BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 59, No. 2, 2011
DOI: 10.2478/v10175-011-0029-7

Efficiency analysis of IEEE 802.11 protocol with block acknowledge and frame aggregation

B. ZIELIŃSKI*

Institute of Computer Science, Silesian University of Technology, 16 Akademicka St., 44-100 Gliwice, Poland

Abstract. The article presents data transmission methods in IEEE 802.11 wireless LAN standard, including transmission efficiency improving methods like block acknowledge and two variants of frame aggregation. On the basis of their operation, an analytical model is derived which allows for estimation of protocol efficiency and effective throughput under perfect conditions for various protocol parameters. With this model, all transmission methods are analysed for their efficiency for two currently used physical layers, namely, OFDM (802.11a/g) and HT (802.11n). The calculation results are presented on graphs and discussed. Finally, so-called throughput upper limit is calculated for all the methods considered in the paper.

Key words: wireless LANs, IEEE 802.11, protocol performance, throughput upper limit.

1. Introduction

IEEE 802.11 standard [1] may be considered nowadays as the most important solution within the range of wireless local area networks (LANs). It is confirmed by relatively large number of devices that allow for transmission according to the standard, as well as constant progress in protocol modification and optimisation in order to achieve higher transmission rates, higher efficiency and support for QoS and multimedia applications. It is worth noting that after ten years of protocol existence, the transmission rate is one hundred times as high as at the beginning (it has grown up from 2 Mbps to 300 Mbps), while the devices themselves are nowadays few tens times cheaper. Moreover, frequency bands used belong to the ISM (Industrial, Scientific and Medical) range, thus, it is not required to obtain a licence to transmit. Therefore, we should not be surprised that the standard is used to form not only local area networks, but also point-to-point links of greater range, which is understandable when there are no real alternatives – GSM/GPRS/UMTS and WiMax networks need more expensive infrastructure. Additionally, WiMax networks are only at the beginning of their applicability, while GSM/GPRS/UMTS have limited transmission rate. From the other point of view, IEEE 802.11 standard is also used in applications for which wireless personal area networks (WPANs) were created, namely, to transmit multimedia and other time-bounded information – many such devices have IEEE 802.11 interface, and not, for example IEEE 802.15.3 that was defined just for these applications [2].

At the moment, when the standard first appeared (about 1999), most of network hardware vendors started to produce almost only network cards and access points. Later, other devices appeared, e.g., wireless bridges that allowed to wirelessly attach any Ethernet-equipped device, without a necessity to modify its hardware structure. Currently, IEEE 802.11-compatible interface is practically a mandatory equipment of laptops and palmtops, it can be also met in some digital satellite receivers, DVD players, digital photo cameras, home media centres or network cameras. It can be also used to carry Bluetooth frames – such possibility has been introduced in Bluetooth 3.0 [3].

Such a great number of IEEE 802.11-compatible network users, as well as its new applications, made it necessary to modify the standard. First of all, new physical layers, which allow for higher transmission rates (802.11 – 2 Mb/s, 802.11b – 11 Mb/s, 802.11a/g – 54 Mb/s), should be mentioned. Also, WPA-family protocols deserve our attention for their increased security level when compared to obsolete and broken WEP protocol. It is worth mentioning that 802.11-compatible networks are practically the only LANs where lack of sufficient security has been noticed and solved, in contrary to Bluetooth or IrDA. Latest modifications of IEEE 802.11 (802.11e) comprise QoS support for multimedia transmissions of various traffic classes and increase network performance by introduction of new acknowledgement strategies. In autumn 2009, new physical layer (802.11n) was finally defined. In this layer, in order to sustain high network efficiency, modifications leading to significant overhead reduction have been introduced. These solutions allow to achieve high effective throughput even at transmission rates of few hundreds Mbps. It is worth noting that "classical" frame exchange methods had a physical layer overhead of few tens percent, which caused insufficient transmission efficiency.

There are several papers which analyse the efficiency of 802.11 protocol, e.g., [4, 5]. In [5], a good explanation of transmission procedures and their influence on effective throughput is given using analytical methods. Using a similar approach, it has been proved [4], that "traditional" frame

*e-mail: Bartlomiej.Zielinski@polsl.pl

exchange rule has a throughput upper limit (TUL) of about 75 Mbps even when transmission rate is infinite. However, new transmission methods have been introduced in 802.11 standard since then. In this paper, using methods similar to those shown in [4, 5], these transmission procedures are analyzed and compared to the "traditional" one. Thus, the paper extends some results presented in [4].

The rest of the paper is organised as follows. First, data exchange methods are presented in the way that allows for protocol efficiency estimation. This comprises standard frame exchange, block acknowledge and two versions of frame aggregation. On the basis of presented considerations, formulas describing protocol efficiency for each frame exchange method are derived. Using these formulas, protocol efficiency for two physical layers – OFDM (Orthogonal Frequency Division Multiplexing, used in 802.11g) and HT (High Throughput, used in 802.11n) – is calculated for various transmission rates and data field capacity. Finally, the throughput upper limit is estimated for all frame exchange methods considered for two physical layers.

