future internet

MDPI

Article

Decoupling-Based Channel Access Mechanism for Improving Throughput and Fairness in Dense Multi-Rate WLANs

Jianjun Lei *, Ying Wang and Hong Yun

School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; S170201006@stu.cqupt.edu.cn (Y.W.); yunhong@vivo.com (H.Y.)
* Correspondence: leijj@cqupt.edu.cn; Tel.: +86-139-9635-3443

Received: 12 November 2019; Accepted: 20 December 2019; Published: 23 December 2019

Abstract: Legacy IEEE 802.11 Medium Access Control (MAC) adopts the Distributed Coordination Function (DCF) mechanism, which provides the same access opportunity for all contenders. However, in dense multi-rate Wireless Local Area Networks (WLANs), the pure distributed control mechanism will cause high collision rate and performance anomaly, which results in low network utilization and wasting valuable channel resources. In this paper, we present a decoupling MAC mechanism (DMAC) based on the idea of contention/reservation to reduce collision and realize collision free data transmission. In proposed mechanism, the channel access time is partitioned into channel contention process and data transmission process. The proposed algorithm makes full use of the distributed random channel access mechanism and performs a centralized collision-free data transmission. Wherein, we also design an adaptive algorithm to adjust the length of the contention period to improve the channel utilization. Furthermore, we further propose two airtime fairness algorithms Improve-DMAC1 (I-DMAC1) and Improve-DMAC2 (I-DMAC2) for delay sensitive network and high throughput network scenarios, respectively, to solve the performance anomaly in multi-rate WLANs, based on DMAC. We verify the effectiveness of these decoupling algorithms through extensive simulations. Moreover, the simulation results show that the proposed algorithms achieve better performance than the 802.11 standard and other protocols.

Keywords: WLANs; performance anomaly; decoupling MAC; airtime fairness

1. Introduction

Nowadays, increasing wireless nodes utilize the IEEE 802.11 mechanism to regulate the wireless channel utilization [1]. The wireless nodes often adjust dynamically data transmission rates according to the signal strength and interference in the environment to achieve better transmission performance. Heusse et al. [2] shows that performance anomaly occurs when different transmission rates simultaneously coexist in the network inevitably, the DCF mechanism in legacy IEEE 802.11 provides an equal long-term transmission opportunity for each node. Thus, if each node has the same bitrate and uses the same frame size, the throughput fairness can be achieved. Multiple bitrates are defined in the legacy 802.11 standard. However, the throughput fairness might sometimes severely degrade the network performance in multi-rate Wireless Local Area Networks (WLANs). The fundamental reason for the degradation is that the lower rate nodes occupy more channel time to transmit data frame, rather than the higher rate nodes. Many algorithms have been proposed to target different types of fairness, such as throughput fairness, time fairness, max-min fairness, and proportional fairness, to remedy this problem. However, in multi-rate WLANs, the ideal channel allocation fairness mechanism is that the channel occupation time is equal to all stations. It confirms that achieving time fairness among stations can tackle the performance anomaly, maximize the system throughput, and improve system performance in [3].

Besides, another serious problem is that as the network density increases; DCF also causes more collisions and heavily deteriorates the network performance. The legacy DCF adopts the binary exponential back-off (BEB) mechanism, when the node has data to send, it randomly selects a backoff counter from zero to contention window (CW). When the backoff counter reaches zero, the node can begin its transmission. However, as the nodes density increases, the probability of nodes choosing the same backoff counter is greater, which caused more collisions and degraded the overall performance. Fundamentally, the bigger CW can result in the wasting of a lot of idle time and the smaller CW can lead to more data collision in the heavy traffic scenario. Therefore, there is a pervasive viewpoint that channel access mechanism should be designed more sophisticated to overcome the highly challenging in dense network scenarios. Furthermore, various mechanisms for tuning the CW sizes were proposed in [4–9] to mitigate collision and improve throughput. The issue that the network capacity will be severely underutilized when the legacy DCF operates as core part of WLANs has already been recognized.

Recently, the standardization of IEEE 802.11ax [10] has been initiated and one of its prime goals is to increase the capacity of Medium Access Control (MAC) in dense WLANs. According to [6] and [11], the orthogonal frequency division multiple access (OFDMA) technology has been introduced into 802.11ax to improve the MAC layer. In 802.11ax, a full channel is divided into multiple orthogonal sub-channels based on OFDMA technology, so as to support multi-user parallel access, and the AP is in charge of scheduling both downlink (DL) and uplink (UL) transmissions. In addition, in the UL transmission phase, a hybrid channel allocation mechanism, including random access and scheduled access, is also proposed [12].

In this paper, from another perspective, we focus on how to avoid the intensive channel contention and solve the performance anomaly in dense multi-rate WLANs. We introduce a decoupling MAC mechanism, namely decoupling MAC mechanism (DMAC), to achieve these goals. Motivated by the idea of hybrid channel mechanism in 802.11ax, DMAC employs a hybrid channel allocation mechanism that integrates a distributed channel contention processing and centralized data transmission scheduling. In DMAC, the channel time is partitioned into the contention process (CP) and data transmission process (TP) by a decoupling channel access mechanism. During contention time period, all of the nodes attempt to transmit a Data Transmission Request (DTR) frame to compete for the channel. The centralized Access Point (AP) receives the DTR frame from the node and inserts it into a potential sequence queue without any transmitting requirement of acknowledgement (ACK) frame in the meantime. In TP, the AP successively schedules all nodes in the queue to send the data consecutively by an (acknowledgement) ACK frame piggybacking approach and, thus, no collision occurs during this period. DMAC also is designed to adjust dynamically the length of CP by an adaptive algorithm in order to further improve the channel utilization. This decoupling channel access mechanism has been demonstrated that it can significantly improve the channel utilization in our previous work [13]. Subsequently, in this paper, we also inherit the merit of the decoupling channel access and simultaneously address the problem of performance anomaly in the dense multi-rate network. Furthermore, we propose two airtime fairness algorithms for different application scenarios, referred to as I-DMAC1 and I-DMAC2 algorithm, respectively. I-DMAC1 provides multiple contention opportunities for high-rate nodes during the contention period, and I-DMAC2 enables the nodes to transmit multiple packets according to their data rates during TP.

The main contributions of this paper are summarized, as follows:

- Firstly, an efficient hybrid channel access mechanism DMAC for dense multi-rate network is proposed. The proposed mechanism can achieve collision-free data transmission by decoupling channel contention and data transmission. Additionally, an adaptive algorithm is designed in DMAC to dynamically adjust the length of CP and further improve the channel utilization.

