

# A New Aggregation Based Scheduling Method for Rapidly Changing IEEE 802.11ac Wireless Channels

Oran Sharon<sup>1</sup>, Yaron Alpert<sup>2</sup>

<sup>1</sup>Department of Computer Science, Netanya Academic College, Netanya, Israel

<sup>2</sup>Intel, Ra'anana, Israel

Email: [oran@math.haifa.ac.il](mailto:oran@math.haifa.ac.il), [Yaron.Alpert@intel.com](mailto:Yaron.Alpert@intel.com)

Received 15 June 2016; accepted 2 August 2016; published 5 August 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

In this paper we suggest a novel idea to improve the Throughput of a rapidly changing WiFi channel by exploiting the standard aggregation schemes in IEEE 802.11ac networks. The idea is based on blindly transmitting several copies of the first 4 MAC Protocol Data Units (MPDU) in the Transmission Window. This increases the probability that the window moves forward, enabling the transmission of new MPDUs and resulting in increased Throughput. It turns out that transmitting 2 copies of each of the first 4 MPDUs yields the best Throughput gain, in the order of 50% - 60% in PHY rates of 1.3 - 3.5 Gbps and few hundreds of bytes MPDUs. The proposed idea has advantage over Link Adaptation since it reduces the actual PHY rate only for the transmission of few MPDUs while Link Adaptation reduces the PHY rate for all the transmissions.

## Keywords

802.11ac, Aggregation, Scheduling, Link Adaptation, Reliability

---

## 1. Introduction

The IEEE 802.11 Standard (WiFi), created and maintained by the IEEE LAN/MAN Standards Committee (IEEE 802) [1], is currently the most important solution within the range of Wireless Local Area Networks (LAN). Since its first release in 1997 the standard provides the basis for Wireless network products using the WiFi brand, and it has been improved in many ways. One of the main goals of these improvements is to optimize the Throughput of the MAC layer, and to improve its Quality-of-Service (QoS) capabilities.

To fulfill the promise of increasing IEEE 802.11 performance and QoS capabilities, and effectively support

more client devices on a network, the IEEE 802.11 working group introduced the fifth generation in IEEE 802.11 networking standards, namely, the IEEE 802.11ac amendment, also known as Very High Throughput (VHT) [1] [2]. IEEE 802.11ac is intended to support fast, high-quality data streaming and nearly instantaneous data syncing and backup to notebooks, tablets and mobile phones. The IEEE 802.11ac final version, IEEE 11ac-2013, released in 2013 [2], leverages new technologies to provide improvements over the previous generation, *i.e.* IEEE 802.11-2012 [3].

The IEEE 802.11ac amendment improves the achieved Throughput coverage and QoS capabilities, compared with previous generations, by introducing improvements and new features in the PHY and MAC layers. In the PHY layer, IEEE 802.11ac (VHT) continues the long-existing trend towards higher Modulation and Coding rates (256 QAM 5/6 modulation), working in wider bandwidth channels (up to 160 MHz) and using 8 spatial streams that enable higher spectral efficiency.

In the MAC layer IEEE 802.11ac includes many of the improvements that were first introduced with IEEE 802.11n [3], also known as High Throughput (HT). A key performance feature first introduced in IEEE 802.11n MAC layer is the ability to aggregate packets in order to reduce transmission overheads in the PHY and MAC layers.

Frame aggregation is a feature of the IEEE 802.11n and IEEE 802.11ac Wireless LAN standards that increases Throughput by sending two or more consecutive data frames in a single transmission, followed by a single acknowledgment frame, denoted *Block Ack* (BACK). Aggregation schemes benefit from amortizing the control overhead over multiple packets. The achievable benefit from data aggregation is often of interest, especially in the face of several factors that can impact its performance, *e.g.*, link rates, error-recovery schemes, inter-frame spacing options, QoS guarantee, etc. IEEE 802.11n introduces, as a pivotal part of its MAC enhancements, three kinds of frame aggregation mechanisms: the Aggregate MAC Service Data Unit (A-MSDU) aggregation, the Aggregate MAC Protocol Data Unit (A-MPDU) aggregation and a Two-Level aggregation that combines both A-MSDU and A-MPDU. The last two schemes group several MAC Protocol Data Unit (MPDU) frames into one large Physical Service Data Unit (PSDU). IEEE 802.11ac also uses these three aggregation schemes, but enables larger MPDU and PSDU sizes.

The IEEE 802.11 standard also defines an Automatic Repeat-Request (ARQ) protocol that enables a transmitter to retransmit lost MPDUs and guarantee in-order reception of MPDUs at the receiver. This protocol is also used to improve the quality of the wireless channel.

Given the use of this protocol, *e.g.* in QoS constrained applications such as Voice and Video, in this paper we consider several methods to further improve the Throughput of the wireless channel by using aggregation. We consider methods in which some MPDU(s) are retransmitted several times in a lossy channel and in a single transmission attempt. We do not assume the model of Transmission opportunities (TXOP) [1], and a station transmits only one PSDU in every transmission attempt. We are not aware of any other research using aggregation to retransmit several copies of the same MPDU(s) in a single transmission attempt in order to increase Throughput.

Improving the quality of the wireless channel is also possible through Link Adaptation (LA) methods, in which a more robust Modulation/Coding Scheme (MCS) is used, at the cost of reducing the available PHY rate. However, there are scenarios in which the Signal-to-Noise-Ratio (SNR) is either rapidly changing, or it is changed in small amounts (dBs). In these cases LA is not used, either because the channel's SNR is not stable and it is changing faster than the LA tracing capability, or the changes are too small to trigger LA. For these scenarios we suggest methods to improve the Throughput of the wireless channel by using the new aggregation schemes.

Notice that by retransmitting MPDUs one actually reduces the PHY rate. However, we suggest in this paper to retransmit only *few* MPDUs, *i.e.* we reduce the PHY rate for only few MPDUs and not for *all* MPDUs as in LA. We show that such a change increases the Throughput considerably.

Another important point to mention is that we improve the Throughput of the *current* IEEE 802.11ac standard which has an upper bound of 64 MPDUs on the Transmission window size. We do not look for solutions that change the current standard as *e.g.* increasing the above upper bound.

Finally, our proposal is not mandatory in the sense that it should or should not be used by all the stations all together in a given time. For example, stations that are close to the AP and have a high SNR and so a low Packet Loss Rate should not use it. Stations that are located far from the AP and has a high Packet Loss Rate should use the proposal. In our later results we show when the proposal has high benefit and should be used.

## 1.1. Our Work

We consider a single pair of transmitter/receiver, over a WiFi wireless channel. Such a scenario is possible when the WiFi channel is used as a Point-to-Point Backhaul (Usage Models 4a, 4b in [4]). The transmitter transmits MPDUs to the receiver using the above mentioned ARQ protocol, and using the A-MPDU/Two-Level aggregation schemes defined in the IEEE 802.11ac standard [1]. We assume a saturated scenario where the transmitter has an infinite number of MPDUs to transmit. We also assume UDP like traffic; the receiver does not transmit above Layer 2 acknowledgments such as TCP Acks to the transmitter. It only transmits Layer 2 Acks. Therefore, the receiver does not contend on the channel and there are no collisions. As mentioned, we also do not assume the use in Transmission opportunities (TXOP) and only one PSDU is transmitted in every transmission event.

We investigate the performance of several methods to improve the Throughput by retransmitting several copies of the same MPDU(s) in a single transmission attempt, using aggregation.

Given the SNR of the wireless channel, the MCS and the aggregation scheme in use, there are two factors that influence the Throughput. First, the transmission success probability of an MPDU. Second, the *Transmission Window* ( $TW$ ) of the ARQ protocol. Only MPDUs in the  $TW$  are allowed for transmission. Therefore, lost MPDUs not only need to be retransmitted, thus wasting transmission time, but if failed MPDUs are located at the beginning of the  $TW$ , the  $TW$  cannot move forward over the MPDUs' sequence, and new MPDUs cannot be transmitted. This results in fewer MPDUs transmitted in a single transmission attempt, again reducing the Throughput.

Blinded retransmission of several copies of the same MPDU(s) in a single transmission attempt has two advantages: First, the success probability of an MPDU that is retransmitted several times is improved. Second, the probability that the *Transmission Window* moves forward, therefore containing new MPDUs, is also increased. A disadvantage of this approach is increasing the transmission time of the PSDU frame by the same data bits. Investigated in this paper is whether this increase is beneficial.

## 1.2. Our Results

We consider the A-MPDU and Two-Level aggregation schemes over several PHY rates: 433.3, 866.7, 1299.9 and 3466.8 Mbps. We consider four MAC Service Data Units' (MSDU) lengths: 128, 512, 1024 and 1500 Bytes. In the Two-Level aggregation we vary the number of MSDUs per MPDU in the range 1 - 7. We also consider several Packet Error Rates (PER): 0.50, 0.45, ..., 0.05. We show that the Throughput is improved using our methods, especially in low PERs, and the improvement can sometimes be in the order of tens of percentages ! Another important aspect of our proposed methods is that they are simple to implement and fully comply with the IEEE 802.11 standard [1].

## 1.3. Previous Works

The performance of the IEEE 802.11 protocol has been investigated in dozens of papers over the years. We only mention a few of those that relate to our current research. The first set of papers deals with the basic access scheme of IEEE 802.11. In [5] [6] the Throughput and Delay performance of the legacy transmission mode (no aggregation) are investigated, with upper and lower limits set on the Throughput and Delay achievable [5]. In [7]-[9] an analytical study of the Throughput of the basic IEEE 802.11, together with collisions, is performed, taking into account the Request-To-Send/Clear-To-Send (RTS/CTS) control mechanism [1]. In [10]-[14] the performance of the legacy transmission mode using Block Ack and RTS/CTS is investigated.

In [15]-[27] the Throughput and Delay performance of the A-MSDU, A-MPDU and Two-Level aggregation schemes is investigated. Several papers assume an error-free channel with-no collisions, several papers assume an error-prone channel and some papers also assume collisions. In [28]-[32] the performance of IEEE 802.11ac is investigated. Papers [29], [32] consider the performance of the aggregation schemes in IEEE 802.11ac and compare the performance of IEEE 802.11ac to that of IEEE 802.11n.

Another set of papers, e.g. [33]-[38], deals with QoS together with the aggregation schemes. In particular, in [38] the use of the ARQ protocol of the IEEE 802.11 standard [1], together with the aggregation schemes, is investigated in relation to QoS guarantee. In this paper we also investigate the use of the ARQ protocol with the aggregation schemes, but this time we investigate another aspect of the aggregation: blinded retransmission of several copies of the same MPDU(s) in the same transmission attempt, in order to improve the Throughput when Link Adaptation is not used. As far as we know, such an aspect of the aggregation schemes has not previously

been investigated.

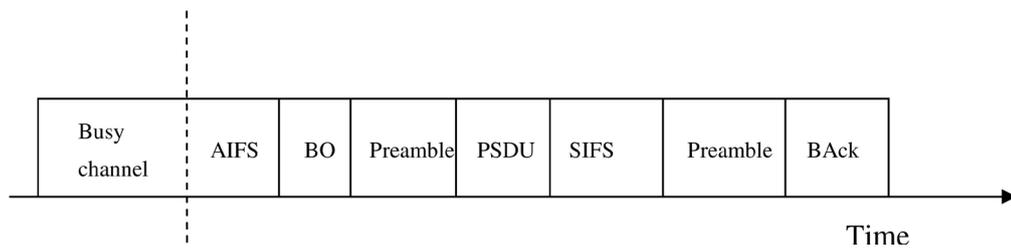
The rest of the paper is organized as follows: In Section 2 we describe the network model used. In Section 3 we show the performance of our methods. Section 4 is a summary of the paper.

## 2. Network Model

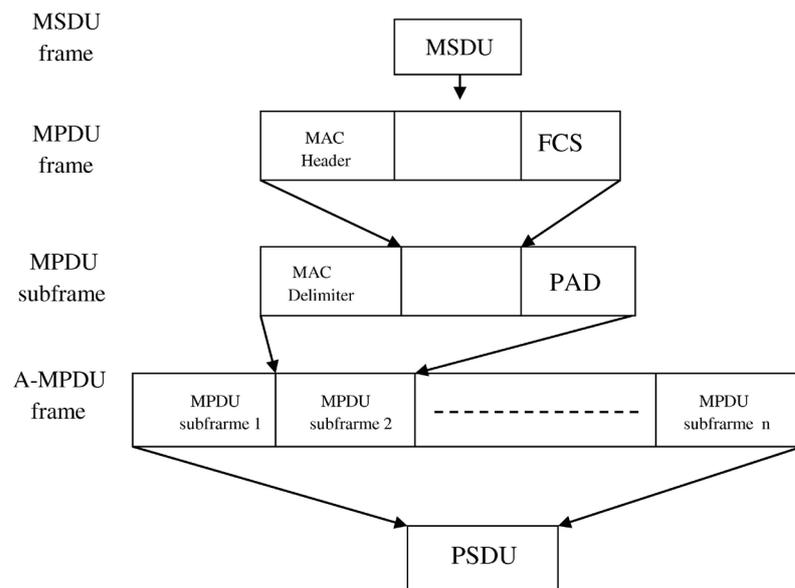
### 2.1. Successful Transmissions

In **Figure 1** we show the IEEE 802.11 channel access method in the case of successful transmissions, *i.e.* without collisions. This is the model that we consider in this paper. The channel access scheme in IEEE 802.11 is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol, which we assume the reader is familiar with. Therefore, we only remind the steps that a station performs in a successful transmission.

