA-CA Mechanism

In 802.15.4 wireless networks, a node transmits packets one after another. It can be seen from the CSMA-CA mechanism in Figure 1 that, for a given packet to be delivered, the node attempts to contend for the channel at most $K$ times under the CSMA-CA mechanism,
where \( K = \text{macMaxCsmaBackoffs} \). With the CSMA-CA mechanism, we define the node’s state as follows:

- State “\( A(i, j) \)” represents that the node intends to gain the channel on the \( i \)-th attempt (i.e., it fails in the previous \( i - 1 \) attempts) and it performs a CCA for the first time at the beginning of the \( j \)-th backoff period, where \( i \in \{1, 2, \ldots, K\}, j \in \{1, 2, \ldots, n_i\} \). Here, from Figure 1, we have:
  \[
  n_i = \min\{2^{X_0+i-1} - 1, 2^{X_1} - 1\}
  \]
  \[
  (i = 1, 2, \ldots, K),
  \]
  where \( X_0 = \text{macMinBe} \) and \( X_1 = \text{macMaxBe} \). From the IEEE 802.15.4 standard [3], \( X_0 \in \{3, 4, \ldots, 8\} \) is defaulted to five, \( X_1 \in \{1, 2, \ldots, X_1\} \) is defaulted to three and \( K \in \{1, 2, 3, 4, 5\} \) is defaulted to four. In addition, we introduce:
  \[
  s_K = \sum_{i=1}^{K} n_i.
  \]

- State “\( B(i, j) \)” represents that the node intends to gain the channel on the \( i \)-th attempt and it performs a CCA for the second time at the beginning of the \( (j + 1) \)-th backoff period (this is the equivalent of performing a CCA at the end of \( j \)-th backoff period because the CCA is conducted at the boundary of two neighboring backoff periods), where \( i \in \{1, 2, \ldots, K\}, j \in \{1, 2, \ldots, n_i\} \);

- State “\( C_0 \)” represents that the node has prepared channel contention for transmitting the packet;

- State “\( C_i(i \geq 1) \)” represents that the channel is not idle according to the CCA on the \( i \)-th attempt;

- State “\( S \)” represents that the node has succeeded in gaining the channel;

- State “\( H \)” represents that the node has halted channel contention as the number of attempts reaches its maximum.

The transitions of the above-defined states are shown in Figure 2, where a circle with the integers \( i \) and \( j \) inside stands for state \( A(i, j) \), a dotted ellipse with the integers \( i \) and \( j \) stands for state \( B(i, j) \), a square with \( C_i \) inside stands for state \( C_i \), a hexagon with the letter “\( S \)” stands for state \( S \) and a hexagon with the letter “\( H \)” stands for state \( H \). From the state transition diagram in Figure 2, we calculated the state transition probabilities as follows:

\[
\begin{align*}
\Pr\{A_{ij}|C_{i-1}\} &= \frac{1}{n_i},
\quad i = 1, 2, \ldots, K, \ j = 1, 2, \ldots, n_i; \\
\Pr\{B_{ij}|A_{ij}\} &= 1 - h_i, \quad i = 1, 2, \ldots, K, \ j = 1, 2, \ldots, n_i; \\
\Pr\{C_i|A_{ij}\} &= h_i, \quad i = 1, 2, \ldots, K - 1, \ j = 1, 2, \ldots, n_i; \\
\Pr\{H|A_{K,j}\} &= h_j, \quad j = 1, 2, \ldots, n_K; \\
\Pr\{C_i|B_{ij}\} &= h_i, \quad i = 1, 2, \ldots, K - 1, \ j = 1, 2, \ldots, n_i; \\
\Pr\{H|B_{K,j}\} &= h_j, \quad j = 1, 2, \ldots, n_K; \\
\Pr\{S|B_{ij}\} &= 1 - h_i, \quad i = 1, 2, \ldots, K, \ j = 1, 2, \ldots, n_i; \\
\Pr\{C_0|H\} &= 1; \\
\Pr\{C_0|S\} &= 1.
\end{align*}
\]
As mentioned previously, $h$ is the probability of the CCA sensing that the channel is busy when the node tries to gain the channel.

Figure 2. The state transition diagram.