- Secondly, we also propose two airtime fairness algorithms I-DMAC1 and I-DMAC2 based on DMAC, and tackle the performance anomaly for different application scenarios in multi-rate network, such as delay-sensitive and high-throughput networks, respectively.
- Thirdly, we establish a mathematical analysis model for DMAC. It is theoretically proven that the system throughput can be improved by decoupling the data transmission and contention process.
- Lastly, we extensively evaluate the performance of the proposed algorithms in saturated and non-saturated network scenarios, and the simulation results verify that I-DMAC1 algorithm is more suitable to the delay-sensitive data transmission scenarios and I-DMAC2 algorithm can perform the better performance in some high throughput demand scenarios.

The remainder of the paper is organized, as follows. Section 2 introduces some literature related to this paper. Section 3 presents the DMAC and I-DMAC algorithm. In Section 4, we validate the performance of this decoupling channel access algorithm by numerical analysis. In Section 5, we present the simulation results. Finally, Section 6 concludes this paper.

2. Related Work

Recent advances in dense WLANs push the researchers to solve the channel collision problem and performance anomaly in order to provide good users' experience. For the collision problem, the most literature attempts to find an optimal CW. In [14], Bianchi uses a two-dimensional Markov chain model to analyze the performance of DCF in a saturated network (each node always has data to send). He demonstrates that the selection of CW values is a key factor affecting the system throughput and derives an optimal CW value that should be related to the nodes' number in the network. Therefore, the solutions that were discussed in many literatures are to dynamically adjust the CW value. In [4], the authors obtain an optimal CW value by estimating the number of nodes according to a collision probability mechanism. In [5], the authors propose an adaptive practical channel observation-based scaled backoff (COSB) to scale-up and scale-down the CW according to the channel observation-based conditional collision probability, which achieves higher throughout and shorter delay. Subsequently, they further propose a cognitive backoff (CB) mechanism that adaptively determines the CW to provide efficient collision avoidance with high throughput and low delay under both dense and sparse conditions in [6]. In [7], the authors obtain the number of nodes by calculating the number of idle slots. In [8], the authors achieve the maximum system throughput by adaptively adjusting the CW according to the network condition. In [9], the authors develop a mathematical model for obtaining the relationship between the CW value and the transmission delay between any two nodes, which can maximize the system throughput by an iterative algorithm. Although the above algorithms can be applied to a dynamic network environment, they need to be simultaneously executed on nodes and AP and result in extra operating cost. Therefore, [15] and [16] present two algorithms that do not require any hardware modifications and only run on nodes. In [15], all of the nodes decrease the back-off counter with a certain probability that can be calculated by some available network parameters during the idle time. In [16], the AP can calculate and assign a back-off value for nodes by running a VBA algorithm in the back-off stage after nodes successfully transmit. Alternatively, some literature attempts to tackle the transmission collision by a contention/reservation algorithm. In [17], the authors propose a Token-DCF protocol, in which each node schedules its neighbor that has the longest queue with a certain probability by its MAC header. The Token-DCF can result in some nodes' starvation when one of its neighbors always has the longest queue. In [18] and [19], the authors propose an Implicit Ordering-MAC (IO-MAC) mechanism for degrading the collision. In [18], the successful transmission node is considered as the predecessor of its following node. The successive node will take part in contenting with a random back-off value in a range from CWmin to 2CWmin-1 when its predecessor fails to transmit or has no more data to transmit. However, the successive node can use a smaller back-off value than the back-off value of its predecessor when several nodes simultaneously follow the IO-rule, which will result in the invalidation of this algorithm. Hence, in [19], the time slot is divided into a reservation period and a contention period. The nodes select its successful transmission order in the current cycle as its back-off value while they enter successfully into the reservation period. However, it also can be inefficient due to lots of idle contention time slots being wasted, while all nodes get into a contention-free status. In [20], a mechanism carrier sense multiple access/contention queuing (CSMA/CQ) based on software defined network (SDN) is proposed to solve the collision problem, but it needs multiple channels due to its separation of operation in the contention and transmission channel.

Aiming to the performance anomaly problem, there have been some significant efforts to improve the system throughput and channel fairness. In [21], the authors verify that modifying the CWmin of station is inversely proportional to the transmission rate that can approximately provide the effectiveness for airtime fairness. Therefore, setting the CWmin of nodes based on their data-rate can achieve the airtime fairness. In [22], the authors propose a distributed average-CW algorithm, which implements time fairness by adjusting the CW of each node in real-time. In [23], the authors present an Idle-Sense algorithm to achieve an approximate time fairness, which dynamically adjusts CW according to the number of estimated idle slot and nodes' rates. In [3], the authors propose an objective function that maximizes system throughput under time fairness condition and, thus, obtains an optimal CW value for each node, which results in high computational cost when some nodes frequently join or leave the network. In [24], the authors propose an algorithm that runs multiple virtual DCF instances and its number is proportional to the node's data-rate. Therefore, the high-rate nodes can obtain more channel access opportunities. In [25] and [26], some approaches by adjusting the size of the packet are developed for achieving time fairness. Fundamentally, they let the size of packet be proportional to the node's data rate. However, the throughput of low-rate nodes can be severely restrained due to its size of packet always being suppressed, even though the highest rate node always transmits the packet with the Maximum Transmission Unit (MTU). Recently, a hybrid slotted-CSMA/CA time-division multiple access (TDMA) has been proposed in [27], which focuses on solving the massive registration of IoT devices at a single centralized AP.

However, none of the above works propose a comprehensive solution to simultaneously solve the channel collision and performance anomaly problem, because the optimal CW is sometimes hard to satisfy these two purposes and the nodes often need a complex distributed cooperation. In our initial work [13], we propose a decoupling channel access algorithm to overcome this limitation. Similar to the CSMA/CQ proposed in [20], we partition the channel utilization into contention and transmission process. However, unlike CSMA/CQ, our algorithm only operates in one channel, thus it is more flexible to be deployed in many single channel scenarios. When compared with our previous work in [13], in this paper, we also address the channel airtime fairness in dense multi-rate scenarios and present the numerical analytical results to theoretically verify its effectiveness of this decoupling channel access mechanism.

3. System Framework and Algorithm Procedure

In this section, we present the algorithm procedure, frame format, and some implementation details. Fundamentally, the DMAC algorithm partitions the channel time into a series of superframe periods. Additionally, each superframe consists of a CP and a TP. The nodes successfully contending channel during CP can send the data packet continuously and with collision-free according to the order in contention queue (CQ) during TP. I-DMAC1 is designed to provide different channel contention probability for nodes to achieve the time fairness in multi-rate scenarios, and I-DMAC2 enable the nodes to transmit several packets in one transmission opportunity without collision occurring.

3.1. Overview

In DMAC, we partition the channel utilization into two time slots: CP and TP. During the CP, all of the nodes transmit a DTR frame to the AP. The nodes that send successfully the AP inserts the DTR into a transmission queue, meanwhile the AP does not transmit the ACK frame to the nodes. During the TP, these nodes will be sequentially scheduled to transmit data by a beacon or ACK from the AP.