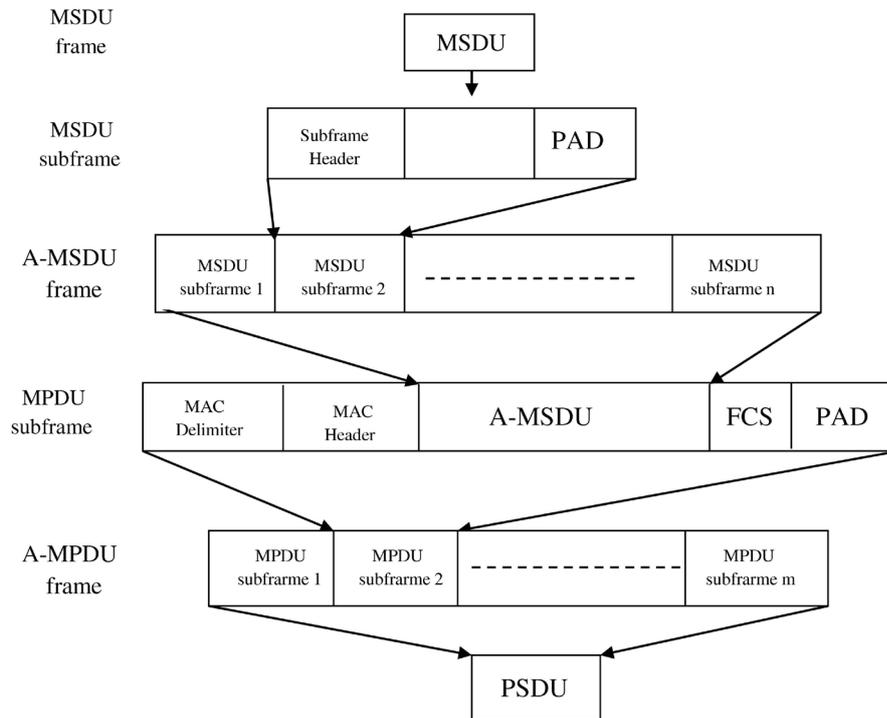
After a station senses an idle channel for a duration equal to the Arbitrary Inter Frame Space (AIFS) and BackOff intervals, it transmits its data frame, denoted PSDU. The PSDU's transmission is also preceded by a PHY preamble. After a SIFS the receiver acknowledges the reception. In the case of the A-MPDU and Two-Level aggregation schemes, the BAck frame is used. The BAck transmission is also preceded by a PHY Preamble. The process of generating the PSDU frame, in the case of the A-MPDU and Two-Level aggregations, is shown in the following **Figure 2** and **Figure 3** respectively. As mentioned, in this paper we assume UDP like traffic, where the receiver does not transmit above-Layer 2 acknowledgments. It only transmits the Layer 2 BAck acknowledgments. Therefore, there are no collisions. We also assume that the transmitter has an infinite number of MPDUs to transmit. Our proposed methods have significant under saturation scenarios only, where the loss of MPDUs delays the transmission of following MPDUs. Therefore, the transmission scenario in **Figure 1** repeats itself.



**Figure 1.** The activity on the channel in the case of a successful transmission.



**Figure 2.** The generation of an A-MPDU frame in A-MPDU aggregation.



**Figure 3.** The generation of A-MSDU and A-MPDU frames in two-level aggregation.

We assume that the transmitter is using the Best Effort Access Category and the following values are taken from the WiFi Alliance (WFA) publications [39]. The WFA is an organization that performs certification tests. The tests ensure reliability of the WiFi brand, and certification programs can be seen from the certified products.

In the Best Effort Access Category the AIFS is  $43 \mu\text{s}$ . The BackOff is a multiple of the *SlotTime* size, which for the OFDM PHY layer is  $9 \mu\text{s}$ . We assume that there are no collisions and so, on average, one half of the minimum BackOff interval is used, *i.e.*  $7.5 \times 9 = 67.5 \mu\text{s}$ . The duration of the PHY preamble, preceding the PSDU transmission, is changed according to the number of spatial streams in use [1]. In this paper it is  $43 \mu\text{s}$  for the most part, corresponding to 3 spatial streams. The SIFS is  $16 \mu\text{s}$ . The Block Acknowledgment (Back) frame is 32 bytes long. Its transmission time, denoted BackTime, is  $32 \mu\text{s}$ , using the Basic PHY Rate of 24 Mbps, and including the legacy PHY Preamble of  $20 \mu\text{s}$ . If the PHY rate  $R$  used for data frame transmissions is lower than 24 Mbps then  $R$  is also used for the Back transmission. However, in this paper we use  $R$ s with higher values than 24 Mbps.

## 2.2. The Aggregation Schemes

### 2.2.1. The A-MPDU Aggregation Scheme

The A-MPDU aggregation scheme is shown in Figure 2. Several MPDUs are inserted for transmission into one PSDU where each MPDU contains only a single MSDU. Up to 64 MPDUs, *with different sequence numbers*, are allowed in one PSDU. Such a PSDU is denoted *A-MPDU frame*. The MPDUs are separated by a MAC Delimiter of 4 octets, and every MPDU is rounded with its delimiter by a PAD, to a length that is an integral multiply of 4 octets.

The advantage of this aggregation scheme is that every MPDU is protected by its own Frame Control Sequence (FCS), so an MPDU either arrives successfully or un-successfully at the receiver, independent of the other MPDUs. The acknowledgment frame is now the Block Ack (Back) frame, which acknowledges every MPDU separately. Other advantages are the relatively small size of the MAC Delimiter and sharing of the AIFS, BackOff, PHY Preambles, SIFS and transmission of the Back frame overheads among several MPDUs. See Figure 1.

In IEEE 802.11ac the maximum length of an A-MPDU frame is 1,048,575 octets. The maximum length of an MPDU within an A-MPDU frame is 11,454 octets. However, neither of the above two size limits is ever reached because of the limit of 2304 bytes on the MSDU's size [3].

### 2.2.2. The Two-Level Aggregation Scheme

The Two-Level aggregation scheme is shown in **Figure 3**. In this aggregation scheme several MPDUs are inserted for transmission into one PSDU, as in the A-MPDU aggregation scheme. However, an MPDU can now contain several MSDUs. Such an MPDU is denoted Aggregated MSDU (*A-MSDU*). Every MSDU is preceded by a SubFrame Header of 14 bytes, and every MSDU with its SubFrame Header is rounded, by a PAD, to a size that is an integral multiple of 4 bytes. The Two-Level aggregation scheme achieves a better ratio than A-MPDU, between the amount of Data octets transmitted to the PHY and MAC layers' overhead.

In IEEE 802.11ac the maximum PSDU's size is 1,048,575 octets and the MPDU's maximum length is 11,454 octets. The upper limit of 64 MPDUs *with different sequence numbers* in the A-MPDU/PSDU also holds in this aggregation scheme.

### 2.3. The Error Model

We assume that the process of frame loss in a Wireless fading channel can be modeled with a good approximation by a low order Markovian chain, such as the two state Gilbert model [40] [41].

In this model the state diagram is composed of two states, "Good" and "Bad", meaning successful or unsuccessful reception of every bit arriving at the receiver, respectively. *Bit-Error-Rate* (BER) is the probability of moving from the Good state to the Bad state.  $(1 - \text{BER})$  is the probability of remaining at the Good state. According to the above model, the success probability of a frame of length  $B$  bits is  $(1 - \text{BER})^B$  and the failure probability  $p$  is given by Equation (1):

$$p = 1 - (1 - \text{BER})^B \quad (1)$$

By the above model one can see that as the frame length  $B$  increases, so does its failure probability.

Notice that errors in the MacDelimiter field(s) in an A-MPDU frame can make the receiver unable to detect the starting point(s) of subsequent MPDUs. In this paper the shortest MPDUs that we consider are of 168 bytes and the longest are of 1540 bytes. The MacDelimiter is 4 bytes and in the worst case about 2% - 3% of the MPDU's length, when considering MPDUs of 168 bytes. For MPDUs of 1540 bytes it is about 0.3%. Therefore, and as observed in real systems, the probability of not detecting the next MacDelimiter after a corrupted MPDU is very slight, and is therefore not mentioned in this paper.

### 2.4. Proposed Transmission Methods

In the A-MPDU and Two-Level aggregation schemes it is possible to transmit up to 64 MPDUs, *with different sequence numbers*, in an A-MPDU/PSDU frame [3]. In the *compressed BAcK* frame described in section 8.3.1.9.3 in [1] is a *Block Ack Bitmap* field containing 64 bits. Every one bit in this field acknowledges the reception of one MPDU in increasing order of sequence numbers, starting from the minimal sequence number that is also included in the frame. This is the reason for the limit of 64 MPDUs with different sequence numbers per A-MPDU frame.

Let  $K$  be the maximum number of MPDUs, with different sequence numbers, that are actually allowed in a PSDU frame.  $K$  ranges between 1 to 64. Notice however, that the standard does not prohibit the transmission of more than 64 MPDUs per A-MPDU frame, as long as there are at most 64 different sequence numbers, and the transmission time of the PSDU is not larger than 5.4 ms. This limit is derived from Equations (9)-(12) in Section 9.26.4 in [1]. We base our proposed methods on this observation.

We are given an infinite sequence of MPDUs to transmit from transmitter A to receiver B. All MPDUs are of the same length. Every MPDU has a probability  $1 - p$  to move successfully from A to B. This probability  $1 - p$  is the same for all MPDUs and all MPDUs' transmissions are independent.

We also have a *Transmission Window (TW)* of size  $W$ . This  $TW$  is part of the ARQ protocol defined in the IEEE 802.11 standard [3]. According to this ARQ protocol, only MPDUs within the  $TW$  are permitted for transmission from A to B. For example, if the MPDUs are numbered 1, 2, 3, 4, ... and  $W = 20$  MPDUs 1-20 to start, MPDUs 1-20 are in the  $TW$  and only these MPDUs are allowed for transmission. The  $TW$  slides over the MPDUs' sequence. After MPDU 1 is transmitted successfully the  $TW$  slides one position and now includes MPDUs 2-21 etc.

In every single transmission from A to B it is permissible to transmit up to  $K$  MPDUs with different sequence numbers,  $K \leq W$ .  $K$  is fixed and given. After the transmission, B notifies A which MPDUs arrived successfully.

MPDUs that did not arrive successfully must be retransmitted.

Let's assume for now that only a single copy of an MPDU can be transmitted in a given transmission. Let's also assume that  $K < W$ . At the first transmission  $K$  MPDUs are transmitted. Some arrive successfully and others not. However, if MPDU number 1 does not arrive successfully, the  $TW$  does not slide! Consider the extreme case where MPDU 1 does not arrive successfully over several transmissions. In such a scenario, the  $TW$  does not slide and a stage will occur where A will not have  $K$  different MPDUs to transmit to B!

Assume the A-MPDU aggregation scheme and that a station transmits  $X$  different MPDUs in a given A-MPDU frame; one copy of each MPDU. The throughput of this single transmission, denoted  $Thr$ , is defined in Equation (2):

$$Thr = \frac{8 \cdot L \cdot X \cdot P_{succ}}{C_1 + T_{sym} \left[ \frac{8 \cdot X \cdot 4 \left[ \frac{MacDelimiter + MacHeader + FCS + L}{4} \right] + 22}{BitsPerSymbol \cdot R} \right]} \quad (2)$$

From **Figure 1** we define  $C_1$  to be  $C_1 = AIFS + BackOff + PHYpreamble + SIFS + BACkTime$  (BACkTime contains the PHY Preamble preceding the BACk transmission). Assuming OFDM PHY layer,  $T_{sym}$  is 4  $\mu s$  and  $BitsPerSymbol$  equals 4.  $L$  is the MSDU's size in bytes and the additional 22 bits in the denominator are due to the SERVICE (16 bits) and TAIL (6 bits) fields that are added to every transmission by the PHY layer conv. protocol [3]. Finally,  $P_{succ}$  denotes the probability that an MPDU arrives successfully at the receiver.

The rationale behind our proposed methods is to blindly retransmit several copies of MPDUs at the beginning of the  $TW$ , in order to increase the probability that the  $TW$  will slide forward and will contain new MPDUs for later transmissions. We examine the cases of retransmitting only several copies of the 1st, 2nd, 3rd and 4th MPDUs in the  $TW$  respectively. These cases are denoted later by *Set1*, *Set2*, *Set3* and *Set4* respectively. As an extreme measure, we also check the possibility of retransmitting several copies of each of the MPDUs that are transmitted in a transmission attempt. This is done in *Set5* further on.