2. Data transmission in IEEE 802.11 standard

In IEEE 802.11 standard, data transmission may proceed according to few frame exchange procedures. For many years, only a single procedure has been defined. Further in this paper, it is referred to as basic frame exchange. Later, together with QoS enhancements, block acknowledge has been proposed in IEEE 802.11e in order to reduce protocol overhead by reducing number of acknowledge frames. Finally, 802.11n [6] introduces frame aggregation which allows even further overhead reduction by merging frames into long frame sequences.

The following analysis considers transmission under perfect conditions. Namely, we assume that:

- in the network, there are only two communicating stations that are always ready to exchange frames,
- there are no collisions or transmission errors, thus, no retransmissions take place,
- frame processing time is negligible.

2.1. Basic frame exchange. In basic frame exchange using DCF (Distributed Coordination Function) protocol, Data and Ack frames alternate. Each frame must be preceded by PLCP (Physical Layer Convergence Protocol) preamble and header. Thus, frame exchange process runs as presented in Fig. 1.

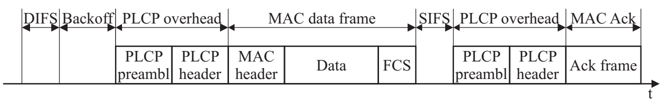

Fig. 1. Basic frame exchange in IEEE 802.11 standard

Bearing in mind frame exchange elements shown on the diagram, transmission cycle duration might be expressed as:

$$T_p = T_{\text{DIFS}} + T_{\text{BO}} + T_{\text{SIFS}} + 2 \cdot T_{\text{PLCP}} + T_{\text{Data}} + T_{\text{Ack}}, \quad (1)$$

where T_{DIFS} and T_{SIFS} are DIFS (Distributed Inter-Frame Space) and SIFS (Short Inter-Frame Space) duration, respectively, while T_{PLCP} – duration of PLCP preamble (T_{prmbl}) and header (T_{hdr}). These values are defined in specifications of individual physical layers and collected in Table 1.

Table 1
Physical layer parameters

Physical layer	$CW_{\min}$	$CW_{\max}$	T_{SIFS}	T_{slot}	T_{prmbl}	T_{hdr}	Additional overhead
OFDM	15	1023	16	9	20	4	$\geq$ 22 bits
HT	15	1023	16	9	16	16	–

In turn, T_{BO} represents backoff period duration, which, under perfect conditions and according to explanations given in [5], may be simplified to

$$T_{BO} = \frac{CW_{\min}}{2} \cdot T_{slot}. \quad (2)$$

T_{slot} is a slot time [s] (Table 1), while $CW_{\min}$ (Contention Window) – a minimal number of contention slots for a given physical layer. In turn, bearing in mind MAC frames formats,

$$T_{\text{Data}} = \frac{8 \cdot (28 + L)}{R_{wl}} \quad (3)$$

and

$$T_{\text{Ack}} = \frac{8 \cdot 14}{R'_{wl}}, \quad (4)$$

where L – data field capacity (often referred to as payload) in bytes, R_{wl} – Data frame transmission rate [bps], and R'_{wl} – Ack frame transmission rate [bps]. Within a single transmission cycle, exactly L data bytes are transmitted. During calculation of frame transmission times, we must take into account any additional overhead resulting from modulations used in a given physical layer, e.g., 32/33 encoding in FHSS (Frequency Hopping Spread Spectrum) as well as tail and pad bits in OFDM (Orthogonal Frequency Division Multiplexing) and ERP (Enhanced Rate Physical).

2.2. Block acknowledge. Block acknowledge mechanism allows for transmission of series of multiple data frames which are then commonly acknowledged. The acknowledge itself may be immediate or delayed; the first one is assumed to support higher transmission efficiency [1]. The Block Acknowledge procedure must be set up prior to transmission and torn down after the transfer is finished. Assuming that the information to be transmitted is sufficiently long, these initial and final frame exchanges do not play an important role from the point of view of protocol efficiency and thus they will not be further considered.

When using Block Acknowledge, transmission cycle consists of multiple (but no more than 64) Data frames. The latest of them is followed by the BlockAckReq frame, after which BlockAck frame appears. All frames are separated by the SIFS period and preceded by the PLCP preamble and header. On the data link layer level, BlockAckReq frame is 24 bytes long. BlockAck is even longer by 128 bytes as it carries fragmentation-specific information for every acknowledged frame (each Data frame may be split into up to 16 fragments, which needs 2 bytes of acknowledge bitmap for each frame). Information exchange process with block acknowledge is explained in Fig. 2.