3.2. Frame Format

DMAC separates the channel time into a series of super-frames, which consists of one CP, one beacon, and one TP slots. Figure 1 gives the format of superframe, DTR frame, and ACK frame. Figure 1a depicts the superframe format. During the CP period, all of the nodes may transmit a DTR frame to the central node AP. The Figure 1b depicts the format of DTR, which includes the Transmission node's MAC Address (TA) and Receiving node's Address (RA). After CP, the AP will activate the data transmission period by broadcasting a beacon. The first node in the sequence queue of the AP will be paged to send a frame, and then the subsequent node also will be successively activated to send a frame by an ACK frame from AP. Finally, all of the potential transmission nodes in the queue can be scheduled to send a frame without any collision occurring. The central node AP transmits periodically the beacon frame, which activates the data transmission procedure and transmits some control information.

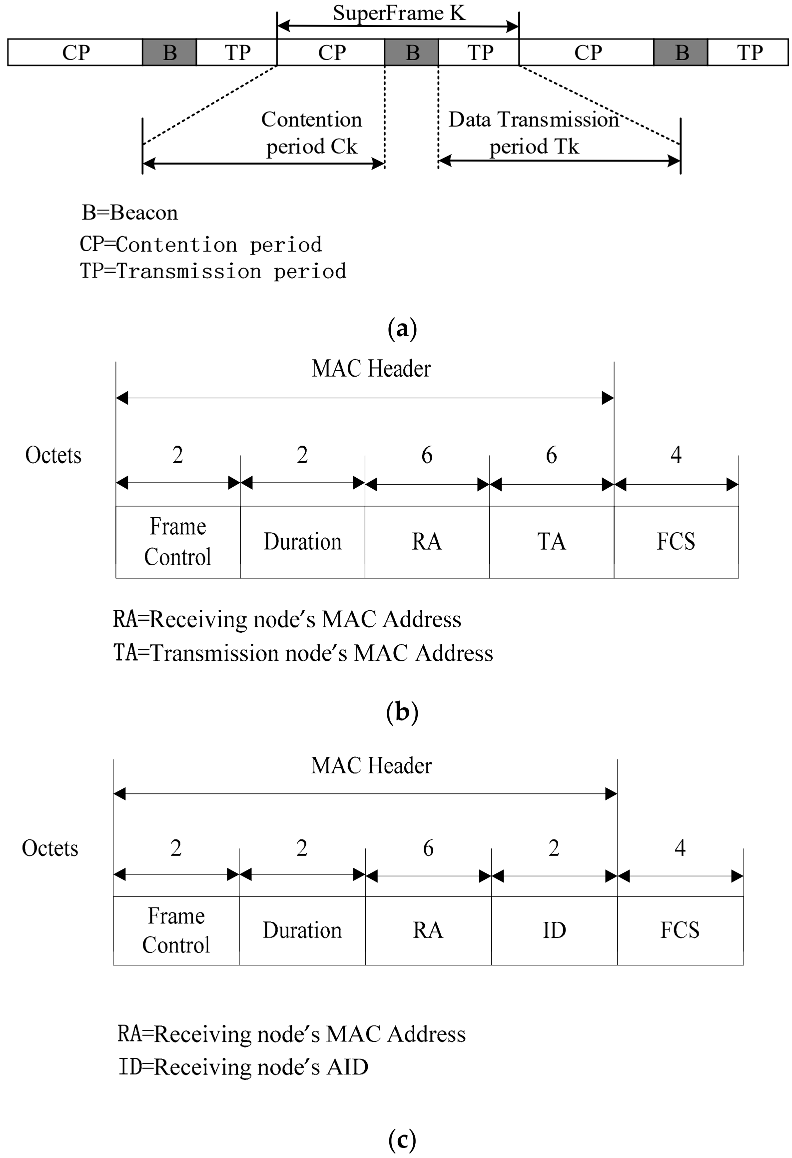

Figure 1. The format of super-frame, Data Transmission Request (DTR) and acknowledgement (ACK) frame. (**a**) superframe; (**b**) DTR frame; (**c**) ACK frame.

3.3. Basic DMAC Procedure

In this section, we detail the procedures of basic DMAC algorithm. We assume that four nodes (denoted as Nodes 1, 2, 3 and 4) connect to the AP and attempt to contend the channel to facilitate the demonstration of DMAC. The basic DMAC includes two procedures: contention processing and data transmission processing. Figure 2 demonstrates the workflow for two phases.

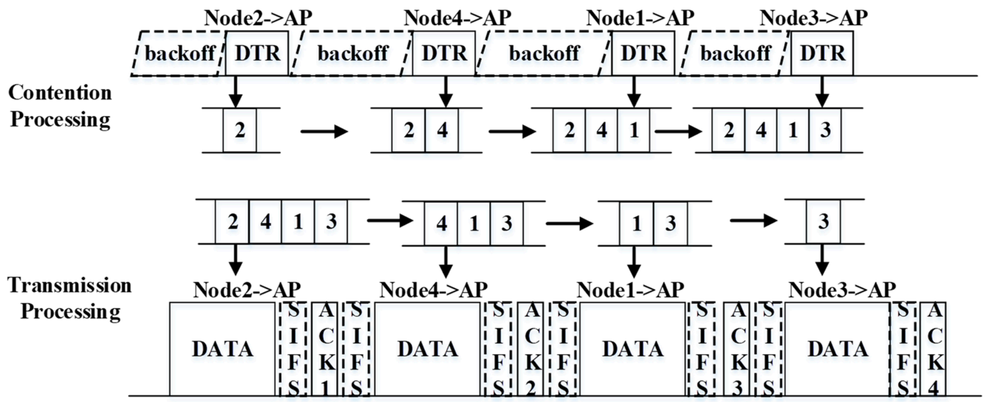

Figure 2. The procedure of basic decoupling Medium Access Control mechanism (DMAC).

In the contention phase, each node selects a random value within a minimum CW as its back-off counter and utilizes CSMA/CA mechanism to compete for the channel. Different from legacy IEEE 802.11 CSMA/CA mechanism, all of the nodes get the same CW by the beacon from the AP and this value is determined according to the number of node in the network. Meanwhile, the node immediately sends a DTR when its counter decreases to zero and the channel is idle. Otherwise, the node will give up this opportunity in the current turn if the collision incurs. Finally, the AP broadcasts a beacon frame to indicate the termination of the contention period. During this phase, the AP follows a rule of First Come First Serve (FCFS) and it puts all nodes that successfully transmit a DTR into CQ. Factually, all of the nodes can be recorded in the queue through their association IDs that are assigned by the association response management frame from the AP during association phase.