We now describe the five methods for retransmission of the MPDUs and compare between the Throughputs that these schemes achieve.

For example, assume there are  $I$  MPDUs in the  $TW$  that were received successfully at B. Let  $X = \min\{K, W - I\}$  and let  $X_{min}$  be the set of  $X$  MPDUs with the smallest indexes in  $TW$  that have not yet arrived successfully at B. The MPDU with the smallest index in  $X_{min}$  is the one at the beginning of the  $TW$ . Let's denote this MPDU by  $MPDU_{min}$ . For example, say  $W = 10$ ,  $K = 9$  and that only MPDUs 2, 4, 5, 6, 7 and 8 arrived successfully at B. Then  $I = 6$ ,  $X = \min\{9, 10 - 6\} = 4$ ,  $X_{min} = \{1, 3, 9, 10\}$  and  $MPDU_{min}$  is MPDU number 1. We use this example in the description of the schemes below.

- *Base*: The  $X_{min}$  MPDUs are transmitted, a single copy of each MPDU. In our example the transmission contains one copy of MPDUs 1, 3, 9, 10.
- *Set1*—1MPDU2, 1MPDU3, 1MPDU4, 1MPDU5: One copy of each MPDU in  $X_{min}$  is transmitted once, except to  $MPDU_{min}$ . In 1MPDU2 this MPDU is transmitted twice in every transmission attempt. In 1MPDU3 this MPDU is transmitted 3 times in every transmission attempt, and so on until 5 times.
- *Set2*—2MPDU2, 2MPDU3, 2MPDU4, 2MPDU5: This set of methods is similar to *Set1* but this time the first two MPDUs in a transmission attempt, *i.e.*  $MPDU_{min}$  and the next MPDU, are transmitted several times.
- *Set3*—3MPDU2, 3MPDU3, 3MPDU4, 3MPDU5: Same as *Set1* and *Set2*, except that now the first 3 MPDUs in a transmission attempt are retransmitted several times.
- *Set4*—4MPDU2, 4MPDU3, 4MPDU4, 4MPDU5: Same as *Set1*, *Set2*, *Set3* except that now the first 4 MPDUs in a transmission attempt are retransmitted several times.
- *Set5*—*All2*, *All3*, *All4*, *All5*: All the MPDUs in the transmission attempt are transmitted several times: In the *All2* method every MPDU is transmitted twice etc.

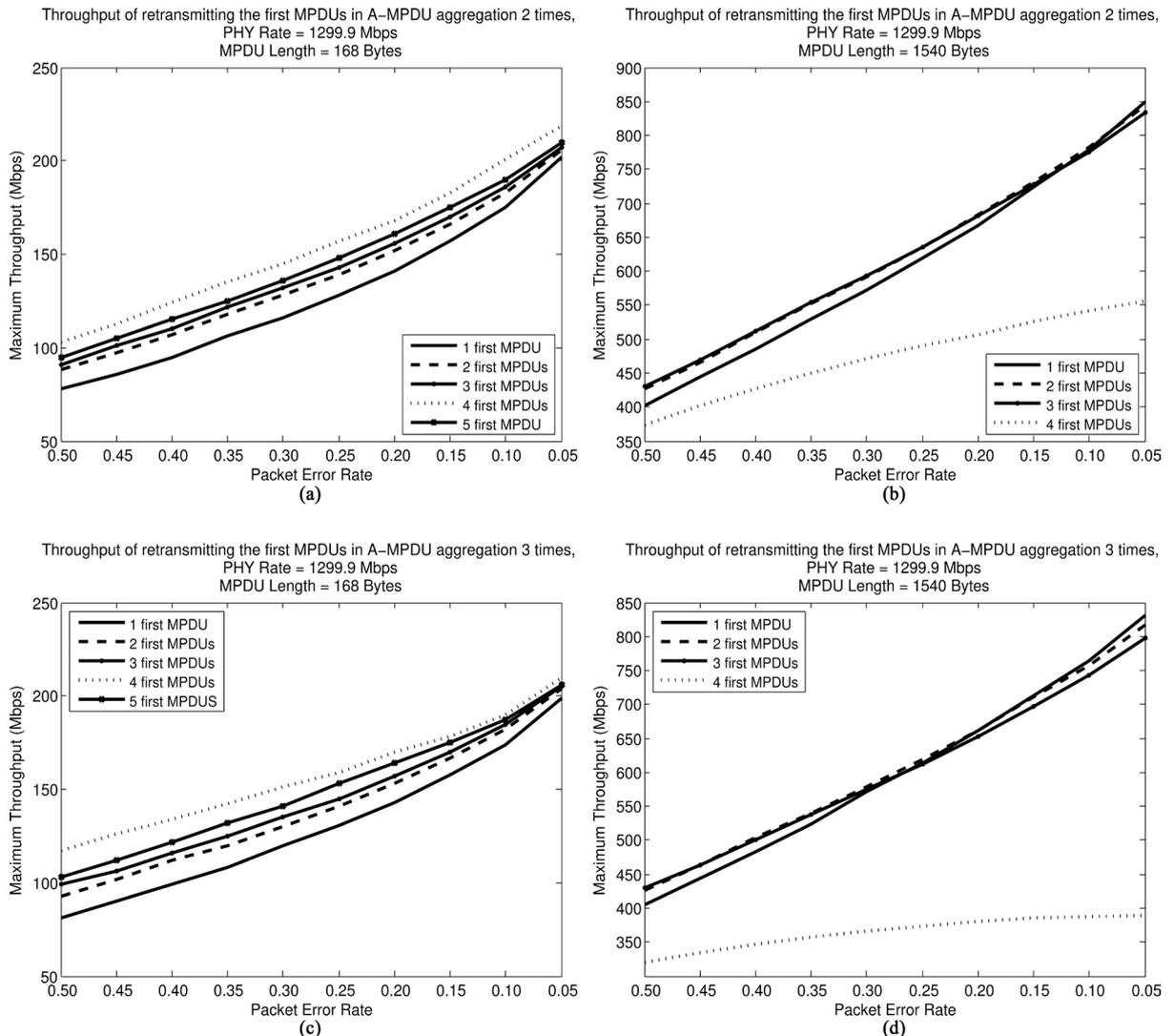
### 3. Performance Results

Our performance results are based on simulations. There is a difficulty in obtaining analytical results for the considered model [17] since basically we have here a kind of a Selective Repeat protocol where only a part of the transmission window is transmitted every time. Analysis of such a protocol is a subject to another research.

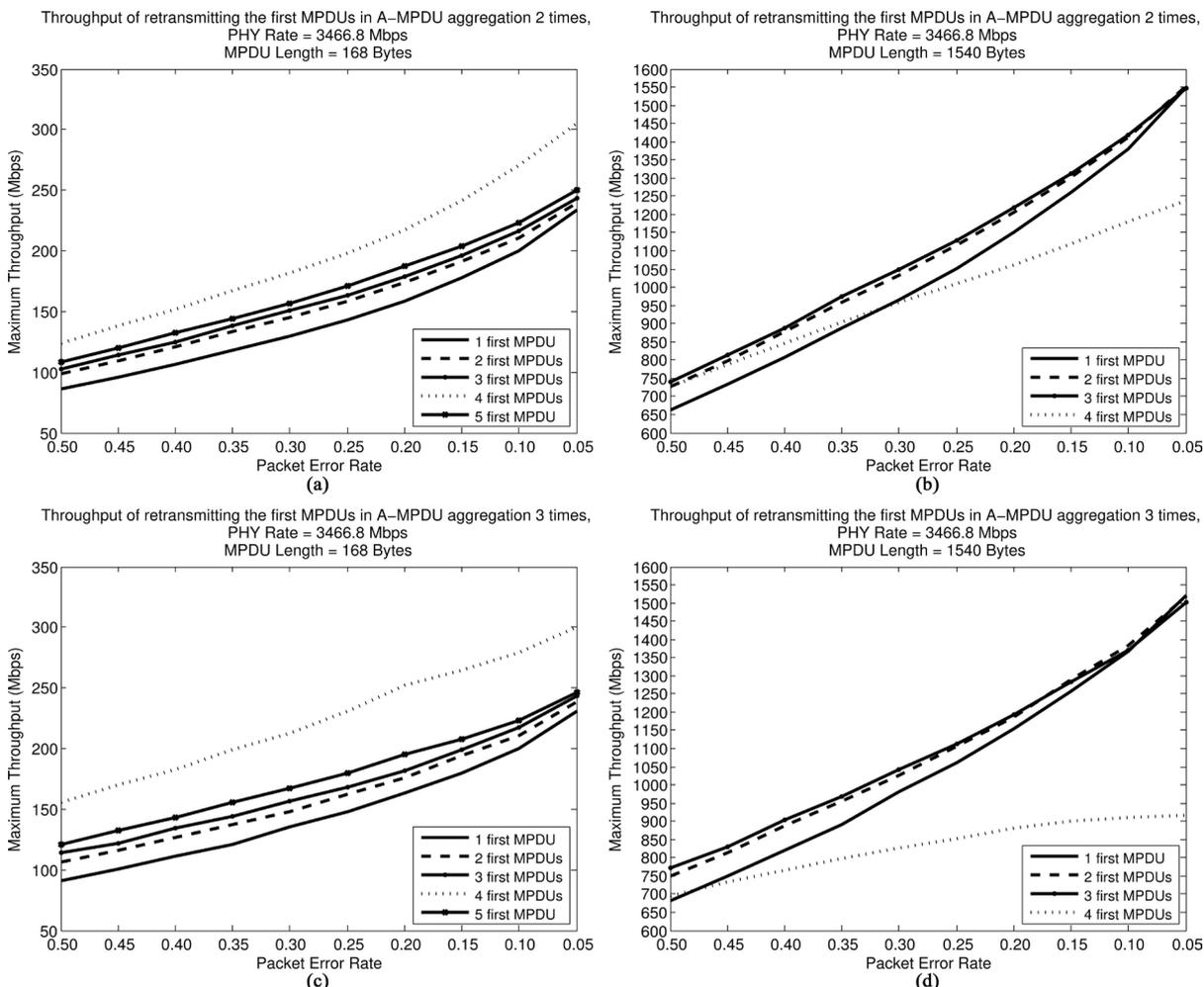
In all the simulations we set  $W$ , the size of the *Transmission Window*, to be 64, the maximum possible in the IEEE 802.11ac standard [1]. We checked the Throughput for all possible  $K$ , the number of MPDUs in every transmission,  $1 \leq K \leq 64$ , and picked the maximum Throughput that is achieved by any of the  $K$ s.

As mentioned, we consider 4 MSDUs' sizes, 128, 512, 1024 and 1500 bytes, which, with the MAC Header, MAC Delimiter, FCS and the rounding to an integral multiply of 4 bytes, become MPDUs of size 168, 552, 1064 and 1540 bytes respectively. We also consider 4 PHY rates: 433.3, 866.7, 1299.9 and 3466.8 Mbps. All these PHY rates correspond to Working Point MCS9. The first assumes 4 spatial streams and a 160 MHz channel. The other three assume a 80 MHz channel with 1, 2 and 3 spatial stream(s) respectively [1].

Our performance results are organized as follows: In **Figure 4** and **Figure 5** we justify our decision to retransmit only the first 4 MPDUs of an A-MPDU frame (Section 3.1). In **Figures 6-8** we show the performance of *Set1*, *Set2* and *Set3* for MPDUs of 1540 bytes respectively, and in **Figure 9**, **Figure 10** we show the performance of *Set1* and *Set4* for MPDUs of 168 bytes respectively (Section 3.2). In **Figure 11**, **Figure 12** we show the performance of *Set5* for MPDUs of 1540 and 168 bytes respectively (Section 3.3). In **Figures 13-16** we show the Throughput improvement over the *Base* method, taking into account all the suggested methods. In **Figure 17** we show that our methods improve the Throughput over a wide range of BER values (Section 3.4) and finally, in **Figure 18** we show results for the Two-Level aggregation (Section 3.5).



**Figure 4.** Throughput comparison when retransmitting the first 1 to 5 MPDUs in an A-MPDU 2 and 3 times, A-MPDU aggregation, MPDUs of 168 and 1540 bytes, PHY rate = 1299.9 Mbps.



**Figure 5.** Throughput comparison when retransmitting the first 1 to 5 MPDUs in an A-MPDU 2 and 3 times, A-MPDU aggregation, MPDUs of 168 and 1540 bytes, PHY rate = 3466.8 Mbps.