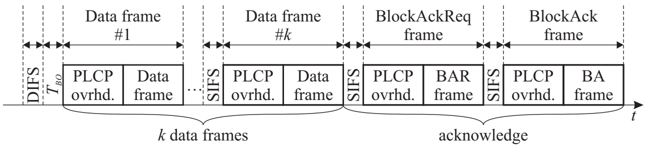

Fig. 2. Frame exchange in Block Acknowledge procedure

Bearing in mind transmission course described above, transmission cycle length may be expressed as:

$$T_p = T_{\text{DIFS}} + T_{\text{BO}} + (k+1) \cdot T_{\text{SIFS}} + (k+2) \cdot \\ \cdot T_{\text{PLCP}} + kT_{\text{Data}} + T_{\text{BAR}} + T_{\text{BA}}, \tag{5}$$

where T_{BAR} – transmission time of a BlockAckReq frame, equal to

$$T_{\text{BAR}} = \frac{8 \cdot 24}{R'_{wl}} \tag{6}$$

and T_{BA} – transmission time of a BlockAck frame, equal to

$$T_{\text{BA}} = \frac{8 \cdot (24 + 128)}{R'_{wl}}. \tag{7}$$

Assuming constant Data frame length, $L_D = k \cdot L$ data bytes are sent within a single transmission cycle.

2.3. Frame aggregation. Frame aggregation is introduced to reduce the PLCP overhead. As the PLCP frame format is set, the only way to reduce the overhead is using a single preamble and header for multiple Data frames. It is especially important for high transmission rates, because PLCP overhead is always transmitted at the lowest rate defined for a given physical layer. Thus, we can say that preamble and header transmission time is constant, while that of PSDU (Physical layer Service Data Unit) decreases with an increasing transmission rate. Therefore, the protocol overhead increases, while its efficiency – decreases. In order to avoid it, in IEEE 802.11n standard two aggregation methods are proposed: A-MSDU (MAC Service Data Unit) and A-MPDU (MAC Protocol Data Unit) [6].

A-MSDU aggregation. A-MSDU aggregation, similarly to Block acknowledge, allows for transmission of a series of Data frames, which are then commonly acknowledged. However, while Block acknowledge requires that each frame was an individual unit containing PLCP preamble and header, frame aggregation allows precede the entire series with a single preamble and header, which are common for all the Data frames. MAC header is also common for all these frames. Each of them is completed by a short, individual header.

When using A-MSDU aggregation, transmission cycle consists of a series of subframes containing individual headers. They are preceded by PLCP preamble and header and typical MAC header. All this information is protected by a common FCS (Frame Check Sequence) and acknowledged with a single Ack frame. Information exchange process with A-MSDU aggregation is explained in Fig. 3.

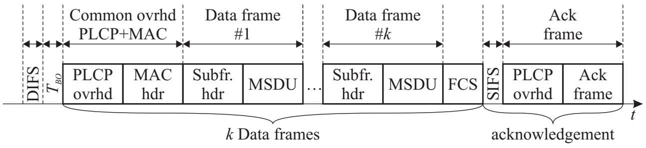

Fig. 3. Frame exchange in A-MSDU frame aggregation

Bearing in mind the transmission course described above, the transmission cycle length may be expressed as:

$$T_p = T_{\text{DIFS}} + T_{\text{BO}} + T_{\text{SIFS}} + 2T_{\text{PLCP}} + T_{\text{MAC}} + kT_{\text{SubFr}} + T_{\text{Ack}}, \tag{8}$$

where k – data block size, while T_{MAC} and T_{SubFr} – transmission times of MAC header and a subframe with its header, respectively. Bearing in mind their formats,

$$T_{\text{MAC}} = \frac{8 \cdot 28}{R_{wl}} \tag{9}$$

and

$$T_{\text{SubFr}} = \frac{8 \cdot 4 \cdot \left\lceil \frac{14 + L}{4} \right\rceil}{R_{wl}}. \tag{10}$$

The length of an aggregated frame is limited to 3839 or 7935 bytes, depending on capabilities of communicating stations. This limit may alter the number and length of subframes in two ways.

In the first method, the sender collects MSDU units of a constant size until the remaining buffer space is not sufficient to place a new unit. In this case, the number of aggregated frames equals to

$$k = \left\lfloor \frac{L_{\max}}{4 \left\lceil \frac{14 + L}{4} \right\rceil} \right\rfloor, \tag{11}$$

thus, the number of data bytes transmitted within a transmission cycle equals to

$$L_D = k \cdot L = \left\lfloor \frac{L_{\max}}{4 \left\lceil \frac{14 + L}{4} \right\rceil} \right\rfloor \cdot L. \tag{12}$$

In the second method, the sender collects MSDU units, and when the remaining buffer space is not sufficient to place a new unit, a shorter unit is added. Its length is selected so that the limit of an aggregated frame is utilised entirely. This variant is less real because of possible difficulties in its implementation. However, as it allows utilise frame length limit more efficiently, it should support higher efficiency. The number of aggregated MSDU units equals to

$$k = \left\lceil \frac{L_{\max}}{4 \left\lceil \frac{14 + L}{4} \right\rceil} \right\rceil, \tag{13}$$

while the number of data bytes transmitted within a transmission cycle equals to $L_{\max}$ decreased by organisation information (subframe headers and stuff bytes). As a result,

$$L_D = L_{\max} - \left\lfloor \frac{L_{\max}}{4 \left\lceil \frac{14 + L}{4} \right\rceil} \right\rfloor \cdot \left(4 \left\lceil \frac{14 + L}{4} \right\rceil - L \right) - 14. \tag{14}$$

Regardless of A-MSDU data field capacity, the total length of the aggregated frame equals always to $L_{\max}$.