In the data transmission phase, the AP picks up first node in the contention queue and it broadcasts a beacon to initiate a data transmission process. The beacon triggers all of the nodes to perform a matching algorithm. Subsequently, the matching node utilizes the subsequent time slot to transmit a data frame. After successfully receiving the data frame, the AP will acknowledge it and piggybacks the ID of the subsequent node into this acknowledgement frame. The subsequent nodes in the CQ can overhead this ACK frame, thus also obtaining a transmission opportunity. Successively, all of the nodes in the CQ will be consecutively scheduled to send a data frame without any collision incurring during data transmission phase. Similar to DCF, A Short Inter-Frame Space (SIFS) is inserted into any two data transmission interval after a successful transmission. Thus, a bigger inter-frame space, such as Distributed Inter-Frame Space (DIFS), automatically triggers a new contention process.

3.4. Improved DMAC (I-DMAC) Algorithm

DMAC decouples the channel usage as data transmission and contention process, which can significantly improve channel utilization. However, in multi-rate network, the channel is often over-occupied by some low-rate nodes and resulted in the performance anomaly problem. Thus, in this section, we also present two-time fairness algorithms: I-DMAC1 and I-DMAC2. I-DMAC1 solves the performance anomaly problem by giving more contention opportunity to the high-rate nodes during contention processing, which is demonstrated by the following pseudo code.

Algorithm 1 Contention Processing for I-DMAC1

Input: The number of nodes
Output: The transmission order stored in Contention Queue Buffer
Initial: AP broadcasts Beacon
1: If the node receives a Beacon frame
2: -Initializes the Contention Window value
3: Data Transmission Processing
4: If idle_time == DIFS and the node's back-off_value != 0
5: -Performs the back-off process
6: If back-off_value == 0 and the channel is idle
7: -Transmits a DTR
8: -Increment the competition counter (NumComp++)
9: If NumComp <= R/R_{min} /* does not reach the maximum number of competitions*/
10: -Chooses a random back-off value
11: -Go to step 4

In Algorithm 1, we assume that the date rate of the lowest rate node is R_{min}. Subsequently, within a CP, a node can send several DTR frames by triggering its back-off counter and its value depends on the ratio of its data rate to R_{min}. Therefore, a high-rate node can obtain more transmission opportunities in a super-frame time. Contrarily, I-DMAC2 directly provides more transmission opportunity to the high-rate nodes during the data transmission period. Thus, it can also solve the performance anomaly problem. Accordingly, we present the partial pseudo code of Algorithm 2.

Algorithm 2 Data Transmission Processing for I-DMAC2

Input: A Beacon piggybacking the first ID of Contention Queue Buffer, R, R_{min}
Output: The node in Contention Queue Buffer transmits data
Initial: AP broadcasts Beacon
1: If a node receives a Beacon frame and its ID is piggybacked in the beacon
2: -Transmits a data frame
3: -AP transmits ACK frame
4: If a node receives an ACK frame and its ID is piggybacked in this ACK frame
5: -Transmits a data frame
6: -Increment the transmission counter (NumTran ++)
7: If NumTran <= R/R_{min} /* does not reach the maximum number of transmissions*/
8: -Go to step 5

3.5. Contention Period Optimization

It is worth noting that more optimal throughput can be obtained by tuning the length of the CP. Therefore, we also design a sophisticated mechanism to adaptively adjust it. In this section, we first borrow a mathematical analytical model derived in [2] and numerically analyze the theoretical optimal solution about the contention period. Let τ be the probability that a node sends a packet in a random discrete time slot and then τ can be obtained, as follows:

$$\tau = \frac{2}{1+W} \tag{1}$$

where W is the length of the CP. When n nodes attempt to compete the channel, let P_{tr} to denote the probability that at least one transmission occurs in a time slot. Let P_s denote the probability that a successful transmission occurs on the channel. Therefore, P_{tr} and P_s can be calculated, as follows:

$$p_{tr} = 1 - (1-\tau)^n \tag{2}$$

$$p_s = \frac{n\tau(1-\tau)^{n-1}}{p_{tr}} = \frac{n\tau(1-\tau)^{n-1}}{1-(1-\tau)^n} \tag{3}$$

In addition, in the CP, the system throughput can be calculated as follows:

$$S = \frac{P_s P_{tr} E[P]}{(1-P_{tr})\delta + P_s P_{tr} T_{succ} + P_{tr}(1-P_s) T_{fail}} \tag{4}$$

where $E[P]$ is the average packet payload, which can be substituted by the length of DTR, since only DTR packet is transmitted during the CP. The σ is the duration of an empty slot, T_{succ} and T_{fail} are the average time of successful transmission and collision, respectively, and they are given by:

$$\begin{cases} T_{succ} = T_{DIFS} + T_{DTR} \\ T_{fail} = T_{DIFS} + T_{DTR} \end{cases} \tag{5}$$

where T_{DIFS} and T_{DTR} are the time of DIFS and transmitting a DTR frame, respectively. Therefore, The Equation (4) can be rewritten, as follows:

$$S = \frac{E[P]}{T_{succ} - T_{fail} + \frac{(1-P_{tr})\delta/P_{tr} + T_{fail}}{P_s}} \tag{6}$$

We can obtain the highest throughput by maximizing the following expression, since $E[P]$, σ, T_{succ}, and T_{fail} are fixed for all nodes.

$$\frac{P_s}{(1-P_{tr})\delta/P_{tr} + T_{fail}} = \frac{n\tau(1-\tau)^{n-1}}{T^*_{fail} - (1-\tau)^n \left(T^*_{fail} - 1\right)} \tag{7}$$

where $T^*_{fail} = T_{fail}/\sigma$ is the collision duration measured in unit time slot σ. After some simplifications, it yields the following equation.

$$(1-\tau)^n - T^*_{fail}\left\{n\tau - \left[1-(1-\tau)^n\right]\right\} = 0 \tag{8}$$

Assuming $\tau << 1$, it results in the following.

$$(1-\tau)^n \approx 1 - n\tau + \frac{n(n-1)}{2}\tau^2 \tag{9}$$

Thus, τ can be approximately derived from the following equation.

$$\tau = \frac{\sqrt{\left[n + 2(n-1)\left(T^*_{fail} - 1\right)\right]/n} - 1}{(n-1)\left(T^*_{fail} - 1\right)} \approx \frac{1}{n\sqrt{T^*_{fail}/2}} \tag{10}$$

Finally, we obtain an approximately optimal contention period configuration by substituting the above into Equation (1).

$$W \approx n\sqrt{2T^*_{fail}} \tag{11}$$

4. Performance Analysis

DMAC divides the channel time into a series of repeated slots, and each of them consists of a CP and a TP. We assume a network with n nodes and each node sends saturated data to the channel. We now only consider one transmission cycle to facilitate the evaluation of DMAC. In each CP, the number of nodes that successfully contend for the channel is calculated as:

$$E[n] = np_{tr}p_s \tag{12}$$

In the TP, the time that spends on transmitting a packet is given by:

$$T^* = 2T_{SIFS} + T_{DATA} + T^*_{ACK} \tag{13}$$

where T^*_{ACK} is the transmission time of an ACK frame. T_{SIFS} is time of a SIFS, and T_{DATA} is the time of data transmission. Hence, when combing Equation (1)–(5), (12), and (13), we can obtain the saturated throughput of DMAC, denoted as:

$$S_{DMAC} = \frac{E[n]E[P]}{(1-P_{tr})\delta + P_s P_{tr} T_{succ} + P_{tr}(1-P_s)T_{fail} + T_{Beacon} + E[n]T^*} \tag{14}$$

where $E[P]$ is the average packet payload and T_{Beacon} is the transmission time of beacon in a transmission cycle.