### 3.1. Why Set1-Set4 Only

In **Figure 4** and **Figure 5** we justify our decision to check the performance of retransmitting only the first 1 to 4 MPDUs in A-MPDU frames. The figures correspond to A-MPDU aggregation only, but the same results also hold for the Two-Level aggregation. In these figures we consider MPDUs of 168 and 1540 bytes. In **Figure 4** we assume a PHY rate of 1299.9 Mbps and in **Figure 5** we assume a PHY rate of 3466.8 Mbps. We see that for MPDUs of 168 bytes, **Figure 4(a)**, **Figure 4(c)** it is worthwhile to retransmit the first 4 MPDUs. Retransmitting the 5th achieves a smaller Throughput for all the PER values we consider. On the other hand, for MPDUs of 1540 bytes, **Figure 4(b)**, **Figure 4(d)** it is preferable to retransmit only the first 3 MPDUs.

These results can be explained as follows: The rationale behind retransmitting the first MPDUs is to enable the *TW* to slide and thus contain new MPDUs. The probability that the first 4 or 5 MPDUs will all corrupt is minimal; on the other hand retransmitting MPDUs increases the transmission time. For short MPDUs this increase is negligible. For larger MPDUs, such as 1540 bytes, this increase is more significant and therefore, retransmitting the 4th MPDU of this size is not efficient.

In **Figure 5** we show the same results for PHY rate of 3466.8 Mbps. In this case the ‘penalty’ for retransmitting MPDUs is lower, however the same relative results are still observed.

### 3.2. Performance of Set1-Set4

The next set of results corresponds to the A-MPDU aggregation scheme. In **Figure 6** we consider the first set of

methods, *Set1*, in which only  $MPDU_{min}$  is retransmitted. We assume MPDUs of 1540 bytes. We also consider 4 PHY rates: 3466.8, 1299.9, 866.7 and 433.3 Mbps, in **Figures 6(a)-(d)** respectively.

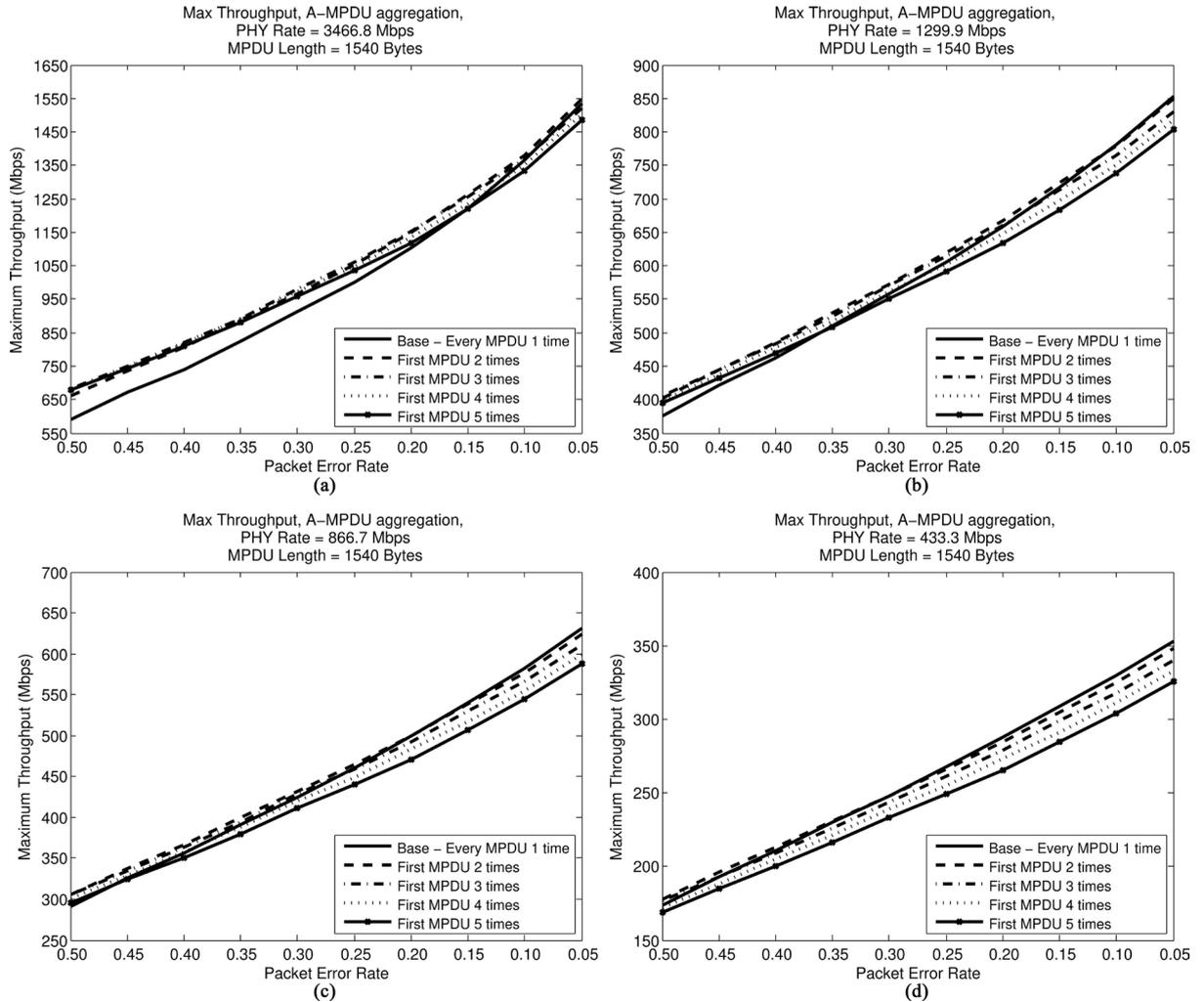
The curve of the *Base* method, in which every MPDU is transmitted once, is the best for PHY rates 866.7 and 433.3 Mbps except for very large PERs. Therefore, it can be concluded that in these PHY rates, none of the methods in *Set1* significantly improve the Throughput of the *Base* method. Notice that all the methods in *Set1* attempt to improve the Throughput by transmitting  $MPDU_{min}$  several times, but this is achieved at the cost of larger PSDUs and longer transmission times. In relatively small PHY rates this cost is high, and is not worth the improvement achieved by retransmitting  $MPDU_{min}$ .

For 3466.8 and 1299.9 Mbps *1MPDU2* achieves the largest Throughputs. In PER = 0.5 the improvements over the *Base* method are 12% and 5% respectively.

In **Figure 7** we show the same results as in **Figure 6**, but for *Set2* of methods. For PHY rates 866.7 and 433.3 Mbps there is a more significant improvement over the *Base* method compared to *Set1*, but for the larger rates of 3466.8 and 1299.9 Mbps the improvement is much more significant: 25% and 15% for PER = 0.5 respectively. Again, the retransmission of size 2, *i.e.* *2MPDU2* is the best method.

In **Figure 8** we show the results for *Set3* of methods. The results are similar to those for *Set1* and *Set2*. Notice that for PER = 0.5 the improvements in Throughputs over the *Base* method are 30% and 17% in the PHY rates 3466.8 and 1299.9 Mbps respectively.

In **Figure 9** and **Figure 10** we show the results for *Set1* and *Set4* respectively, when the MPDU size is 168



**Figure 6.** Maximum throughput when retransmitting the first MPDU in window, A-MPDU aggregation, MPDU size 1540 bytes.

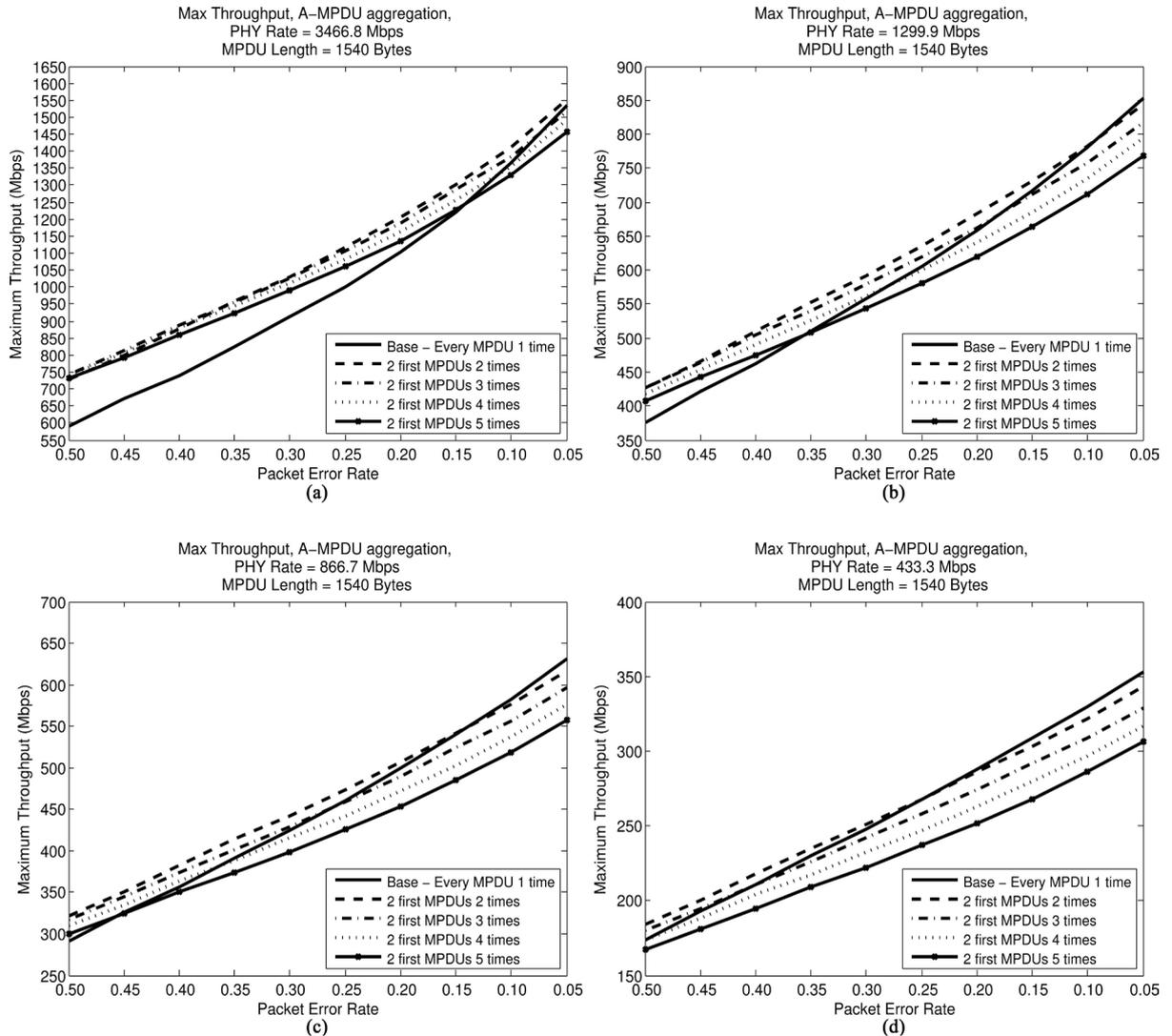
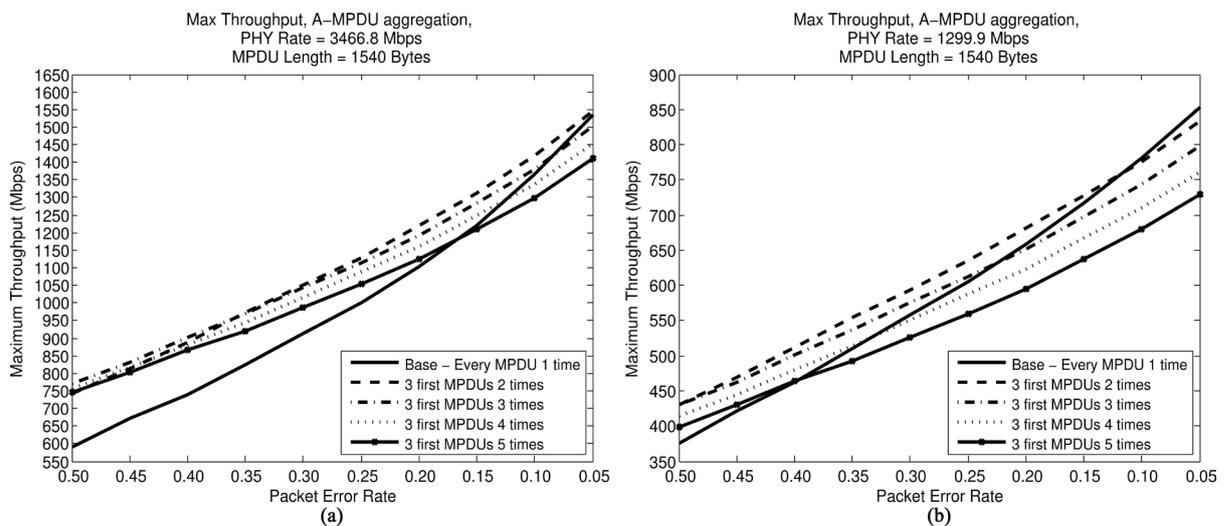


Figure 7. Maximum throughput when retransmitting the first 2 MPDUs in window, A-MPDU aggregation, MPDU size 1540 bytes.