A-MPDU aggregation. When using A-MPDU aggregation, transmission cycle consists of a block of Data frames. Entire block is preceded with only a single PCLP preamble and header. The Data frames are transmitted immediately one after another, without even a SIFS gap. The cycle ends with a slightly modified block acknowledge. In A-MPDU aggregation, BlockAckReq frame is not necessary because aggregation forces the use of block acknowledge. Besides, A-MPDU aggregation does not allow fragmentation, thus, BlockAck frame is substantially shorter than that of Block Acknowledge mechanism. The number of aggregated frames may not exceed 64, and the total length of an aggregated frame may not exceed 65535 bytes. Frame exchange process with A-MPDU aggregation is explained in Fig. 4.

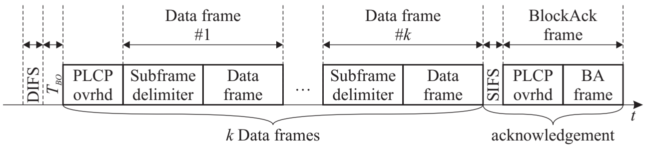

Fig. 4. Frame exchange in A-MPDU frame aggregation

Bearing in mind transmission course described above, transmission cycle length may be expressed as:

$$T_p = T_{\text{DIFS}} + T_{\text{BO}} + T_{\text{SIFS}} + 2 \cdot T_{\text{PLCP}} + kT'_{\text{Data}} + T'_{\text{BA}}, \quad (15)$$

where k – data block size, T'_{Data} represents Data frame transmission time with aggregation overhead:

$$T'_{\text{Data}} = \frac{8 \cdot \left(4 + 28 + 4\left\lceil \frac{L}{4} \right\rceil\right)}{R'_{wl}} \quad (16)$$

T'_{BA} represents transmission time of a modified BlockAck frame, equal to

$$T'_{\text{BA}} = \frac{8 \cdot (24+8)}{R'_{wl}}. \quad (17)$$

Number of aggregated frames may not exceed 64, and the total length of the aggregated frame may not exceed 65535 bytes.

3. Protocol efficiency analysis

For each frame exchange method aforementioned, we can calculate an effective throughput dividing the total data length transmitted within a transmission cycle by the cycle duration:

$$V_{ef} = \frac{L_D}{T_p}. \quad (18)$$

In turn, protocol efficiency can be expressed as

$$\eta = \frac{V_{ef}}{R_{wl}}. \quad (19)$$

The calculations were performed for two physical layers, namely, OFDM (Orthogonal Frequency Division Multiplexing used in 802.11a/g) and HT (High Throughput used in 802.11n). These two layers are used nowadays – OFDM is more popular while HT allows for higher transmission rates. It does not seem reasonable to consider older physical layers, because most of them is no longer used and – due to relatively low transmission rates – they perform efficiently enough even with basic frame exchange method.

The calculations were done for data field capacity (L) of:

- 2304 bytes (maximum for 802.11 standard),
- 1500 bytes (maximum for Ethernet, with which 802.11 often interoperates),
- 256 bytes (maximum for AX.25 protocol used in amateur Packet Radio network [7])
- 48 bytes (close to Ethernet minimum or ATM cell size [8]).

During calculations, it was assumed that acknowledge frames are always transmitted at the highest basic transmission rate, not exceeding the rate at which data frame was received. For example, in OFDM physical layers, basic rates are 6, 12 or 24 Mbps.

3.1. Basic frame exchange. Calculation results for basic frame exchange and OFDM physical layer is presented in Fig. 5, whereas for HT physical layer – in Fig. 6.

It can be easily seen that the efficiency of basic frame exchange is unsatisfactory. Although it performs sufficiently well for obsolete physical layers – i.e., DSSS and HR-DSSS (802.11b) – where transmission rate does not exceed 11 Mbps, it shows worse performance with OFDM physical layer, especially for higher transmission rates. Indeed, when longest frames are used (2304 data bytes), protocol efficiency on the data link layer degrades from about 94% at the transmission rate of 6 Mbps to less than 70% at 54 Mbps. Similar dependencies can be observed for other considered frame lengths. For the most often used frame length of 1500 data bytes, protocol performance degrades to less than 60% for 54 Mbps. It means that almost half of the transmission rate is lost for protocol overhead. Bearing in mind that we consider data link layer only and no higher layer protocols, we might expect even worse performance of the entire protocol stack. In this case, real transmission rate, even in perfect conditions (no retransmissions), might degrade to about 20–25 Mbps. These values are close to real network achievements.