The state transition diagram is explained as follows. When the node intends to transmit a packet, then the node stays in state $C_0$ as it needs to contend for the channel and thus, it experiences a random backoff period in which there are $n_1$ integers $(1, 2, \ldots, n_1)$ with each being equally likely to be picked according to the CSMA-CA mechanism. Therefore, it has the probability of $1/n_1$ that the node transfers from state $C_0$ to state $A(i, j)$. Equivalently, Equation (3) holds. In the case of the node being in state $A(1, j)$, if the CCA indicates that the channel is idle, the node needs to perform the CCA again at the beginning of the next backoff period due to $CW = 2$, i.e., its state changes to $B(i, j)$. Otherwise, its state becomes $C_1$, which indicates that the node fails in gaining the channel on the first attempt. That is, $Pr\{B_1|A_1,j\} = 1 - h$ and $Pr\{C_1|A_1,j\} = h$ $(j = 1, 2, \ldots, n_1)$. Similarly, we can analyze the case when the node is in state $A(i, j)$ $(i > 1)$, which leads to Equations (4) and (5). In the case of the node being in state $A(K, j)$, if the channel is sensed to be busy, which happens
with probability of \( h \), it causes the node to be in state \( H \) (i.e., the node senses the channel is in use on the \( K \)-th attempt and thus, the node halts channel contention). This yields (6). In the case of the node being in state \( B(i, j) \), if the channel is sensed to be busy, which occurs with the probability of \( h \), it brings the node to either state \( C_i \) if \( i < K \) or state \( H \) if \( i = K \), whereas a clear channel brings the node to the success state \( S \). Thus, Equations (7)–(9) hold true. In the case of the node being in states \( S \) or \( H \), which indicate the completion of the channel contention for the current packet, the node’s state changes back to \( C_0 \), i.e., the node prepares to transmit the next packet in its transmission buffer. This is reflected in (10) and (11).

\[ A_{i,j}, B_{i,j}, C_{i}, H \text{ and } S \text{ denote the probabilities of the node being in states } a_{i,j}, b_{i,j}, c_{i}, p_{b} \text{ and } p_{s}, \text{ respectively. From Figure 2, after considering all states, we used the following equations for the Markov chain, according to the flow balance:} \]

\[
c_0 = p_s + p_h; \quad (12)
\]

\[
a_{i,j} = c_{i-1} \frac{1}{n_i}, \quad i = 1, 2, \ldots, K, j = 1, 2, \ldots, n_i; \quad (13)
\]

\[
b_{i,j} = (1-h)a_{i,j}, \quad i = 1, 2, \ldots, K, j = 1, 2, \ldots, n_i; \quad (14)
\]

\[
c_i = h(\sum_{j=1}^{n_i} a_{i,j} + \sum_{j=1}^{n_i} b_{i,j}), \quad i = 1, 2, \ldots, K - 1; \quad (15)
\]

\[
p_h = h(\sum_{j=1}^{n_K} a_{K,j} + \sum_{j=1}^{n_K} b_{K,j}); \quad (16)
\]

\[
p_s = (1-h)\sum_{i=1}^{K} \sum_{j=1}^{n_i} b_{i,j}. \quad (17)
\]

Obviously, we used the following normalization equation for all probabilities:

\[
p_s + p_h + \sum_{i=0}^{K-1} c_i + \sum_{i=1}^{K} \sum_{j=1}^{n_i} a_{i,j} + \sum_{i=1}^{K} \sum_{j=1}^{n_i} b_{i,j} = 1. \quad (18)
\]

From Equations (13)–(18), we calculated:

\[
A \bar{P} = \bar{e}, \quad (19)
\]

where \( \bar{P} \) and \( \bar{e} \) are vectors, with each having \( K + 2(K + 1) \) elements, and \( A \) is a square matrix with each column having the same dimension as vector \( \bar{P} \). Specifically, vector \( \bar{e} \) contains elements of 0 s, except for the last element of one, i.e., \( \bar{e} = (0, 0, \ldots, 0, 1)^T \). In addition, vector \( \bar{P} \) consists of all state probabilities as \( \bar{P} = (p_s, p_h, \bar{e}_1, \bar{e}_2, \ldots, \bar{e}_K, \bar{b}_1, \bar{b}_2, \ldots, \bar{b}_K)^T \), in which we define row vectors \( \bar{e}_i = (c_0, c_1, \ldots, c_{K-1}), \bar{a}_i = (a_{i,1}, a_{i,2}, \ldots, a_{i,n_i}) \) and \( \bar{b}_i = (b_{i,1}, b_{i,2}, \ldots, b_{i,n_i}) \), where \( i = 1, 2, \ldots, K \). Moreover:

\[
A = \begin{bmatrix}
\mathbb{I}_K & -\mathbb{I}_K & \cdots & -\mathbb{I}_K \\
\mathbb{I}_K & -\mathbb{I}_K & \cdots & -\mathbb{I}_K \\
\cdots & \cdots & \cdots & \cdots \\
\mathbb{I}_K & -\mathbb{I}_K & \cdots & -\mathbb{I}_K \\
-\mathbb{I}_K & \mathbb{I}_K & \cdots & \mathbb{I}_K \\
-\mathbb{I}_K & \mathbb{I}_K & \cdots & \mathbb{I}_K \\
\cdots & \cdots & \cdots & \cdots \\
-\mathbb{I}_K & \mathbb{I}_K & \cdots & \mathbb{I}_K \\
1 & 1 & \cdots & 1 \\
0 & 0 & \cdots & 0 \\
\end{bmatrix} \quad (20)
\]

In (20), the last row consists of all \( 1 \) s, which corresponds to (18). Here, we define \( \mathbb{I}_x \) as the row vector with \( x \) 1 s. In addition, we define \( \mathbf{0}_x \) as the row vector with \( x \) 0 s. Meanwhile,
we define \( \vec{0}_x \) and \( \vec{1}_x \) as the \( x \)-element column vectors consisting of 0 s and 1 s, respectively. The matrix blocks in (20) are introduced as follows. \( \vec{I}_x \) stands for the \( x \)-th order identity matrix (its diagonal elements are 1 s and the other elements are 0 s). Additionally, we calculated:

\[
\mathbb{B}_{2K \times K} = \begin{bmatrix}
\vec{i}_{n_1} & 0 & \cdots & 0 \\
0 & \vec{i}_{n_2} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \vec{i}_{n_K}
\end{bmatrix},
\]

(21)

\[
\mathbb{D}_{2 \times n_K} = \begin{bmatrix}
\vec{0}_{n_K} & \vec{0}_{n_K} \\
\vec{0}_{n_K} & \vec{1}_{n_K}
\end{bmatrix},
\]

(22)

\[
\mathbb{E}_{2 \times n_K} = \begin{bmatrix}
(1 - h)\vec{i}_{n_K} & (1 - h)\vec{i}_{n_K} \\
\vec{0}_{n_K} & \vec{0}_{n_K}
\end{bmatrix},
\]

(23)

\[
\mathbb{F}_{(K-1) \times K} = \begin{bmatrix}
\vec{0}_{K-1} & -\vec{i}_{K-1}
\end{bmatrix},
\]

(24)

\[
\mathbb{G}_{(K-1) \times K} = \begin{bmatrix}
h\vec{i}_{n_1} & \vec{0}_{n_2} & \cdots & \vec{0}_{n_{K-1}} & \vec{0}_{n_K} \\
\vec{0}_{n_1} & h\vec{i}_{n_2} & \cdots & \vec{0}_{n_{K-1}} & \vec{0}_{n_K} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\vec{0}_{n_1} & \vec{0}_{n_2} & \cdots & h\vec{i}_{n_{K-1}} & \vec{0}_{n_K}
\end{bmatrix}.
\]

(25)

From Equation (19), we obtained \( \tilde{\mathbf{P}} = h^{-1}\vec{\mathbf{e}} \), which contains all state probabilities. With the derived probabilities \( \tilde{b}_{i,j} \), from (17), we obtained the probability of successfully gaining the channel as follows:

\[
p_s(K, X_0, X_1) = (1 - h) \sum_{i=1}^{K} \sum_{j=1}^{n_i} \tilde{b}_{i,j},
\]

(26)

where the success probability is expressed as function of \( K \) and \( n_i \) is given in (1).

4.3. Statistics for the CSMA-CA Mechanism

Using the probabilities derived in the previous section, we could obtain some statistics under the CSMA-CA mechanism.