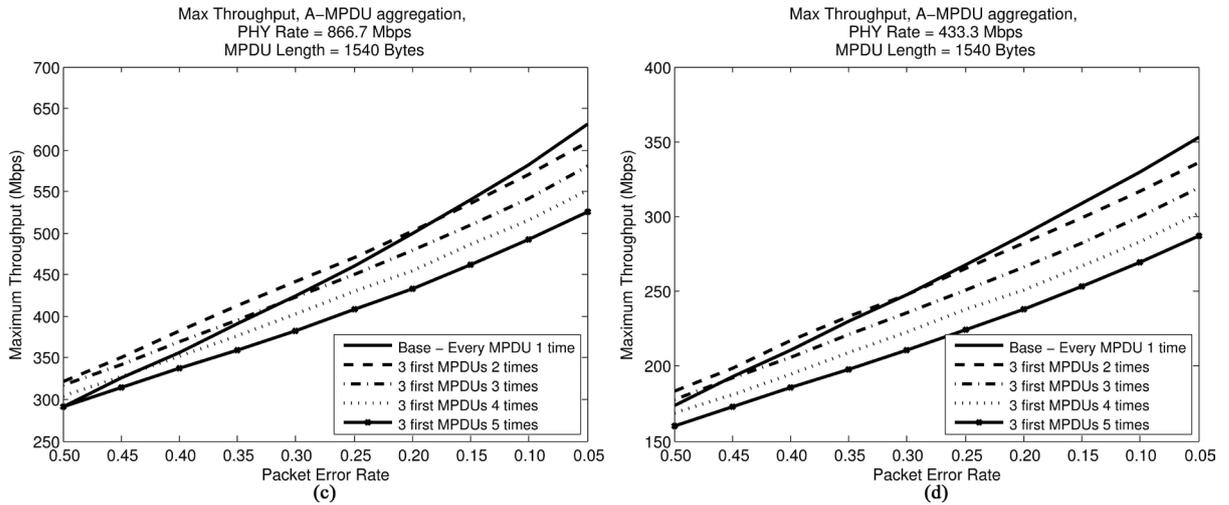


Figure 8. Maximum throughput when retransmitting the first 3 MPDUs in window, A-MPDU aggregation, MPDU size 1540 bytes.

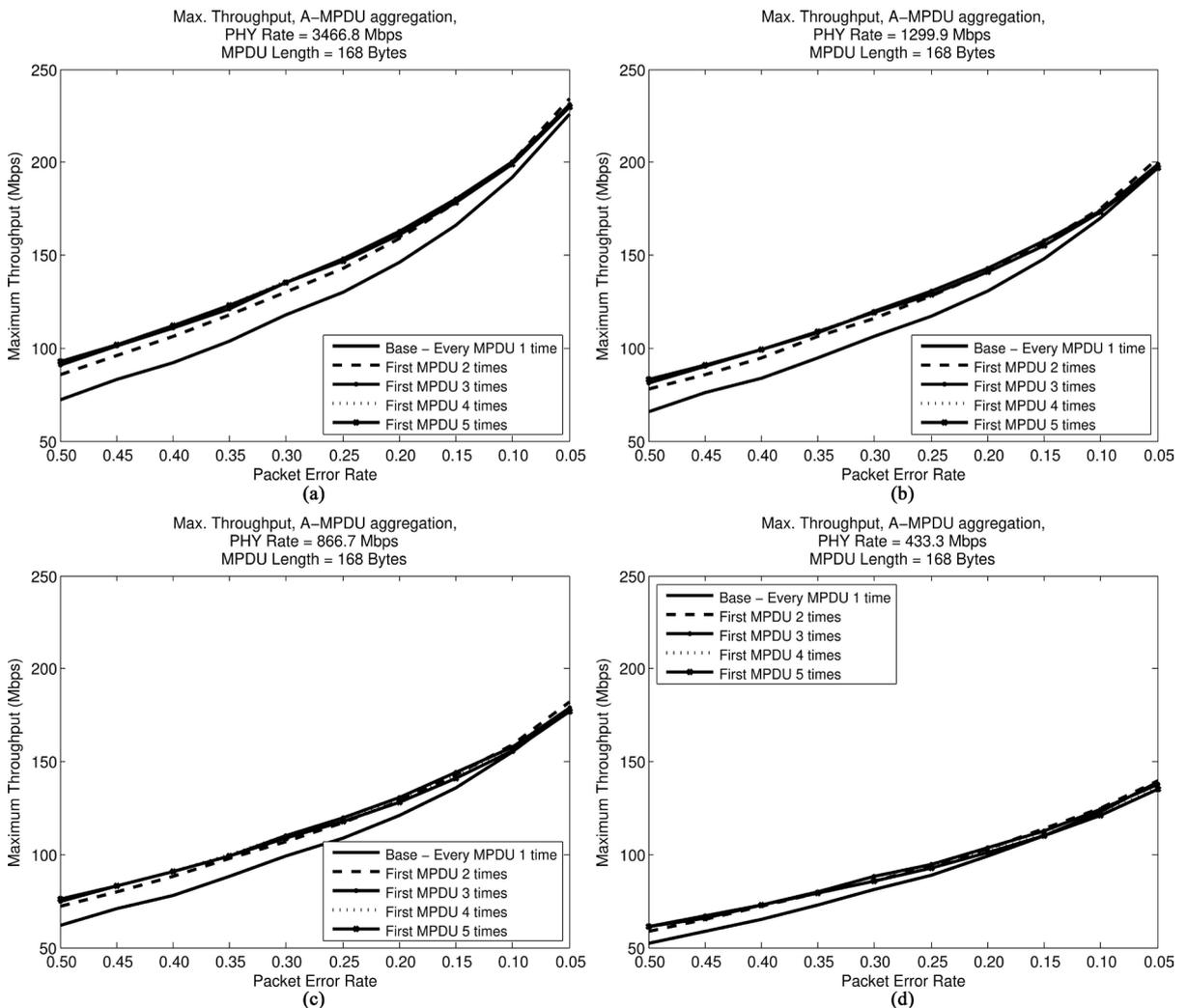
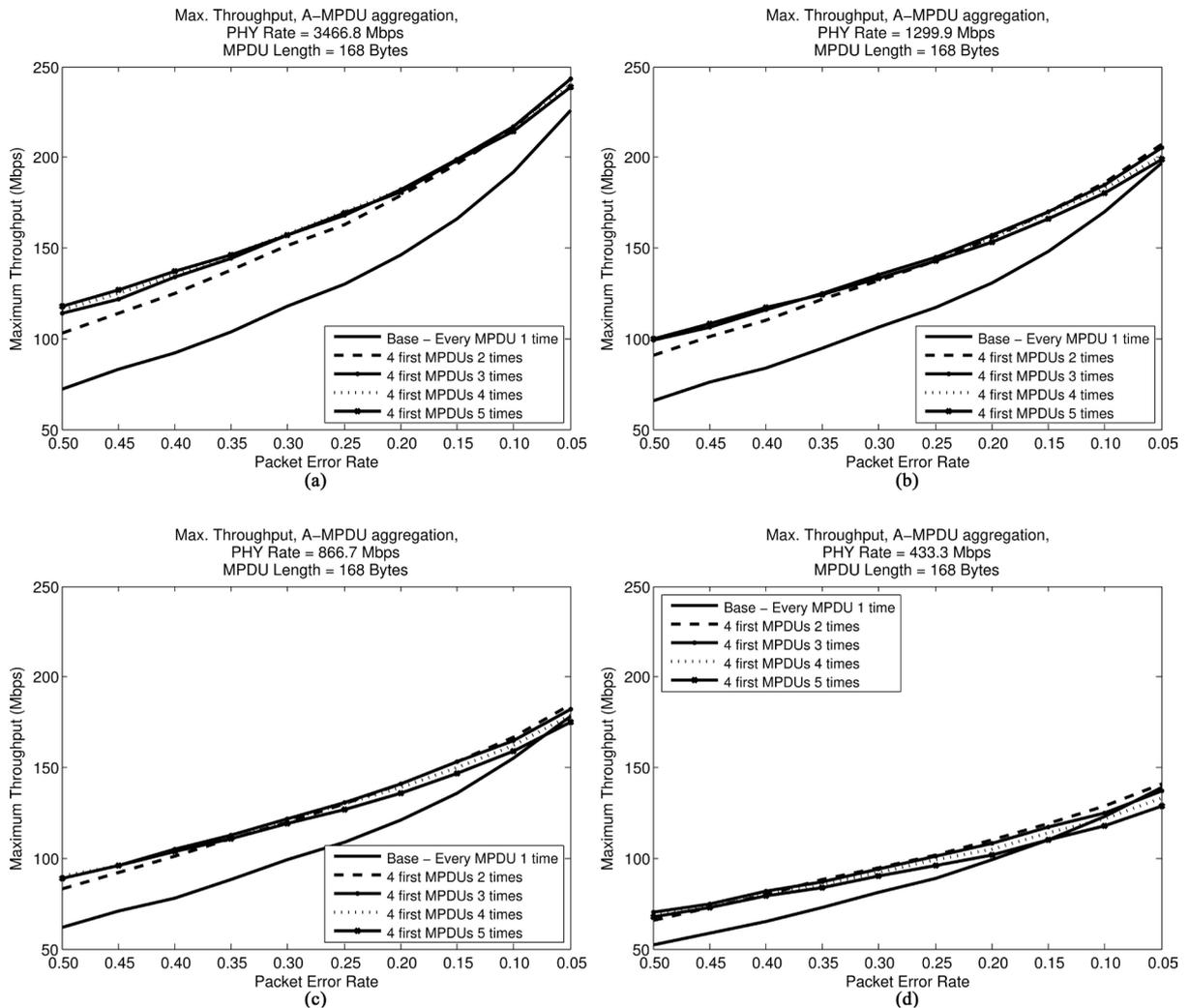


Figure 9. Maximum throughput when retransmitting the first MPDU in window, A-MPDU aggregation, MPDU size 168 bytes.



**Figure 10.** Maximum throughput when retransmitting the first 4 MPDUs in window, A-MPDU aggregation, MPDU size 168 bytes.

bytes. We omit the results for *Set3* and *Set4* as they fall between *Set1* and *Set4*.

One can see that in all the PHY rates there is an improvement in the Throughput over the *Base* method because the MPDUs are relatively short. Therefore, the penalty due to retransmissions is small compared to MPDUs of 1540 bytes. In *Set1*, [Figure 9](#), the maximum Throughput is always received for *1MPDU5*, i.e. it is worthwhile transmitting  $MPDU_{min}$  several times. For PHY rates 3466.8 and 1299.9 Mbps, and PER = 0.5, the improvements in the Throughput over the *Base* method are 29% and 25% respectively. When considering *Set4*, [Figure 10](#), it is not always worth transmitting the first 4 MPDUs 5 times when the PER is relatively low. The additional transmission time in these cases does not warrant the improvement in the Throughput. For low PERs it is worthwhile transmitting the first 4 MPDUs twice. For PER = 0.5 and PHY rates 3466.8 and 1299.9 Mbps the improvements in Throughput are 63% and 51% respectively.

The conclusion from this set of results is that small size MPDUs benefit more from *Set1-Set4* than large size MPDUs over a large scale of PHY rates, while large size MPDUs are beneficial only in large PHY rates. As a rule, transmitting 2 copies of MPDUs, i.e. *1MPDU2*, *2MPDU2*, *3MPDU2* and *4MPDU2* is the most efficient method.