In the case of HT physical layer (Fig. 6), the situation is dramatic. For transmission rates exceeding 100 Mbps, protocol efficiency for the longest frames falls below 50%. For rates exceeding 300 Mbps, it falls below 25% and for the highest transmission rate of 600 Mbps it equals to about 15%. If shorter frames are used, the situation is obviously worse. Thus, having a high-transmission speed physical layer is not enough to obtain high effective throughput even in perfect conditions. For example, the fastest transmission rate gives only about 90 Mbps effective throughput. Therefore, although the transmission rate is ten times as fast as in OFDM physical layer (54 Mbps), the effective throughput is only 3–4 times faster. It clearly shows that more effective frame exchange methods are really necessary if we want to obtain really high throughput.

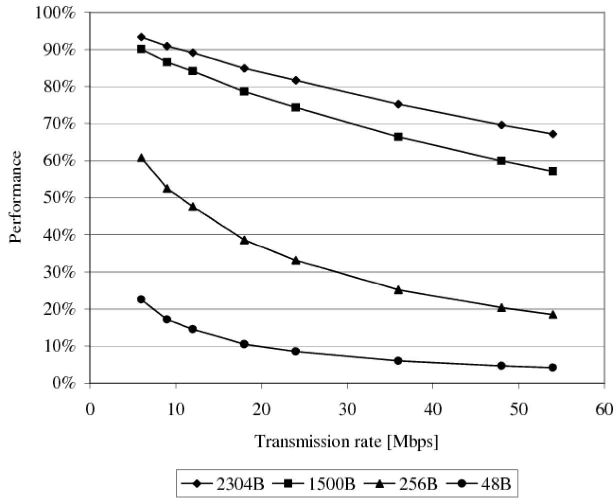

Fig. 5. 802.11 protocol performance for OFDM physical layer using basic frame exchange

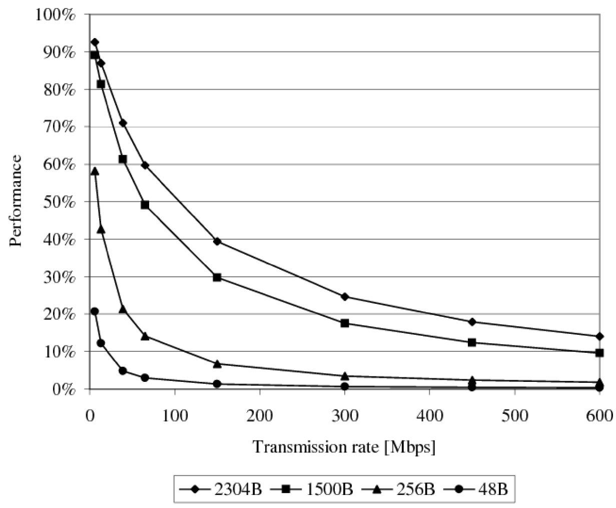

Fig. 6. 802.11 protocol performance for HT physical layer using basic frame exchange

3.2. Block acknowledge. Calculation results for block acknowledge with the OFDM physical layer are presented in Fig. 7, while with HT one – in Fig. 8.

It can be easily seen that the block acknowledge significantly increases the efficiency of 802.11 protocol with OFDM physical layer. When using the longest frames (2304 B) we get protocol efficiency of almost 90%, even for the highest transmission rate (54 Mbps); for 1500-bytes frames efficiency is only a little lower. Shorter frames cause visible protocol performance degradation, however, even for 256-bytes frames it is almost 50%. We can therefore say that the block acknowledge is a sufficient mechanism to get satisfactory performance of currently used 802.11a and 802.11g networks.

In the case of HT physical layer, the situation is not that optimistic. At the rates exceeding 100 Mbps, efficiency falls below 90% even when longest frames are used; at the maximum rate of 600 Mbps it falls below 40%. We can therefore say that block acknowledge – although it increases HT-based protocol performance – is not sufficient to assure effective operation of high transmission rate networks, even for the maximum block size. We can say, that HT physical layer with block acknowledge has similar performance as OFDM physical layer with “traditional”, basic frame exchange method.

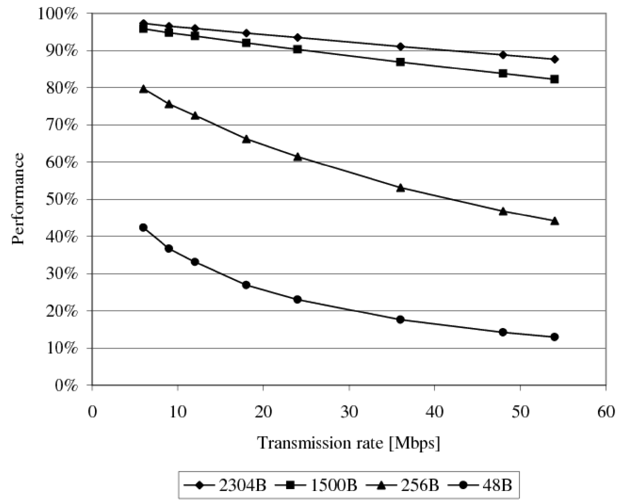

Fig. 7. 802.11 protocol performance for OFDM physical layer using block acknowledge