For comparison, we also present the system throughput of DCF mechanism that is derived in [4], denoted as S_{DCF}, is given by

$$S_{DCF} = \frac{P_s P_{tr} E[P]}{(1-P_{tr})\delta + P_s P_{tr} T_s^{DCF} + P_{tr}(1-P_s)T_f^{DCF}} \tag{15}$$

where $E[P]$ is the average packet payload, T_s^{DCF} and T_f^{DCF} are the average time of successful transmission and collision, respectively. With regard to the basic IEEE 802.11 DCF, T_s^{DCF} and T_f^{DCF} are given by:

$$\begin{cases} T_s^{DCF} = T_{DIFS} + T_{DATA} + T_{SIFS} + T_{ACK} \\ T_f^{DCF} = T_{DIFS} + T_{DATA} \end{cases} \tag{16}$$

where T_{ACK} is the transmission time of ACK packet.

Figure 3 demonstrates how the throughput evolves with the transmission probability τ when we set the number of contender as 20. Some parameters, such as σ, T_s^{DCF}, T_f^{DCF}, T_{succ}, T_{fail}, T_{Beacon}, and T^* are considered as invariant constants and Table 1 gives the other parameter configurations.

Table 1. Simulation Parameters.

Parameters	Value	Parameters	Value
Packet_Payload	200,500,1000 bytes	Slot_Time	20 us
MAC_Header	224 bits	DIFS	50 us
PHY_Header	192 bits	SIFS	10 us
Data_Frame_Rate	2 Mbps	ACK	128 bits
Simulation_Time	1800 s	DTR	160 bits

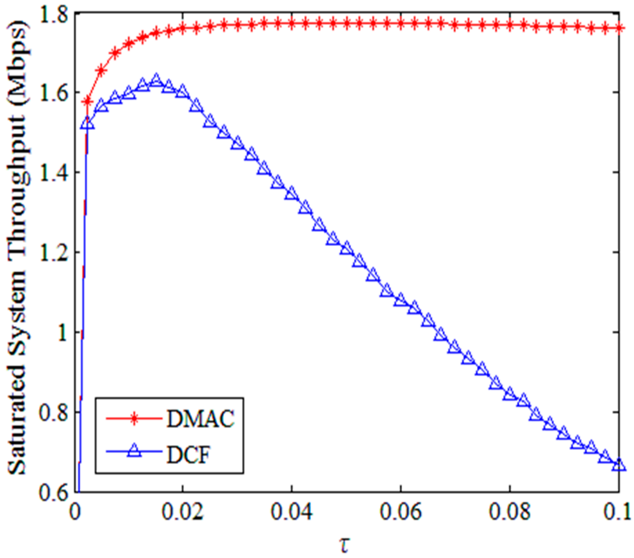

Figure 3. Saturated system throughput versus τ.

It is obvious that the throughput of DMAC always increases and then reaches a certain threshold when $\tau = 0.03$. However, the throughput of DCF gradually decreases when $\tau = 0.0175$. Regardless of τ, the throughput of DMAC is much higher than that of DCF. We also present our simulation results to verify the analytical results in the following section.

5. Simulation Results

This section provides the simulation results for evaluating DMAC and I-DMAC. Table 1 details the parameters for MAC layers. In simulations, the AP can communicate directly with all nodes and they can sense the carrier each other. We investigate the impact over system throughput, channel utilization and delivery delay, and compare them with the legacy 802.11 DCF mechanism and some existing analogous algorithm, such as hybrid MAC protocol with implicit ordering (IO-MAC) 19] and transmission time-based scheme (TTBS) [25].

5.1. DMAC

We consider two different traffic condition, namely saturated and non-saturated. In the saturated scenario, all of the nodes generate packets with an interval time of 1 millisecond (ms) and thus they always have a non-empty transmitting queue. In order to investigate the impact of different packet sizes, we also choose three kinds of packet size: 200, 500, and 1000 bytes. In non-saturated scenario, we connect 50 nodes to one AP and choose a fixed packet size as 1000 bytes. In addition, the packet interval time for nodes is 10 ms and 100 ms, respectively. We also define a variable θ as their proportion and, thus, θ can be regarded as the network saturated degree. Figure 4 shows the system throughput and channel utilization under saturated and non-saturated scenarios.