### 3.3. Performance of *Set5*

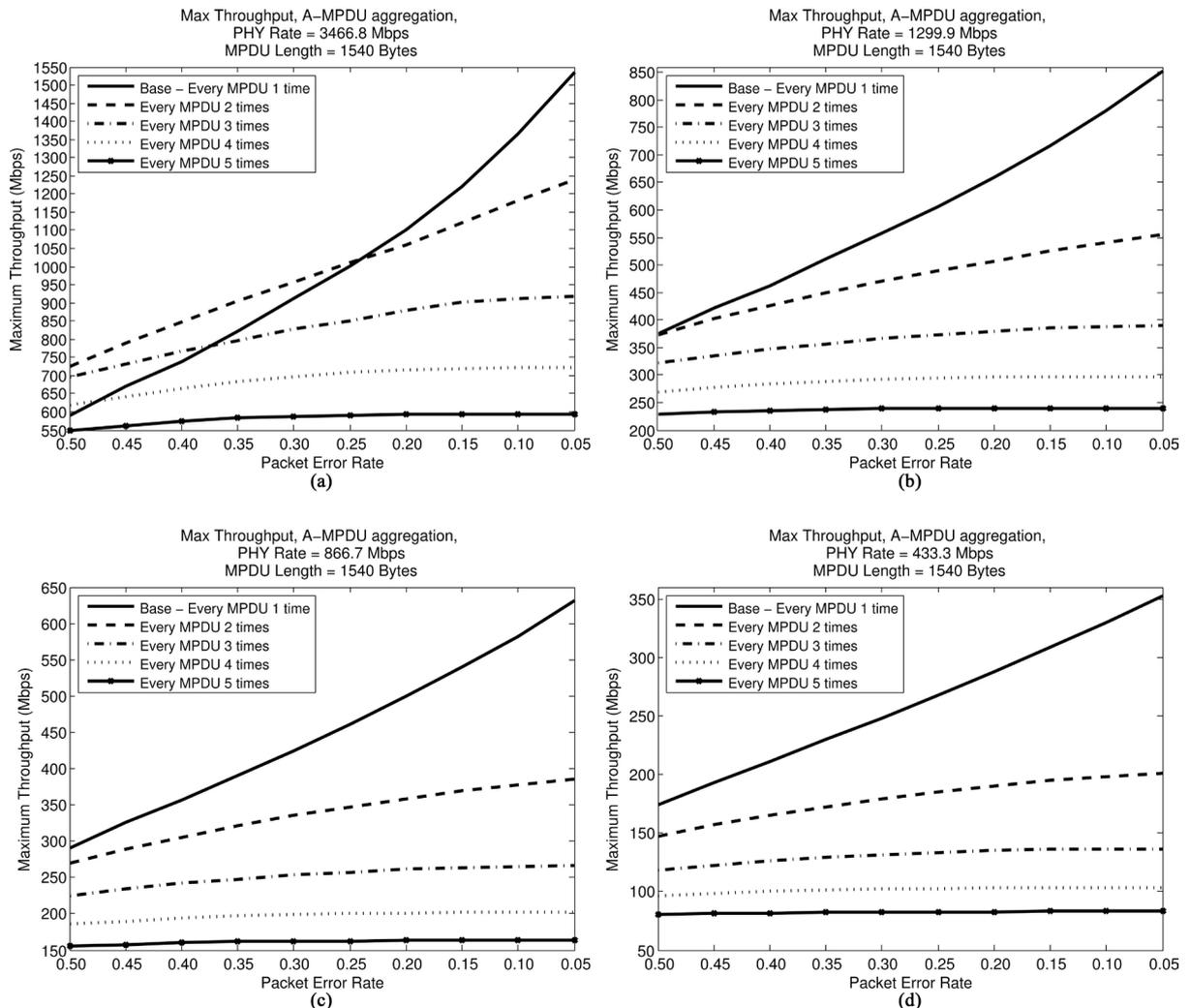
In [Figure 11](#) and [Figure 12](#) we show the results for *Set5* for MPDUs of 1540 and 168 bytes respectively, as well

as for all the considered PHY rates. For MPDUs of 1540 bytes, **Figure 11**, a significant improvement in the Throughput over the *Base* method appears only for a PHY rate of 3466.8 Mbps, and large PERs. For example, an improvement of 24% is observed in the case of PER = 0.5. In *Set5* all the MPDUs are retransmitted in the transmission; thus the PSDU and its transmission time increase significantly. This penalty is significant in relatively small PHY rates, and it becomes evident that only in the largest PHY rate the retransmission of all the MPDUs becomes beneficial.

For MPDUs of 168 bytes, **Figure 12**, a significant improvement over the *Base* method is achieved for all the PHY rates. For small size MPDUs the penalty in time, when retransmitting MPDUs, is minimal. Therefore, the retransmission is beneficial. Notice that for the small PHY rates, 433.3 and 866.7 Mbps, *ALL2* is most efficient, while for the 1299.9 and 3466.8 Mbps PHY rates *ALL2* and *ALL5* can also be efficient. In these PHY rates the time penalty of retransmissions is smaller. Thus, it is worth retransmitting MPDUs an additional number of times in order to increase their success probability.

### 3.4. Overall Throughput Improvement

In **Figures 13-16** we consider MPDUs of length 168, 552, 1064 and 1540 bytes respectively. In every figure we only consider PHY rates 3466.8 and 1299.9 Mbps. We show 2 curves: one for the Throughput of the *Base* method. The second indicates, for every PER, the maximum achievable Throughput considering all the new me-



**Figure 11.** Maximum throughput when retransmitting every MPDU in window, A-MPDU aggregation, MPDU size 1540 bytes.

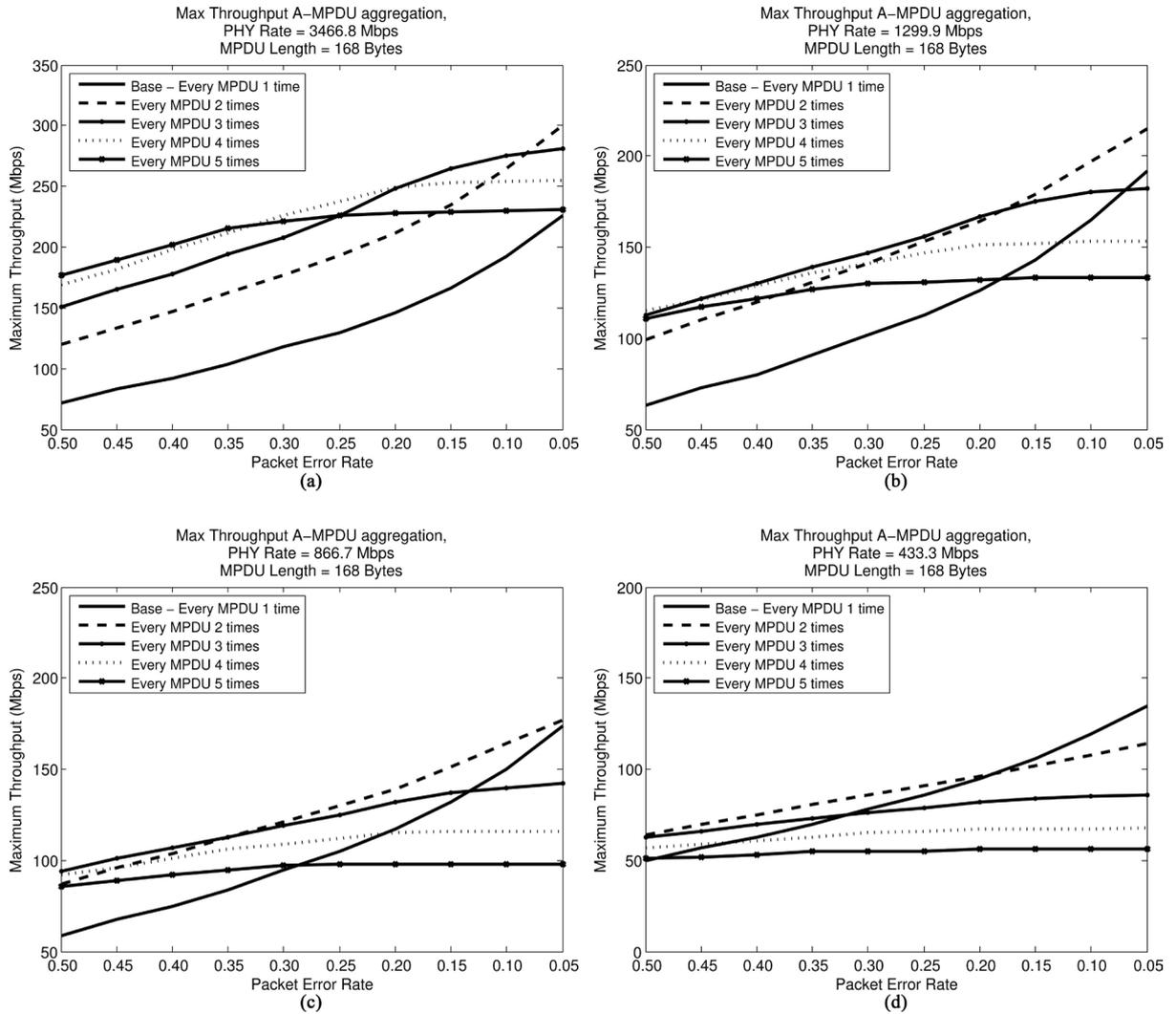


Figure 12. Maximum throughput when retransmitting every MPDU in window, A-MPDU aggregation, MPDU size 168 bytes.

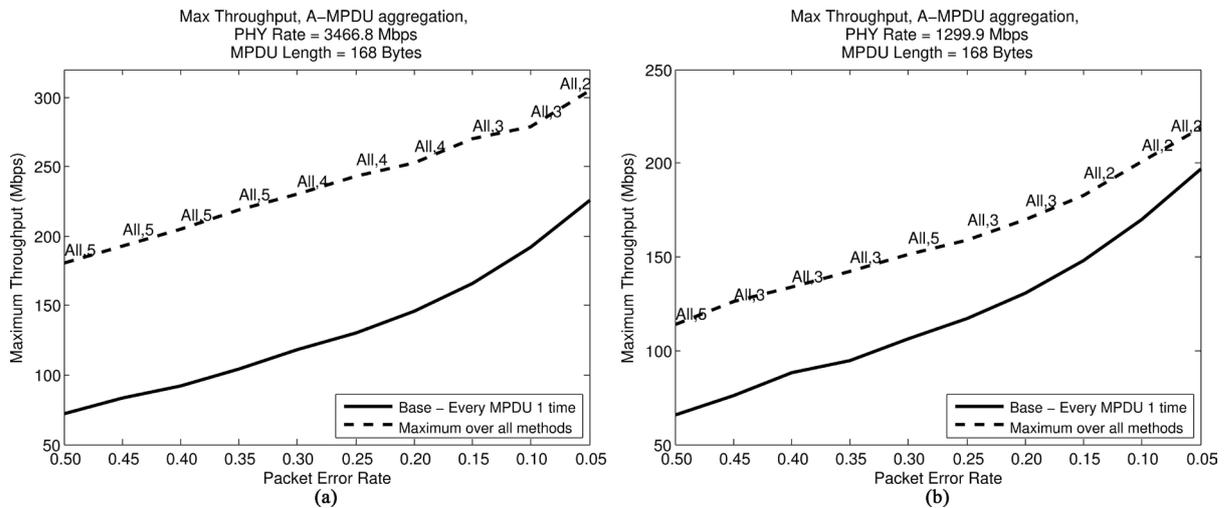


Figure 13. Maximum throughput over all methods vs. Base method, A-MPDU aggregation, MPDU size 168 bytes.

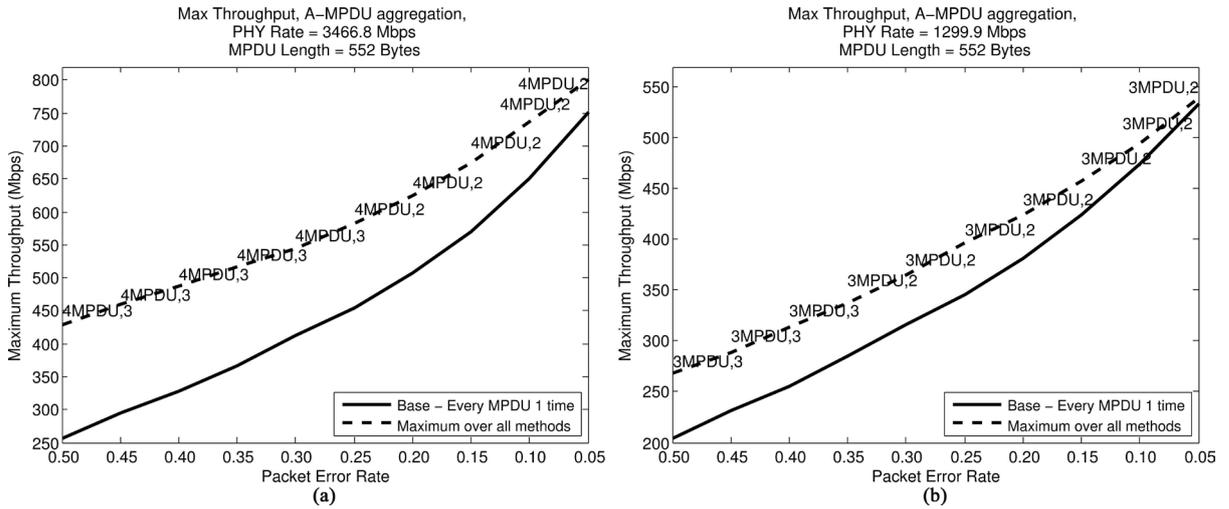


Figure 14. Maximum throughput over all methods vs. *Base* method, A-MPDU aggregation, MPDU size 552 bytes.