Firstly, the expected number of times that the node fails in gaining the channel for transmitting a packet was:

\[
N_f(K, X_0, X_1) = \sum_{i=1}^{K-1} i \left( \frac{c_i}{\sum_{j=1}^{K-1} c_j} \right) = \frac{\sum_{i=1}^{K-1} i c_i}{\sum_{j=1}^{K-1} c_j},
\]

(27)

where \( c_i / \sum_{j=1}^{K-1} c_j \) is the probability that the node senses a busy channel on the \( i \)-th trial, which is a function of \( X_0 \) and \( X_1 \).

Secondly, in the CSMA-CA algorithm, the node needs to conduct the CCA twice to check whether the channel is idle. On the \( i \)-th attempt to gain the channel, there are \( n_i \) backoff periods from which the node picks one with an identical probability of \( 1/n_i \). Thus, the node averagely suffers from \( (1 + 2 + \cdots + n_i)/n_i = (n_i + 1)/2 \) backoff periods before the first CCA. After the backoff, the node performs the first CCA. In the case of the first CCA sensing that the channel is idle, which happens with probability of
1 − h, the second CCA is conducted, which consumes one more backoff period regardless of whether the second CCA senses the idle channel or not. Hence, the double CCA consumes a time of (1 − h) × 1 + h × 0 = 1 − h backoff period, on average. As a result, on the i-th attempt, the expected number of backoff periods that the node experiences is

\[(n_i + 1)/2 + 1 - h = (n_i - 2h + 3)/2, \quad i = 1, 2, \ldots, K.\]

By noticing that the node attempts to gain the channel only in states \(C_0, C_1, \ldots, C_{K-1}\), we obtained the probability of the node conducting the i-th attempt as \(c_{i-1}/\sum_{j=0}^{K-1} c_j\), \(i = 1, 2, \ldots, K\).

Thus, the expected number of backoff periods that the node experiences for transmitting a packet was:

\[N_{bp}(K, X_0, X_1) = \sum_{i=1}^{K} \frac{c_{i-1}}{\sum_{j=0}^{K-1} c_j} \left(\frac{n_i - 2h + 3}{2}\right)\]
\[= \frac{\sum_{i=1}^{K} c_{i-1}(n_i - 2h + 3)}{2 \sum_{j=0}^{K-1} c_j}. \quad (28)\]

In addition, the expected number of backoffs that the node experiences for transmitting a packet, which is equal to the expected number of transmission trials, was:

\[N_b(K, X_0, X_1) = \sum_{i=1}^{K} \frac{c_{i-1}}{\sum_{j=0}^{K-1} c_j} = \frac{\sum_{i=1}^{K} ic_{i-1}}{\sum_{j=0}^{K-1} c_j}. \quad (29)\]

Thirdly, in order to transmit a packet, the node can use the channel once if it succeeds in gaining the channel, which occurs with probability \(p_s(\cdot)\), and it does not use the channel otherwise. Hence, the expected number of times that the node gains the channel successfully is \(p_s(\cdot) \times 1 + [1 - p_s(\cdot)] \times 0 = p_s(\cdot)\). By considering the expected time consumed by the node in contending for the channel as \(N_{bp}(\cdot)\sigma\), where \(\sigma\) is the duration of one backoff period set to aUnitBackoffPeriod, a parameter defined in the MAC layer of the IEEE 802.15.4 standard, we defined Channel Contention Efficiency (CCE) as:

\[\delta(K, X_0, X_1) = \frac{p_s(K, X_0, X_1)}{N_{bp}(K, X_0, X_1)\sigma}. \quad (30)\]

The CCE in (30) can be explained as the number of times that the node successfully gained the channel per unit of backoff time.