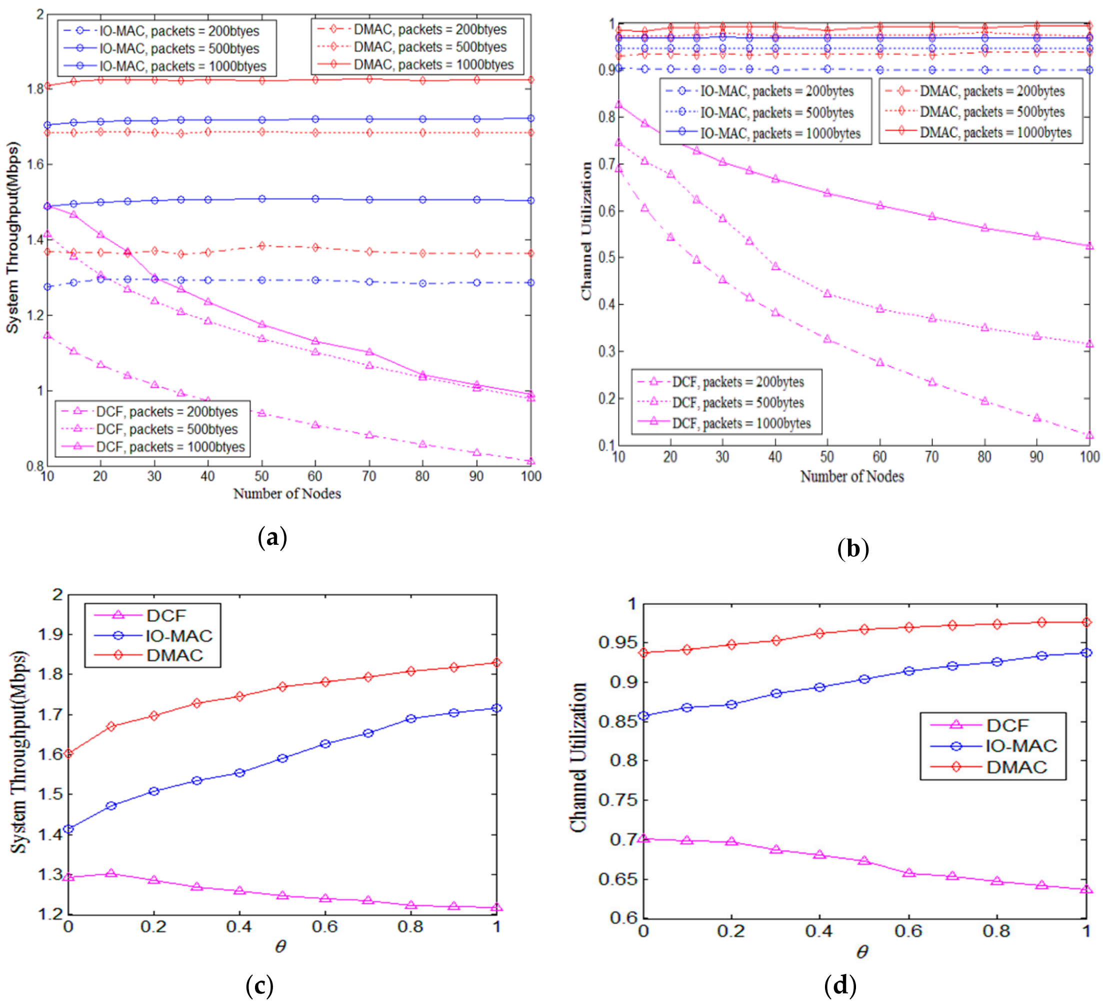

Figure 4. Throughput and utilization in saturated and non-saturated scenarios. (**a**) System Throughput in saturated scenario; (**b**) Channel utilization in saturated scenario; (**c**) System Throughput in non-saturated scenario; and, (**d**) Channel utilization in non-saturated scenario.

Figure 4a,b depict the system throughput and channel utilization varying with different network densities in saturated scenario. When compared to legacy DCF and IO-MAC, the DMAC always keep significant improvement, regardless of the packet size, which is due to its data transmission period that builds a collision-free schedule and performs consecutive transmission. Although the IO-MAC algorithm can also degrade collisions, it still exits much idle time, even though all nodes get into reservation status. As it can be observed in the figure, DMAC has at least 6.8% higher system throughput than IO-MAC. Figure 4c,d show the system throughput and channel utilization in non-saturated scenario. The curves show an increase of the throughput as θ grows. In contrast, throughput and utilization for DCF both keep decreasing due to an augmented number of collisions as the network saturated degree increases. Especially in approximate saturated condition (when $\theta \approx 1$), DCF suffers from an increasing number of collisions. Additionally, nevertheless, DMAC performs longer periods of collision-free operation due to all nodes getting saturated.

We also investigate the delay by letting all nodes to delivery one Mega size file to the AP. In the saturated scenario, different packet size and network density are considered, similar to the previous simulations. Additionally, in the non-saturated scenario, we connect 50 nodes to one AP and choose a fixed packet size with 1000 bytes too. Subsequently, the maximum, minimum, and average delivery time for all nodes can be obtained under different saturated degree and Figure 5 shows their curves.

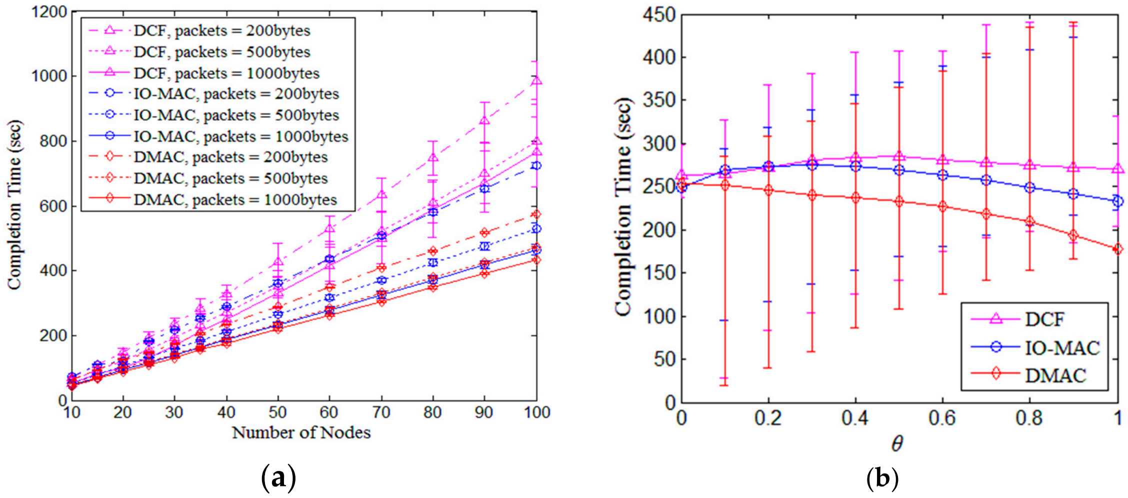

Figure 5. Delay in saturated and non-saturated scenarios. (**a**) Completion time in saturated scenario; (**b**) Completion time in non-saturated scenario.

Figure 5a,b show the delivery time in saturated and non-saturated scenarios, respectively. We can see that DMAC can provide almost the same file delivery delay for all nodes and it has the lowest delay under different packet size and network density, which is due to its high channel utilization by eliminating collisions. In the non-saturated scenario, the saturated degree has a strong impact on all algorithms and the delivery time fluctuates greatly for different nodes. This is because the channel is often not efficiently utilized when some nodes compete for it. Nevertheless, DMAC always has the lowest average delivery time than the other two algorithms.

5.2. I-DMAC

I-DMAC springs from a modification to DMAC's channel decoupling access mechanism and emphatically addresses the airtime fairness in the multi-rates network scenarios. Therefore, we also present the evaluating results of I-DMAC in terms of throughput, utilization, delay, and fairness in this section. We utilize four different data rate nodes (1, 2, 5.5, and 11 Mbps) that join the network as one group; therefore, one group increases the network density once. All of the nodes are assumed to be saturated and their packet sizes are 1000 bytes. In simulations, we use the Jain's Fairness Index (JFI) [28] as the metric of the channel fairness and compare I-DMAC with TTBS, since they both concentrate on fairness.

As indicated in Figure 6a,b, I-DMAC1 and I-DMAC2 always keep stable in terms of throughput and utilization and they have almost the same channel utilization. From the perspective of throughput, I-DMAC2 can obtain more significant throughput improvement, which, due to that, I-DMAC2 can continuously transmit several packets in the data transmission period once it competes for the channel, thus reducing the collision rather than the other algorithms. On the contrary, the TTBS only changes the size of the packet and enables the high-rate nodes to transmit larger packets once. Definitely, it cannot relieve the collision, especially in the dense scenario, which leads to the fact that the throughput decline steadily as the number of nodes increases.