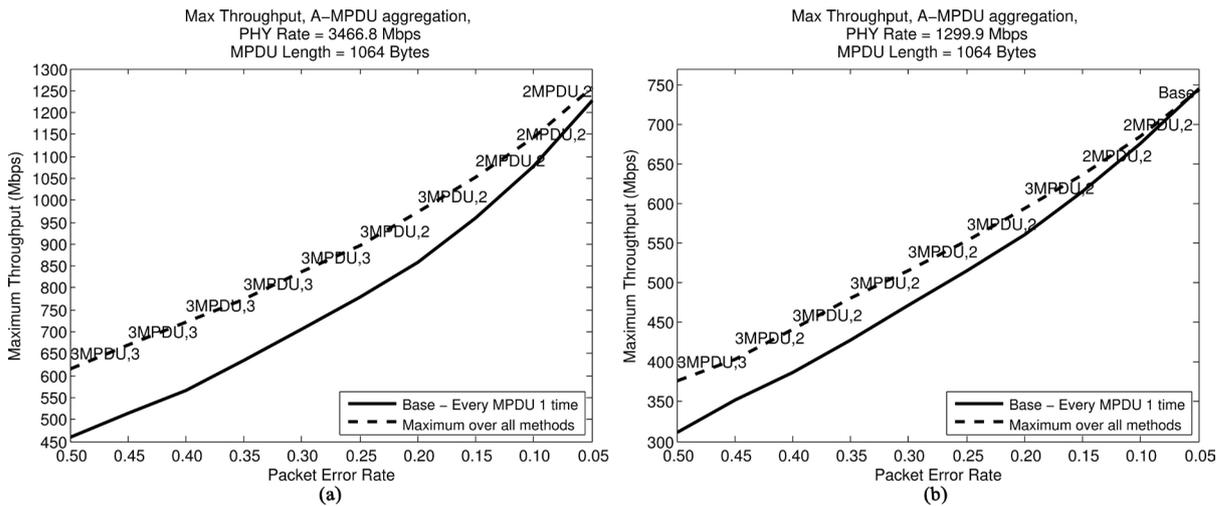


Figure 15. Maximum throughput over all methods vs. *Base* method, A-MPDU aggregation, MPDU size 1064 bytes.

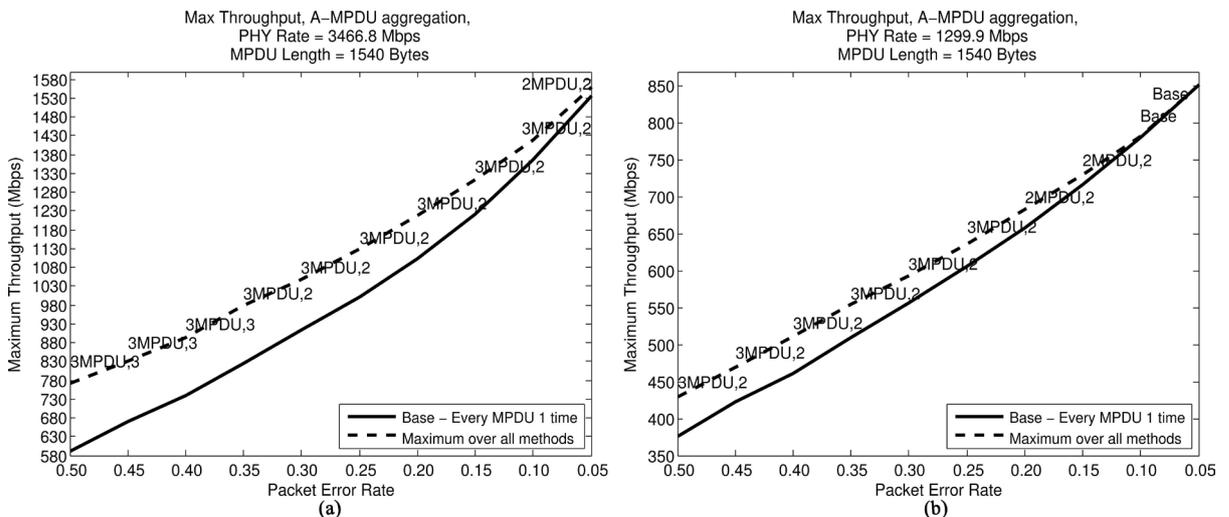


Figure 16. Maximum throughput over all methods vs. *Base* method, A-MPDU aggregation, MPDU size 1540 bytes.

thods. We also show the method that achieves the maximum Throughput. For example, in **Figure 13(a)** and  $PER = 0.5$ , it is most efficient to transmit each MPDU 5 times. At the same point in **Figure 14** it is best to use  $4MPDU3$ , *i.e.* to transmit the first 4 MPDUs 3 times.

From **Figures 13-16** one can see that for MPDUs of 168 bytes, the percentage of the Throughput improvement is the largest. For example, for PHY rate 3466.8 Mbps and  $PER = 0.5$ , it is 257%. For  $PER = 0.05$  it is 33%.

Clearly, as the  $PER$  decreases, the improvement in the Throughput decreases as well. The channel becomes reliable, the need for retransmissions is smaller and the penalty of longer PSDUs' transmission times is therefore more significant. The reason that smaller MSDUs show greater improvement (in percentage) in the Throughput is due to the overhead associated with every transmission. As the MSDU's length is smaller, the transmission time of an MSDU is smaller, and the size of the overhead ( $C_1$  in Equation (2)) is more significant. When the size of the overhead is more significant, the penalty of transmitting the same MSDU several times is relatively lower and therefore, the improvement in percentage is more significant. When later discussing the Two-Level aggregation method we provide an analytical explanation to the argument above.

Overall, from **Figures 13-16** one can see that the new methods improve the Throughput considerably, mainly in  $PER$ s in the range 0.10 - 0.50.

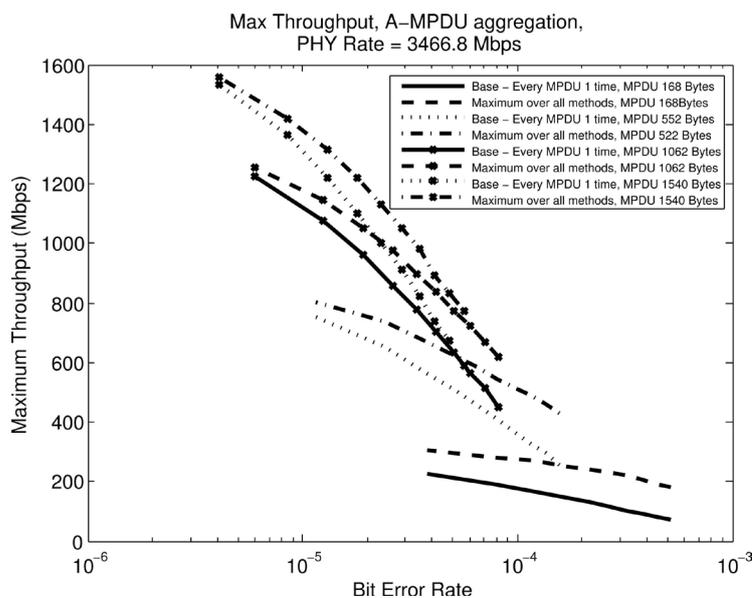
In **Figure 17** we show part (A) of **Figures 13-16** in one figure, but this time the x-axis is the BER. We normalized the  $PER$  in **Figures 13-16** into BER according to Equation (1) in order to show that the improvement in the Throughput is achieved over a wide range of BER values.

### 3.5. Throughput Improvement in Two-Level Aggregation

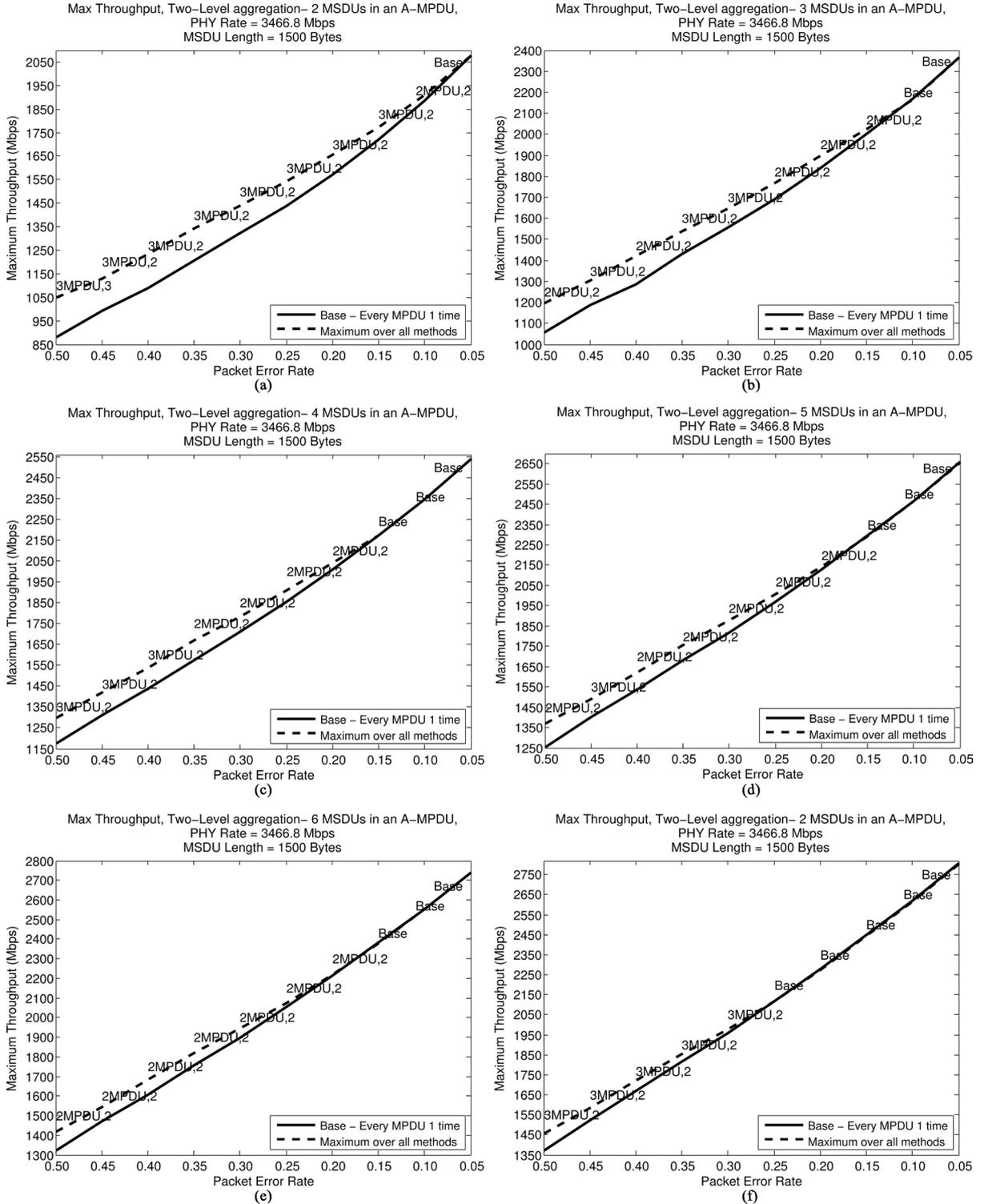
In **Figure 18** we consider the Two-Level aggregation, an MSDU's size of 1500 bytes and 2-7 MSDUs per A-MPDU frame in **Figures 18(a)-(f)** respectively. We again compare between the Throughput of the *Base* method to the maximum achieved by all the new methods.

One can see in **Figure 18** that as the number of MSDUs in an A-MPDU frame increases, the percentage of improvement by the new methods decreases. For  $PER = 0.5$  it is 20%, 14%, 10%, 9%, 7.5% and 5.4% for 2 to 7 MSDUs per A-MPDU respectively. This phenomena can be explained by some approximation as follows: Let's assume the case of 2 MSDUs per A-MPDU and let  $C_1$  be the overhead as defined in Equation (2). In the *Base* method let  $T$  be the transmission time of an average length PSDU and  $B$  the average number of bits that arrive successfully at the receiver.

Let's assume that the same new method  $S$  is best for all MSDUs in an A-MPDU. Notice that in  $S$ , and again assuming 2 MPDUs per A-MPDU frame, the transmission time of an average PSDU is now  $T + T_s$  because in



**Figure 17.** Maximum throughput over all methods and *Base* method vs. BER, A-MPDU aggregation, MSDUs' size 128, 512, 1024, 1500 bytes.



**Figure 18.** Maximum Throughput over all methods vs. *Base* method, Two-Level aggregation, 2-7 MSDUs per A-MPDU frame, MSDU size 1500 bytes.

$S$  several MPDUs are transmitted several times. Let  $B_s$  be the average number of bits that arrive successfully at the receiver. Then, the percentage of the improvement of  $S$  over the *Base* method, for 2 MSDUs per A-MPDU frame, appears in Equation (3):

$$PerformanceRatio = \frac{\frac{B_s}{C_1 + T + T_s}}{\frac{B}{C_1 + T}} \quad (3)$$

Lets assume now  $X$  MSDUs per A-MPDU frame, where  $3 \leq X \leq 7$  and  $X = 2 \cdot \alpha$ ,  $\alpha > 1$ . In this case, for the *Base* method  $B$  becomes  $\alpha B$  and  $T$  becomes  $\alpha T$ . Recall that we consider the same PER. For  $S$  it turns out that  $B_s$  becomes  $\alpha B_s$ , and  $T + T_s$  becomes  $\alpha(T + T_s)$ . Now the percentage of the improvement is as shown in Equation (4):

$$PerformanceRatio = \frac{\frac{\alpha B_s}{C_1 + \alpha(T + T_s)}}{\frac{\alpha B}{C_1 + \alpha T}} \quad (4)$$

One can easily verify that for  $\alpha > 1$  the percentage of improvement is lower, and it decreases as  $\alpha$  increases. Intuitively, given the same overhead  $C_1$ , multiplying the PSDU's transmission time at  $S$  by  $\alpha$  is more significant than at the *Base* method. This is because in  $S$  the PSDU's transmission time is larger. This reduces the attractiveness of  $S$  as  $\alpha$  increases.