Using (26) and (28), we calculated the CCE to be as follows:

\[\delta(K, X_0, X_1) = \frac{2(1 - h) \sum_{j=0}^{K-1} c_j \sum_{i=1}^{K} b_{i,j}}{\sigma \sum_{i=1}^{K} c_{i-1}(n_i - 2h + 3)}. \quad (31)\]

4.4. Optimization Problem

Our scheme aims to maximize the CCE in (31). We formulated the optimization problem (OP) as follows:

\[
\begin{align*}
\text{max} & \quad \delta(K, X_0, X_1) \\
\text{w.r.t.} & : K, X_0, X_1 \\
\text{s.t.} & : \begin{cases} 
K \in \{1, 2, 3, 4, 5\}; \\
X_1 \in \{3, 4, \ldots, 8\}; \\
X_0 \in \{1, 2, \ldots, X_1\}. 
\end{cases} 
\end{align*}
\]

(32)

The constraints in the above optimization problem are from IEEE 802.15.4 standard [3]. Considering that the number of triple \((K, X_0, X_1)\)s that satisfy the constraints is quite small, we used an exhaustive search to find the solution to the optimization problem.
5. Performance Evaluation

In this section, we present the simulation results to evaluate the effectiveness of the proposed ECCE. The simulation programs were written in MATLAB.

We assumed that there are \( m \) nodes in the 802.15.4 WSN. Due to the focus of this paper being on channel contention through allowing the nodes to set the three key parameters of \( \text{macMinBe} \), \( \text{macMaxBe} \) and \( \text{macMaxCsmaBackoffs} \) for the CSMA-CA mechanism, we did not consider the topology of the WSN. Instead, each of the \( m \) nodes had a random traffic, i.e., the arrival time of a data packet was a random variable; and a packet arrival triggered the node to contend for the channel. We let each node apply the ECCE scheme to dynamically adjust its own parameters for \( \text{macMinBe} \), \( \text{macMaxBe} \) and \( \text{macMaxCsmaBackoffs} \) through the optimization problem in (32).

Considering that Gamma distribution has two parameters that can be adjusted to fit most distributions, we assumed that data packet arrivals at a node follow a Gamma distribution with a shape parameter \( \alpha \) and scale parameter \( \beta \), which has the probability density function of:

\[
 f(x) = \begin{cases} 
 \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}}, & x \geq 0; \\
 0, & x < 0. 
\end{cases} 
\]  

(33)

Here, \( \Gamma(\alpha) = \int_0^\infty e^{-y} y^{\alpha-1} dy \). The mean and variance of the above distribution were \( \alpha \beta \) and \( \alpha \beta^2 \), respectively. Additionally, MATLAB adopted the probability density function in (33).

In addition, we used \( \text{Gam}(\alpha, \beta) \) to represent the function that produced a random number, obeying the Gamma distribution with parameters of \( \alpha \) and \( \beta \).

We let \( \sigma \) be the duration of backoff period, i.e., \( \sigma = a \text{InitBackoffPeriod} \), which is a parameter defined in the MAC layer of the IEEE 802.15.4 standard.

The process of simulation proceeded microsecond by microsecond. The logic of mimicking the ECCE is shown in Figure 3. The notations in the figure are defined as follows: \( T \) and \( T_{\text{tot}} \) stand for current time and the total simulation time, respectively; and \( T_{\text{arr}} \), \( T_{\text{bf}} \) and \( T_{\text{tx}} \) represent the packet arrival time, backoff end time and transmission end time, respectively.

The simulation logic can be outlined as follows:

1. Initialize the simulation parameters and variables;
2. Advance the time on the basis of microseconds. At each microsecond, first check whether a packet has arrived. If yes, keep the packet in the buffer and then consider contending for the channel for transmission in the case of the buffer being empty. Then, check whether the node is in backoff or not. In the case when a backoff ends, the node abides by the traditional CSMA-CA mechanism, starting from the CCA. Lastly, check whether the node is transmitting a packet. The completion of the transmission resets the relative parameters.

In the simulation process, our optimization problem in (32) was applied when the node fails in gaining the channel so that the MAC parameters in the CSMA-CA were adjusted to adapt to channel competition.