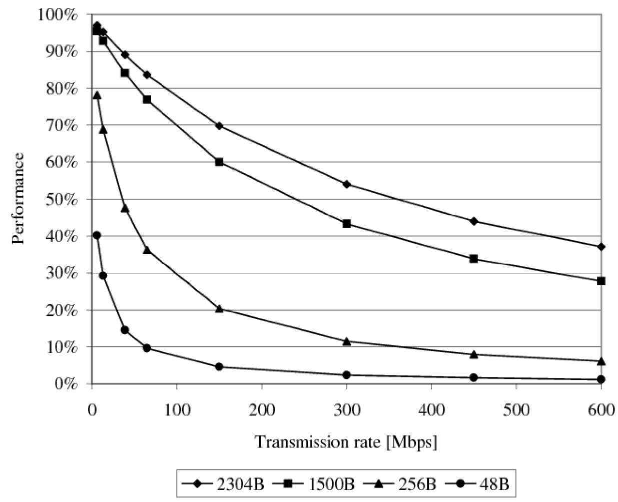

Fig. 8. 802.11 protocol performance for HT physical layer using block acknowledge

When the block contains a small number of data frames, block acknowledge may not bring performance grow. Even if we neglect overhead resulting from block acknowledge setup and teardown, block acknowledge frame (BlockAck) is much longer than the “traditional” one (Ack). Thus, it seems necessary to estimate minimum number of data frames for which transmission using block acknowledge increases network performance when compared to basic frame exchange. It shows, that for high transmission rates, e.g., 54 Mbps and above, block acknowledge is more efficient than traditional one even when the block size is $k \geq 2$, while for lower rates, e.g., 6 Mbps, when $k \geq 3$. This property does not depend on size of data frames. Thus, we can say that block acknowledge should be used whenever possible in order to bring network efficiency improvement.

3.3. A-MSDU aggregation. Calculation results for 802.11 protocol with A-MSDU aggregation in both considered versions and $L_{\max}$ limited to 3839 bytes are presented in Fig. 9. It shows that the variant with filling up to the limit brings advantage only for long frames (1500 and 2304 data bytes), while for short frames (48 and 256 data bytes) results for both variants are similar, and the curves on the graph overlap on each other. The difference between variants with or without filling up to the limit are most visible at the highest transmission rate (54 Mbps). For frames of 2304 bytes this difference is about 10%, for 1500-bytes frames – about 5%.

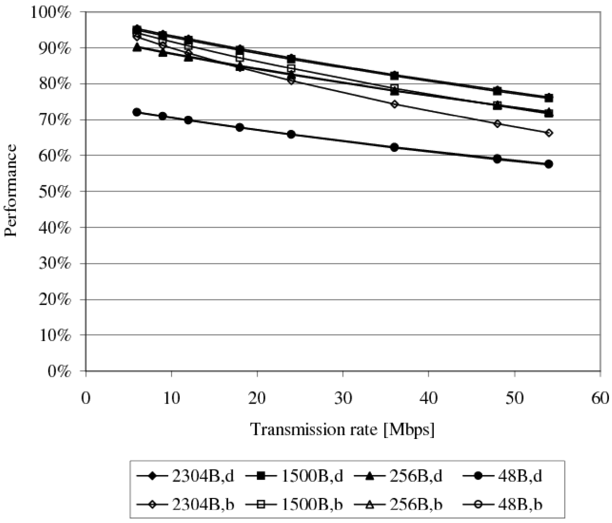

Fig. 9. 802.11 protocol performance for OFDM physical layer using A-MSDU aggregation with filling (d) and without filling (b) for $L_{\max} = 3839$ B

Similar results, but performed for $L_{\max} = 7935$ bytes, are presented in Fig. 10. In this case, differences between both variants are practically invisible and do not exceed 1%. Bearing in mind the results for both limits of the aggregated frame, one can say, that filling mechanism, except few cases, does not bring expected advantage. Taking into consideration possible difficulties in its practical implementation, its application may be regarded as unreasonable.