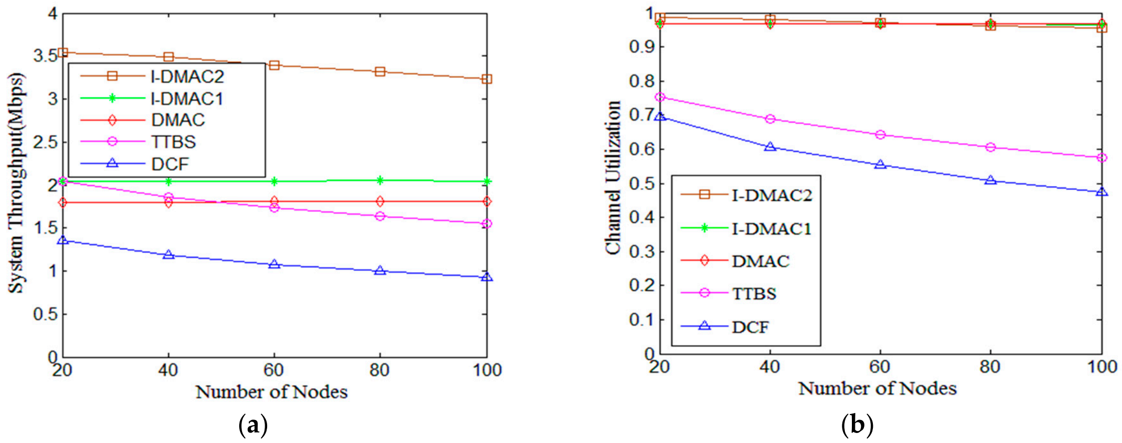

Figure 6. Throughput utilization in multi-rate scenario. (**a**) System throughput; and, (**b**) Channel utilization.

Analogously, we also investigate the transmission time of these algorithms for all nodes transmitting one mega size file to AP. It can be observed from Figure 7 that the I-DMAC1 significantly outperforms the other algorithms due to the fair channel utilization and lower collision occurrence. The completion time of TTBS is the highest and it obviously fluctuates. The reason is that the high-rate nodes occupy excessively the airtime and some low-rate nodes can always be in the state of "starvation". In the dense scenario (100 nodes), I-DMAC2 has the similar situation with the TTBS. This is also because high-rate nodes occupy too much transmission opportunities in the data transmission period.

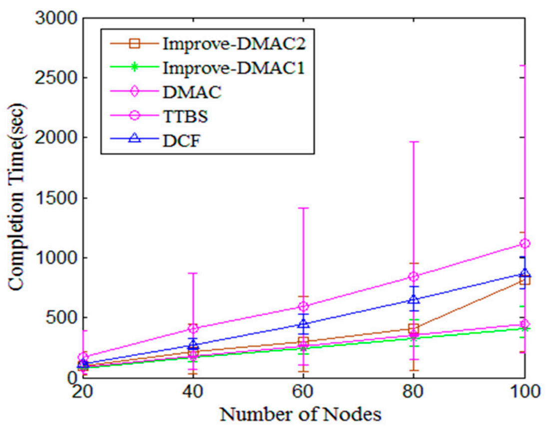

Figure 7. Transmission time under multi-rate scenario.

Finally, we also investigate the airtime fairness for all contenders by letting them operate for 3600 seconds. Except for TTBS, the packet size for all algorithms is 1000 bytes. The JFI curves in Figure 8 show JFI for all tested protocols. It can be observed that the fairness index for all algorithms does not change significantly as the network density increases. Remarkably, the JFI of TTBS is close to 1, which reflects that it obtains the best airtime fairness. The JFI of I-DMAC2 and I-DMAC1 also exceed 0.9 and 0.8, respectively. However, DMAC and DCF perform worse performance due to their no consideration for multi-rate environment.

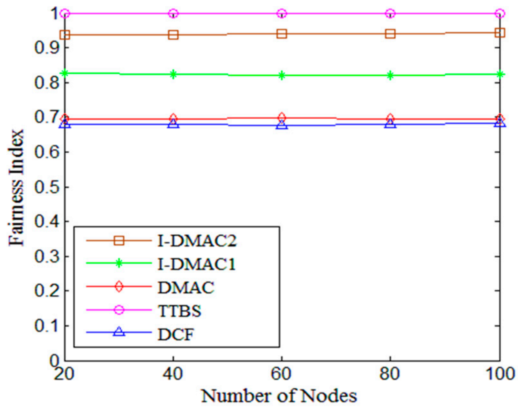

Figure 8. Fairness index under multi-rate scenario.

In summary, I-DMAC1 and I-DMAC2 are not as good as TTBS in term of airtime fairness. However, they obtain better throughput, utilization, and delay. In addition, except for delay, I-DMAC2 always performs better performance than I-DMAC1. Therefore, I-DMAC1 is suitable for some delay-sensitive applications and I-DMAC2 can provide the better system throughput and airtime fairness for the other applications.

6. Conclusions

Legacy DCF coordinates the channel access among multiple nodes by employing a simple distributed control mechanism. However, with various multimedia applications being deployed, this pure distributed access control becomes hard to meet the requirements of these applications and it hinders the further improvement of throughput. In this paper, we present a decoupling DMAC algorithm by partitioning the channel access into contention period and data transmission period, which inherits the merit of distributed random channel access and enables nodes to perform a collision-free data transmission. We present the algorithm procedure and framing details and validate the effectiveness of this channel decoupling mechanism by numerical analysis. In addition, we also address the performance anomaly problem and propose the I-DMAC1 and I-DMAC2 algorithms applied to some multi-rate scenarios. By running some extensive simulations, these proposed algorithms are verified to significantly improve the system throughput and channel utilization. Inspired by these latest machine learning-based MAC algorithms in [29,30], for our future work, we plan to take advantage of these machine learning-based mechanisms to optimize our proposed mechanism and integrate our proposed algorithm in 802.11ax.

Author Contributions: Conceptualization, J.L.; data curation, Y.W.; formal analysis, J.L. and Y.W.; funding acquisition, J.L.; investigation, H.Y.; methodology, J.L. and H.Y.; project administration, J.L.; resources, Y.W.; software, H.Y.; supervision, J.L.; validation, H.Y.; visualization, Y.W.; writing, original draft, H.Y.; writing, review and editing, J.L., Y.W., and H.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Science Foundation of China (61602073) and Chongqing Postgraduate Research and Innovation Project (CYS19260).

Acknowledgments: We would like to appreciate the anonymous CSOC2019 reviewers for their constructive comments on an earlier version of this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. IEEE. *Standard 802.11-2012-IEEE Standard for Information Technology–Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks–Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*; IEEE: Piscataway, NJ, USA, 2012; pp. 1–425.