We compared the ECCE to the CSMA-CA scheme for the 802.15.4 standard and the ABA proposed in [20] in terms of CCE, delay, percentage of reduced delay (PoRD), throughput, percentage of increased throughput (PoIT) and collision probability. Here, the PoRD is defined as the ratio \((a_1 - a)/a\), where \( a \) stands for the delay of ECCE and \( a_1 \) is the delay of the CSMA-CA or the ABA; and PoIT is \((b - b_1)/b_1\), where \( b \) represents the throughput of ECCE and \( b_1 \) is the throughput of the CSMA-CA or the ABA. The following results are from a simulation with \( 10^4 \) time slots, with each having length of 1160 \( \mu \)s.
In order to clearly observe the CCE in our ECCE, we set \( m = 10, 12, \ldots, 30 \) and \( \beta = 0.01, 0.02, 0.04 \) (in seconds), and fixed \( \alpha = 1 \), which yielded the simulation results in Figure 4, where the CCE is the average over the number of nodes. It can be clearly seen from Figure 4 that the ECCE had a higher CCE than that of the CSMA-CA and the ABA. This agrees with the aim of our ECCE to maximize the CCE. The same settings for the parameters \( m, \alpha \) and \( \beta \) led to the delay, throughput, PoIT and PoRD shown in Figures 5–8. From Figure 5, we can observe the following: (1) the proposed ECCE scheme outperformed the CSMA-CA and the ABA in throughput; and (2) the throughput in the ECCE nearly kept steady with the growth of the number of nodes (i.e., \( m \)). In Figure 6, we can observe that the ECCE scheme also outperformed the CSMA-CA and the ABA in delay and that the packet delay under the ECCE scheme gradually increased with the number of nodes, due to the fact that channel contention becomes more serious when more nodes join in contending for the channel. The reason that the proposed ECCE scheme had better throughput and delay than the other two schemes is that the ECCE scheme could adapt to the contention situation by setting the three key parameters in the CSMA-CA scheme to their respective optimal values via the optimization problem in (32). It can be seen from Figures 7 and 8 that PoIT and PoRD differed with different values of parameter \( \beta \). The average interval of packet arrivals obeying the probability density function in (33) was \( \alpha \beta \). As a result, in the case of \( \alpha \) being fixed (we set \( \alpha = 1 \)), growth in \( \beta \) led to fewer packet arrivals in the given time period and thus, throughput decreased. Figure 7 indicates that, for a given \( m \), as \( \beta \) grew, the PoIT comparing the ECCE to the ABA clearly reduced whereas the PoIT comparing the ECCE to the CSMA-CA varied slightly. Figure 8 shows that variation in \( \beta \), i.e., traffic variation, did
not affect the PoRD too much. This is because the delay of an incoming packet does not occur until the packet arrives at a node.

![Graph](attachment:image.png)

(a) CCE vs. $m$ when $\beta = 0.01$.

![Graph](attachment:image.png)

(b) CCE vs. $m$ when $\beta = 0.02$.

![Graph](attachment:image.png)

(c) CCE vs. $m$ when $\beta = 0.04$.

Figure 4. The impact of $m$ on CCE when $\alpha = 1$. 
Figure 5. The impact of $m$ on throughput when $\alpha = 1$. 

(a) Throughput vs. $m$ when $\beta = 0.01$. 

(b) Throughput vs. $m$ when $\beta = 0.02$. 

(c) Throughput vs. $m$ when $\beta = 0.04$. 
Figure 6. The impact of $m$ on delay when $\alpha = 1$. 

(a) Delay vs. $m$ when $\beta = 0.01$.

(b) Delay vs. $m$ when $\beta = 0.02$.

(c) Delay vs. $m$ when $\beta = 0.04$. 
The simulation using the same settings for the parameters showed that the collision probabilities in the ECCE, CSMA-CA and ABA were close. We present the results of the simulation when $m = 10, 12, \ldots, 30 \alpha = 1$ and $\beta = 0.04$ in Figure 9.

In summary, the proposed ECCE could improve throughput and reduce delay by optimizing the three key parameters of the CSMA-CA scheme introduced in the MAC layer of the IEEE 802.15.4 standard.
6. Conclusions

Since the first publication of the IEEE 802.15.4 standard, the standard has been frequently renewed. The CSMA-CA mechanism for the node to contend for the channel, however, has remained almost unchanged. Channel contention efficiency is heavily affected by the three parameters defined in the CSMA-CA mechanism. The proposed ECCE enables a node in an 802.15.4 network to set its optimal values by solving the formulated optimization problem. With the proposed ECCE scheme, the nodes operating with the IEEE 802.15.4 standard can improve throughput and reduce packet delay. In the future, we will consider formulating an analysis model for 802.15.4 networks and setting optimal values for 802.15.4 networks in which a node has different kinds of traffic.