Calculation results for HT physical layer are presented in Fig. 11. These results were achieved for both aggregated frame length limits, i.e., $L_{\max} = 3839$ bytes and $L_{\max} = 7935$ bytes. As one can see on the graph, higher aggregated frame limit allows for higher protocol performance, however, for high transmission rates it is still relatively low. For example, for the highest transmission rate (600 Mbps), we get only 35% efficiency even if the longest frames (2304 data bytes) are used. Decrease of frame length has only a slight influence on performance – we can get 33% efficiency for 256-bytes frames and 28% for 48-bytes frames. For the lower aggregated frame length limit, the efficiency ranges from 16% to 22% at the highest transmission rate. A-MSDU aggregation is therefore worse than block acknowledge for longer frames, while it is better for shorter ones. As A-MSDU mechanism was introduced in order to increase protocol performance during transmission of short frames, we can say that this goal has been achieved. It should be however noted that block acknowledge allows for transmission, within a transmission cycle, of up to 64 frames of up to 2304 data bytes each, which gives the total of over 140 kbytes. At the highest transmission rate of 600 Mbps, transmission cycle lasts for about 4300 μs. A-MSDU aggregation allows for transmission of no more than 7900 bytes within a transmission cycle, which, at the highest rate of 600 Mbps, lasts for about 290 μs. Therefore, block acknowledge allows for sending of 18 times more data within 14 times longer time. Thus, we should not be surprised that block acknowledge performs better for longer frames. For shorter frames, its performance is limited by the maximum block size, while for longer frames, more influence results from aggregated frame length limit in A-MSDU aggregation.

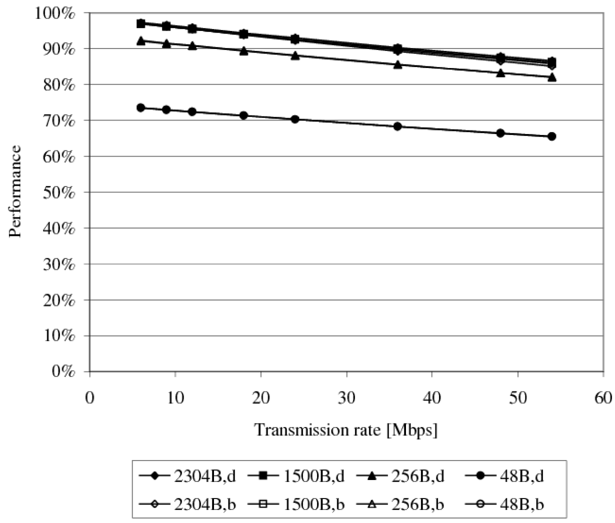

Fig. 10. 802.11 protocol performance for OFDM physical layer using A-MSDU aggregation with filling (d) and without filling (b) for $L_{\max} = 7935$ B

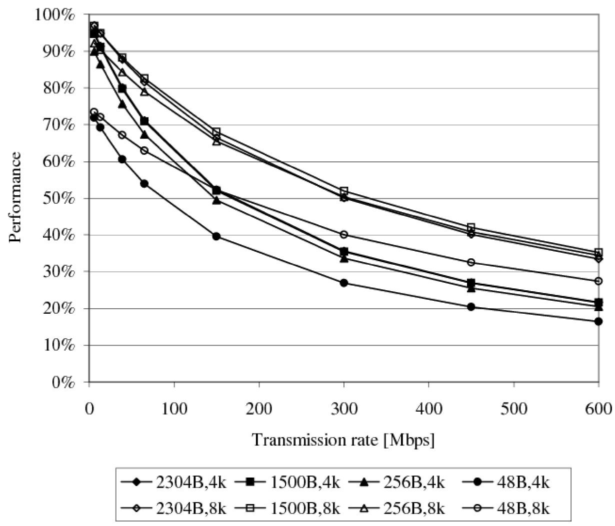

Fig. 11. 802.11 protocol performance for HT physical layer using A-MSDU aggregation without filling for $L_{\max} = 3839$ B (4k) and $L_{\max} = 7935$ B (8k)

3.4. A-MPDU aggregation. Calculation results for A-MPDU aggregation and OFDM physical layer are presented in

Fig. 12. For comparison, performance without respect to aggregate frame length limit was also calculated. As we can see on the graph, application of A-MPDU aggregation allows obtain performance of about 95 to 97% for every transmission rates, if only long frames are used. The difference between 1500-bytes and 2304-bytes frames is very small and invisible on the graph. In case of 256-bytes frames, the efficiency is also high – above 80% – and its decrease with increasing transmission rate is small. For the shortest frames, protocol performance ranges from 45 to 55%, depending on transmission rate.

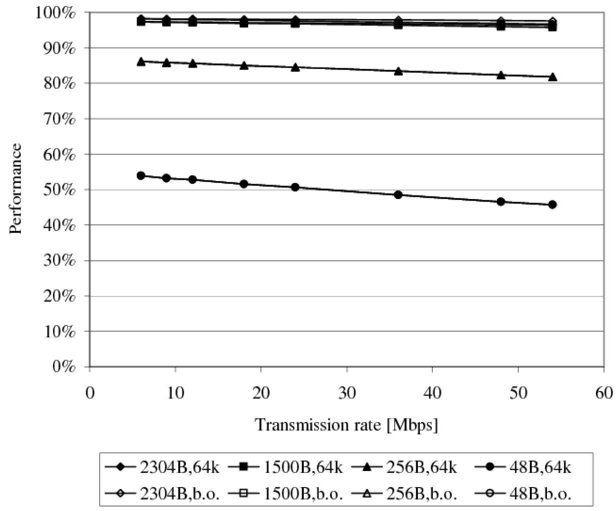

Fig. 12. 802.11 protocol performance for OFDM physical layer using A-MPDU aggregation with frame length limit (64k) and without limit (b.o.)