## 4. Conclusions

We propose several methods to improve the Throughput of a WiFi channel based on aggregation and blindly retransmit several copies of the same MPDU(s) within a single transmission attempt. We examine the performance of retransmitting the first 1, 2, 3 or 4 MPDUs in the transmission attempt, and the case in which all the MPDUs in the transmission attempt are retransmitted the same number of copies. We show that there are cases, especially in large PERs and small size MPDUs, where a significant improvement in the Throughput is achieved.

This research is only a first step in exploring the retransmission of MPDUs within aggregation. The aim of this paper is mainly to introduce the idea and show its feasibility. It is possible now to implement the idea and evaluate its performance in any WiFi transmission scenario based on aggregation, e.g. in a WiFi cell where collisions are possible, and/or in TCP traffic where a single pair of transmitter/receiver can collide due to the transmission of above-Layer 2 Acks. Another research direction is to integrate this idea with the use of Link Adaptation.

## References

- [1] IEEE P802.11-REVmc<sup>TM</sup>/D5.2 (2016) Draft 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, New York.
- [2] IEEE Std. 802.11ac<sup>TM</sup> (2013) 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) Specific Requirements, Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz. IEEE, New York.
- [3] IEEE Std. 802.11<sup>TM</sup> (2012) 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, New York.
- [4] De Vegt, R. (2009) 802.11ac Usage Models Document. IEEE 802.11-09/0161r2.
- [5] Xiau, Y. and Rosdahl, J. (2002) Throughput and Delay Limits of IEEE 802.11. *IEEE Communication Letters*, 6, 355-357. <http://dx.doi.org/10.1109/LCOMM.2002.802035>
- [6] Lee, H.C. (2012) A MAC Throughput in the Wireless LAN. In: Guo, S., Ed., *Advanced Wireless LAN*, InTech, Bucheon City, Korea, 23-69.
- [7] Bianchi, G. (2000) Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Communication*, 18, 535-547. <http://dx.doi.org/10.1109/49.840210>
- [8] Chatzimisios, P., Boucouvalas, A.C. and Vitsas, V. (2005) IEEE 802.11 Wireless LANs: Performance Analysis and Protocol Refinement. *EURASIP Journal on Applied Signal Processing*, 1, 67-78.

- <http://dx.doi.org/10.1155/wcn.2005.67>
- [9] Carvalho, M.M. and Garcia-Luna-Aceves, J.J. (2003) Delay Analysis of IEEE 802.11 in Single-Hop Networks. *Proceeding of 11th IEEE International Conference on Network Protocols (ICNNP)*, Atlanta, 4-7 November 2003, 146-155. <http://dx.doi.org/10.1109/icnp.2003.1249764>
- [10] Wen, Y.F., Lin, F.Y.S. and Lai, K.W. (2004) Access Delay and Throughput Evaluation of Block Ack under 802.11 WLAN. *Proceeding of IASTED Communication and Computer Networks Conference*, Cambridge, 8-11 November 2004, Track 438-063.
- [11] Tinnirello, I. and Choi, S. (2005) Efficiency Analysis of Burst Transmissions with Block Ack in Contention-Based 802.11e WLANs. *Proceeding of IEEE International Conference on Communication (ICC)*, 5, 3455-3460. <http://dx.doi.org/10.1109/icc.2005.1495062>
- [12] Cabral, O., Segarra, A. and Velez, F.J. (2008) Implementation of IEEE 802.11e Block Acknowledgement Policies. *Proceeding of the World Congress on Engineering (WCE)*, London, 2-4 July 2008, 741-746.
- [13] Gross, J., Punal, O. and Emmelmann, M. (2009) Multi-User OFDMA Frame Aggregation for Future Wireless Local Area Networking. *Proceeding of IFIP Networking Conference*, Aachen, 11-15 May 2009, 220-233. [http://dx.doi.org/10.1007/978-3-642-01399-7\\_18](http://dx.doi.org/10.1007/978-3-642-01399-7_18)
- [14] Chen, H. (2012) Throughput Analysis of Block-Ack in IEEE 802.11n. *The 2nd International Conference on Computer Application and System Modeling (ICCA SM)*, Taiyuan, 27-29 July 2012, 956-959. <http://dx.doi.org/10.2991/iccas.m.2012.243>
- [15] Sidelnikov, A. and Choi, J.S. (2006) Fragmentation/Aggregation Scheme for Throughput Enhancement of IEEE 802.11n WLAN. *Proceeding of IEEE Asia Pacific Wireless Communications Symposium (APWCS)*, Daejeon, August 2006, 24-25.
- [16] Lin, Y. and Wong, V.W.S. (2006) Frame Aggregation and Optimal Frame Size Adaptation for IEEE 802.11n WLANs. *IEEE Global Telecommunications Conference (GLOBECOM)*, San Francisco, 27 November-1 December 2006, 1-6. <http://dx.doi.org/10.1109/glocom.2006.925>
- [17] Ginzbyrg, B. and Kesselman, A. (2007) Performance Analysis of A-MPDU and A-MSDU Aggregation in IEEE 802.11n. *IEEE Sarnoff Symposium*, Princeton, 30 April 2007-2 May 2007, 1-5. <http://dx.doi.org/10.1109/SARNOF.2007.4567389>
- [18] Skotdoulis, D., Ni, Q., Chen, H.H., Stephens, A.P., Kiu, C. and Jamalipour, A. (2008) IEEE 802.11n MAC Frame Aggregation Mechanisms for Next-Generation High-Throughput WLANs. *IEEE Wireless Communications*, **15**, 40-47. <http://dx.doi.org/10.1109/MWC.2008.4454703>
- [19] Kim, B.S., Huong, H.Y. and Sung, D.K. (2008) Effect of Frame Aggregation on the Throughput Performance of IEEE 802.11n. *Wireless Communications and Networking Conference (WCNC)*, Las Vegas, 31 March 2008-3 April 2008, 1740-1744. <http://dx.doi.org/10.1109/WCNC.2008.310>
- [20] Chan, K. (2009) Evaluation and Enhancements in 802.11n WLANs—Error-Sensitive Adaptive Frame Aggregation. Master Thesis, San Jose State University, San Jose. [http://scholarworks.sjsu.edu/edt\\_projects/79/](http://scholarworks.sjsu.edu/edt_projects/79/)
- [21] Wang, C.Y. and Wei, H.Y. (2009) IEEE 802.11n MAC Enhancement and Performance Evaluation. *Mobile Networks and Applications*, **14**, 760-771. <http://dx.doi.org/10.1007/s11036-008-0129-2>
- [22] Selvan, T. and Srikanth, S. (2010) A Frame Aggregation Scheduler for IEEE 802.11n. *National Conference on Communication (NCC)*, Chennai, 29-31 January 2010, 1-5. <http://dx.doi.org/10.1109/NCC.2010.5430156>
- [23] Zielinski, B. (2011) Efficiency Analysis of IEEE 802.11 Protocol with Block Acknowledge and Frame Aggregation. *Bulletin of the Polish Academy of Sciences Technical Sciences*, **59**, 235-243. <http://dx.doi.org/10.2478/v10175-011-0029-7>
- [24] Daldoul, Y., Ahmed, T. and Meddour, D.E. (2011) IEEE 802.11n Aggregation Performance Study for the Multicast. *IFIP Wireless Days (WD)*, Niagara Falls, 10-12 October 2011, 1-6. <http://dx.doi.org/10.1109/wd.2011.6098211>
- [25] Saif, A., Othman, M., Subramaniam, S. and Hamid, N.A. (2012) An Enhanced A-MSDU Frame Aggregation Scheme for 802.11n Wireless Networks. *Wireless Personal Communications*, **6**, 683-706. <http://dx.doi.org/10.1007/s11277-011-0358-8>
- [26] Kolap, J., Krishnan, S. and Shaha, N. (2012) Frame Aggregation Mechanism for High-Throughput 802.11n WLANs. *International Journal of Wireless & Mobile Networks (IJWMN)*, **4**, 141-153. <http://dx.doi.org/10.5121/ijwmn.2012.4309>
- [27] Kim, Y., Monroy, E., Lee, O., Park, K.J. and Choi, S. (2012) Adaptive Two-Level Frame Aggregation in IEEE 802.11n WLAN. *18th Asia-Pacific Conference on Communication (APCC)*, Jeju Island, 15-17 October 2012, 658-663. <http://dx.doi.org/10.1109/apcc.2012.6388276>
- [28] Park, N. (2011) IEEE 802.11ac: Dynamic Bandwidth Channel Access. *IEEE International Conference on Communications (ICC)*, Kyoto, 5-9 June 2011, 1-5. <http://dx.doi.org/10.1109/icc.2011.5963089>

- [29] Ong, E.H., Knecht, J., Alanen, O., Chang, Z., Huovinen, T. and Nihtila, T. (2011) IEEE 802.11ac: Enhancements for Very High Throughput WLANs. *IEEE 22nd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, Toronto, 11-14 September 2011, 849-853.
- [30] Chang, Z., Alanen, O., Huovinen, T., Nihtila, T., Ong, E.H., Knecht, J. and Ristaniemi, T. (2012) Performance Analysis of IEEE 802.11ac DCF with Hidden Nodes. *IEEE 75th Vehicular Technology Conference (VTC)*, Yokohama, 6-9 May 2012, 1-5. <http://dx.doi.org/10.1109/vetecs.2012.6240054>
- [31] Bellalta, B., Barcelo, J., Staehle, D., Vinel, A. and Oliver, M. (2012) On the Performance of Packet Aggregation in IEEE 802.11ac MU-MIMO WLANs. *IEEE Communication Letters*, **16**, 1588-1591. <http://dx.doi.org/10.1109/LCOMM.2012.081612.120744>
- [32] Sharon, O. and Alpert, Y. (2014) MAC Level Throughput Comparison: 802.11ac vs. 802.11n. *Physical Communication Journal*, **12**, 33-49. <http://dx.doi.org/10.1016/j.phycom.2014.01.007>
- [33] Changwen, L. and Stephens, A.P. (2006) Delayed Channel Access for IEEE 802.11e Based WLAN. 2006 *IEEE International Conference on Communication*, **10**, 4811-4817.
- [34] Skordoulis, D., Ni, Q., Min, G. and Borg, K. (2008) Adaptive Delayed Channel Access for IEEE 802.11n WLANs. *IEEE International Conference on Circuits and Systems for Communications (ICCSC)*, Shanghai, 26-28 May 2008, 167-171. <http://dx.doi.org/10.1109/iccsc.2008.42>
- [35] Li, T., Ni, Q., Malone, D., Leith, D., Xiao, Y. and Turletti, T. (2009) Aggregation with Fragment Retransmission for Very High-Speed WLANs. *IEEE/ACM Transactions on Networking*, **17**, 591-604. <http://dx.doi.org/10.1109/TNET.2009.2014654>
- [36] Skordoulis, D., Ni, Q. and Zarakovitis, C. (2009) A Selective Delayed Channel Access (SDCA) for the Highthroughput IEEE 802.11n. *IEEE Wireless Communication and Networking Conference (WCNC)*, Budapest, 5-8 April 2009, 1-6.
- [37] Camps-Mur, D., Gomony, M.D., Perez-Costa, X. and Silent-Ribes, S. (2012) Leveraging 802.11n Frame Aggregation to Enhance QoS and Power Consumption in Wi-Fi Networks. *Computer Networks*, **56**, 2896-2911. <http://dx.doi.org/10.1016/j.comnet.2012.05.004>
- [38] Sharon, O. and Alpert, Y. (2015) The Combination of Aggregation, ARQ, QoS Guarantee and Mapping of Application Flows in Very High Throughput 802.11ac Networks. *Physical Communication Journal*, **17**, 15-36. <http://dx.doi.org/10.1016/j.phycom.2015.06.002>
- [39] Wi-Fi CERTIFIED n Interoperability Test Plan Version v2.0.38 Published (2011).
- [40] Lemmon, J. (2002) Wireless Link Statistical Bit Error Rate Model. Technical Report 02-934, US Department of Commerce. <http://www.its.bldrdoc.gov/publications/download/TR-02-394.pdf>
- [41] Zorzi, M., Rao, R. and Milstein, L.B. (1995) On the Accuracy of a First-Order Markov Model for Data Transmission on Fading Channels. 1995 *4th IEEE International Conference on Universal Personal Communications*, Tokyo, 6-10 November 1995, 211-215.



**Submit or recommend next manuscript to SCIRP and we will provide best service for you:**

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>