Author Contributions: Conceptualization, Y.-H.Z. and L.J.; methodology, Y.-H.Z.; software, L.J. and Y.Z.; validation, L.J. and Y.Z.; writing—original draft preparation, L.J.; writing—review and editing, Y.-H.Z.; supervision, Y.-H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This project was supported by the National Natural Science Foundation of China under grant numbers 61772470 and 61432015.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The funders had no role in the design of the study.

1. IEEE Computer Society. IEEE Standard for Local and Metropolitan Area Networks-Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs); LAN/MAN Standards Committee of the IEEE Computer Society: New York, NY, USA, September 2006.
2. IEEE Computer Society. IEEE Standard for Local and Metropolitan Area Networks-Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) (Revision of IEEE Std 802.15.4-2006); LAN/MAN Standards Committee of the IEEE Computer Society: New York, NY, USA, September 2011.
3. IEEE Computer Society. IEEE Standard for Low-Rate Wireless Networks (Revision of IEEE Std 802.15.4-2011); LAN/MAN Standards Committee of the IEEE Computer Society: New York, NY, USA, September 2015.
4. IEEE Computer Society. IEEE Standard for Low-Rate Wireless Networks (Revision of IEEE Std 802.15.4-2015); LAN/MAN Standards Committee of the IEEE Computer Society: New York, NY, USA, May 2020.
5. Zhu, Y.-H.; Gong, S.; Chi, K.; Li, Y.; Fang, Y. Optimizing Superframe and Data Buffer to Achieve Maximum Throughput for 802.15.4 Based Energy Harvesting Wireless Sensor Networks. IEEE Internet Things J. 2021, 8, 3689–3704. [CrossRef]
6. Tao, Z.; Panwar, S.; Gu, D.; Zhang, J. Performance analysis and a proposed improvement for the IEEE 802.15.4 contention access period. In Proceedings of the 2006 IEEE Wireless Communications and Networking Conference (WCNC), Las Vegas, NV, USA, 3–6 April 2006; pp. 1811–1818.
7. Abegunde, J.; Xiao, H.; Spring, J. A Dynamic Game with Adaptive Strategies for IEEE 802.15.4 and IoT. In Proceedings of the 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, China, 23–26 August 2016; pp. 473–480.

8. Zhang, Z.; Lu, X. An Advanced Design of MAC Layer for Smart Metering in Internet of Things. In Proceedings of the 2020 International Wireless Communications and Mobile Computing (IWCMC), Limassol, Cyprus, 15–19 June 2020; pp. 165–171.

9. Shu, F.; Sakurai, T. Analysis of an Energy Conserving CSMA-CA. In Proceedings of the IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference, Washington, DC, USA, 26–30 November 2007; pp. 2536–2540.

10. Zhong, Y.; Xu, P.; Bi, G.; Bao, F.S. Analysis of Energy Efficiency and Power Saving in IEEE 802.15.4. In Proceedings of the 2007 IEEE Wireless Communications and Networking Conference, Hong Kong, China, 11–15 March 2007; pp. 3330–3334.

11. Wang, C.; Yin, L.; Oien, G.E. Energy-efficient medium access for wireless sensor networks under slow fading conditions. In Proceedings of the 2009 Sixth International Conference on Broadband Communications, Networks, and Systems, Madrid, Spain, 14–16 September 2009; pp. 1–6.

12. Brienza, S.; Guglielmo, D.D.; Anastasi, G.; Conti, M.; Neri, V. Strategies for optimal MAC parameter setting in IEEE 802.15.4 wireless sensor networks: A performance comparison. In Proceedings of the 2013 IEEE Symposium on Computers and Communications (ISCC), Split, Croatia, 7–10 July 2013; pp. 000898–000903.

13. Ha, J.Y.; Kim, T.H.; Park, H.S.; Choi, S.; Kwon, W.H. An Enhanced CSMA-CA Algorithm for IEEE 802.15.4 LR-WPANs. IEEE Commun. Lett. 2007, 11, 461–463. [CrossRef]

14. Zhang, Y.; Zhou, Y.; Gao, L.; Qian, Y.; Li, J.; Shu, F. A blind adaptive tuning algorithm for reliable and energy-efficient communication in IEEE 802.15.4 networks. IEEE Trans. Veh. Technol. 2017, 66, 8605–8609. [CrossRef]

15. Khandish, B.; Lee, E.; Park, H.; Suk, J. An efficient backoff scheme in wireless sensor networks. In Proceedings of the 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN), Prague, Czech Republic, 3–6 July 2018; pp. 332–337.