2. Heusse, M.; Rousseau, F.; Berger-Sabbatel, G.; Duda, A. Performance anomaly of 802.11 b. In Proceedings of the IEEE INFOCOM, San Francisco, CA, USA, 30 March–3 April 2003; pp. 836–843.
3. Le, Y.; Ma, L.; Cheng, W.; Cheng, X.Z. A Time Fairness-Based MAC Algorithm for Throughput Maximization in 802.11 Networks. *IEEE Trans. Comput.* **2015**, *64*, 19–31. [CrossRef]
4. Morino, Y.; Hiraguri, T.; Yoshino, H. A Novel Contention Window Control Scheme Based on a Markov Chain Model in Dense WLAN Environment. In Proceedings of the 3rd International Conference on Artificial Intelligence, Modelling and Simulation, Kota Kinabalu, Malaysia, 2–4 October 2015; pp. 417–421.
5. Ali, R.; Shahin, N.; Kim, S.W. Channel observation-based scaled backoff mechanism for high-efficiency WLANs. *Electron. Lett.* **2018**, *54*, 663–665. [CrossRef]
6. Shahin, N.; Ali, R.; Kim, S.W.; Kim, Y.T. Cognitive backoff mechanism for IEEE802.11ax high-efficiency WLANs. *J. Commun. Netw.* **2019**, *21*, 158–167. [CrossRef]
7. Yu, Q.; Zhuang, Y.; Ma, L. Dynamic contention window adjustment scheme for improving throughput and fairness in IEEE 802.11 wireless LANs. In Proceedings of the Global Communications Conference (GLOBECOM 2012), Anaheim, CA, USA, 3–7 December 2012; pp. 5074–5080.
8. Hong, K.; Lee, S.K.; Kim, K. Channel Condition Based Contention Window Adaptation in IEEE 802.11 WLANs. *IEEE Trans. Commun.* **2012**, *60*, 469–478. [CrossRef]
9. Le, Y.; Ma, L.; Cheng, W.; Cheng, X.Z.; Chen, B. Maximizing throughput when achieving time fairness in multi-rate wireless LANs. In Proceedings of the IEEE INFOCOM, Orlando, FL, USA, 23–30 March 2012; pp. 2911–2915.
10. Bellalta, B. IEEE 802.11ax: High-efficiency WLANS. *IEEE Wirel. Communs.* **2016**, *23*, 38–46. [CrossRef]
11. Ali, R.; Shahin, N.; Bajracharya, R.; Kim, B.S.; Kim, S.W. A Self-Scrutinized Backoff Mechanism for IEEE 802.11ax in 5G Unlicensed Networks. *Sustainability* **2018**, *10*, 1201. [CrossRef]
12. Ali, R.; Kim, S.W.; Kim, B.S. Design of MAC Layer Resource Allocation Schemes for IEEE 802.11ax: Future Directions. *IETE Tech. Rev.* **2018**, *35*, 28–52. [CrossRef]
13. Lei, J.J.; Yun, H. Decoupling Channel Contention and Data Transmission in Dense Wireless Infrastructure Network. In Proceedings of the 8th Computer Science On-line Conference, Hlavni Mesto Praha, Czech Republic, 24–27 April 2019.
14. Bianchi, G. Performance utanalysis of the IEEE 802.11 distribed coordination function. *IEEE J. Sel. Areas Commun.* **2000**, *18*, 535–547. [CrossRef]
15. Karaca, M.; Bastani, S.; Landfeldt, B. Modifying Backoff Freezing Mechanism to Optimize Dense IEEE 802.11 Networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 9470–9482. [CrossRef]
16. Kim, J.D.; Laurenson, D.I.; Thompson, J.S. Centralized Random Backoff for Collision Resolution in Wi-Fi Networks. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 5838–5852. [CrossRef]
17. Hosseinabadi, G.; Vaidya, N. Token-DCF: An Opportunistic MAC protocol for Wireless Networks. In Proceedings of the Fifth International Conference on Communication Systems and Networks (COMSNETS 2013), Bangalore, India, 7–10 January 2013; pp. 1–9.
18. Ko, H.; Lee, G.; Kim, C. IO-MAC: An Enhancement of IEEE 802.11 DCF Using Implicit Ordering. *Wirel. Pers. Commun.* **2013**, *2*, 1467–1473. [CrossRef]
19. Feng, B.; Hong, S.; Wang, Z. IO-MAC: A novel hybrid MAC protocol with implicit ordering in WLANs. In Proceedings of the Sixth International Conference on Wireless Communications and Signal Processing (WCSP 2014), Hefei, China, 23–25 October 2014.
20. Zhao, Q.; Xu, F.; Yang, J. CSMA/CQ: A Novel SDN-Based Design to Enable Concurrent Execution of Channel Contention and Data Transmission in IEEE 802.11 Networks. *IEEE Access* **2017**, *5*, 2534–2549. [CrossRef]
21. Lin, P.; Chou, W.; Lin, T. Achieving airtime fairness of delay-sensitive applications in multirate IEEE 802.11 wireless LANs. *IEEE Commun. Mag.* **2011**, *49*, 169–175. [CrossRef]
22. Cali, F.; Conti, M.; Gregori, E. Dynamic Tuning of the IEEE 802.11 Protocol to Achieve A Theoretical Throughput Limit. *IEEE ACM Trans. Netw.* **2000**, *6*, 785–799. [CrossRef]
23. Heusse, M.; Rousseau, F.; Guillier, R. Idle sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs. In Proceedings of the ACM SIGCOMM Computer Communication Review, Philadelphia, PA, USA, 22–26 August 2005; pp. 121–132.
24. Yazici, M.A.; Akar, N. Running Multiple Instances of the Distributed Coordination Function for Air-Time Fairness in Multi-Rate WLANs. *IEEE Commun. Mag.* **2013**, *12*, 5067–5076. [CrossRef]

25. Alhaji, I.A.; Tijjani, B.I.; Ali, M.H. Determination of Performance Enhancement in IEEE 802.11 multi-rate WLAN. *Int. J. Digit. Inf. Wirel. Commun.* **2015**, *5*, 257–269.
26. Hu, S.; Li, J.; Pan, G. Performance and Fairness Enhancement in IEEE 802.11 WLAN networks. *AEU-Int. J. Electron. Commun.* **2014**, *68*, 667–675. [CrossRef]
27. Shahin, N.; Ali, R.; Kim, Y.T. Hybrid slotted-csma/ca-tdma for efficient massive registration of iot devices. *IEEE Access* **2018**, *6*, 18366–18382. [CrossRef]
28. Jian, R.; Chiu, D.M.; Hawe, W. *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems. Digital Equipment Corporation*; Eastern Research Laboratory: Hudson, MA, USA, 1984.
29. Ali, R.; Shahin, N.; Zikria, Y.B. Deep Reinforcement Learning Paradigm for Performance Optimization of Channel Observation–based MAC Protocols in Dense WLANs. *IEEE Access* **2018**, *99*, 3500–3511. [CrossRef]
30. Ali, R.; Qadri, Y.A.; Zikria, Y.B. Q-learning-enabled channel access in next-generation dense wireless networks for IoT-based eHealth systems. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 1–12. [CrossRef]