16. He, J.; Tang, Z.; Chen, H.; Zhang, Q. An accurate and scalable analytical model for IEEE 802.15.4 slotted CSMA/CA networks. IEEE Trans. Wirel. Commun. 2009, 8, 440–448. [CrossRef]

17. Bouazzi, I.; Bhar, J.; Atri, M. A Queue Enhanced Backoff Algorithm for wireless sensor networks. In Proceedings of the 2017 2nd International Conference on Anti-Cyber Crimes (ICACC), Abha, Saudi Arabia, 26–27 March 2017; pp. 160–165.

18. Boudour, G.; Heusse, M.; Duda, A. Improving performance and fairness in IEEE 802.15.4 networks with capture effect. In Proceedings of the 2013 IEEE International Conference on Communications (ICC), Budapest, Hungary, 9–13 June 2013; pp. 1683–1687.

19. Moulik, S.; Misra, S.; Chakraborty, C. Performance Evaluation and Delay-Power Trade-off Analysis of ZigBee Protocol. IEEE Trans. Mob. Comput. 2019, 18, 404–416. [CrossRef]

20. Khanafar, M.; Guennoun, M.; Mouftah, H.T. An Efficient Adaptive Backoff Algorithm for Wireless Sensor Networks. In Proceedings of the 2011 IEEE Global Telecommunications Conference-GLOBECOM 2011, Houston, TX, USA, 5–9 December 2011; pp. 1–6.

21. Faridi, A.; Palattella, M.R.; Lozano, A.; Dohler, M.; Boggia, G.; Grieco, L.A.; Camarda, P. Comprehensive evaluation of the IEEE 802.15.4 MAC layer performance with retransmissions. IEEE Trans. Veh. Technol. 2010, 59, 3917–3932. [CrossRef]

22. Azdad, N.; Boukhari, M. Performance analysis of the beacon-enabled operation of IEEE 802.15.4 under WBANs. In Proceedings of the 2019 International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco, 29 October–1 November 2019; pp. 1–5.

23. Moulik, S.; Dey, D.; Ray, K.; Das, K. Reliable Transmission of Critical Packets in IEEE 802.15.4-based Body Area Networks. In Proceedings of the 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), Goa, India, 16–19 December 2019; pp. 1–6.

24. Alaparthi, N.; Parvataneni, S.R.; Kanderao, N.V.S.; Jagtap, S.V.; Donghu, A.; Korivi, H. Performance comparison of CSMA, MACA, Generic MAC and Sensor MAC channel access protocols for ZigBee WSN with RIPv2 as Routing protocol. In Proceedings of the 2019 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, 22–23 March 2019; pp. 1–5.

25. Nabila, A.; Mohamed, E. An enhanced backoff strategy for fair channel access in WBAN-based health monitoring systems. In Proceedings of the 2020 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, 9–11 June 2020; pp. 1–4.

26. Singh, K.; Prabhakar, T.V.; Kuri, J. Improving Access to Network Resources in IoT Networks. In Proceedings of the 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS), Bengaluru, India, 7–11 January 2020; pp. 259–266.

27. Aboubakar, M.; Roux, P.; Kellil, M.; Bouabdallah, A. An Efficient and Adaptive Configuration of IEEE 802.15.4 MAC for Communication Delay Optimisation. In Proceedings of the 2020 11th International Conference on Network of the Future (NoF), Bordeaux, France, 12–14 October 2020; pp. 1–7.

28. Musaddiq, A.; Rahim, T.; Kim, D.-S. Enhancing IEEE 802.15.4 Access Mechanism with Machine Learning. In Proceedings of the 2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN), Jeju Island, Korea, 17–20 August 2021; pp. 210–212.

29. Alimorad, N.; Maadani, M.; Mahdavi, M. REO: A Reliable and Energy Efficient Optimization Algorithm for Beacon-Enabled 802.15.4-Based Wireless Body Area Networks. IEEE Sens. J. 2021, 21, 19623–19630. [CrossRef]