Removal of aggregated frame length limit brings some advantages in case of longer frames. Such behaviour is as expected, because it allows transmit a maximum number of frames (64) of any size allowed by the frame format (i.e., no more than 2304 data bytes). When the limit is set, the number of transmitted frames depends on their length. This effect takes place, when frame size exceeds about 1 Kbyte. With increasing frame length, number of aggregated frames decreases.

Calculation results for A-MPDU aggregation and HT physical layer are presented in Fig. 13. For comparison, performance without respect to aggregate frame length limit was also calculated. As we can see on the graph, application of A-MPDU aggregation allows for protocol efficiency over 80%, however, its degradation with increasing transmission rate is visible. For example, for longer frames, i.e., containing 1500 or 2304 data bytes, efficiency drops from about 96% for 6 Mbps to about 80% for 600 Mbps. Removal of aggregated frame length limit allows achieve efficiency of about 85% for 1500-bytes frames and 90% for 2304-bytes ones. In the case of shorter fames, performance is lower and it does not depend on the presence of aggregated frame length limit. For 256-bytes frames, performance degrades from about 86% for 6 Mbps transmission rate to about 50% for 600 Mbps. Similar values for the shortest frames are 55% and 16%, respectively.

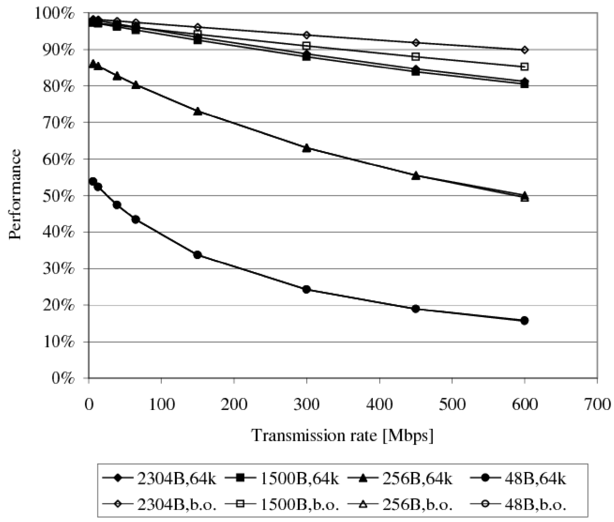

Fig. 13. 802.11 protocol performance for HT physical layer using A-MPDU aggregation with frame length limit (64k) and without limit (b.o.)

In the case of HT physical layer, removal of the aggregated frame length limit seems much more promising than for OFDM. We must however remember that when the limit is removed, the network adapter requires more than twice as much memory. For example, for transmission of 64 maximum-size frames, required buffer capacity is about 150 Kbytes. It seems that the cost resulting from limit removal is disproportionately high when compared to obtainable benefits. In some cases however, this cost may be reasonable.

4. Throughput upper limit

It has been proved [4], that throughput upper limit (TUL) exists for 802.11 networks with basic frame exchange. TUL is calculated, assuming also perfect operating conditions (no collisions or transmission errors) and infinitely high transmission rate. In this case, transmission time of all data link layer frames (such as Data, Ack and others) is zero. Thus, during calculation of transmission cycle duration, only PLCP overhead counts, namely, T_{SIFS}, T_{DIFS}, T_{prmbl} and T_{hdr}. Transmission cycle duration does not therefore depend on Data frame length, or – to be more precise – on data field capacity. However, TUL does depend on it, because it influences on number of data bytes transmitted within a cycle.

Calculated TUL values for OFDM physical layer (compatible with IEEE 802.11a and 802.11g standards) are collected in Table 2. Calculations were performed for data field capacity of 2304, 1500, 256 and 48 bytes.

Table 2
Throughput upper limit for OFDM physical layer [Mbps]

Frame length [B]	Basic	BlockAck	A-MSDU (4k)	A-MSDU (8k)	A-MPDU
2304	117.78	434.25	111.37	184.12	3119.12
1500	76.68	282.72	145.02	183.35	3093.61
256	13.09	48.25	173.24	174.07	791.98
48	2.45	9.05	136.89	139.26	148.50

The following data exchange methods were considered:

- basic (DCF),
- block acknowledge with block length set to $k = 64$ frames,
- A-MSDU aggregation with aggregated frame length limit set to $L_{\max} = 3839$ bytes (4k),
- A-MSDU aggregation with aggregated frame length limit set to $L_{\max} = 7935$ bytes (8k),
- A-MPDU aggregation with aggregated frame length limit set to 65535 bytes and block size limited to $k = 64$ frames.

As it is shown in Table 2, the basic data exchange method in 802.11 standard limits TUL to about 118 Mbps, but only when longest possible frames are used. Decrease of frame length to 1500 bytes causes TUL fall down to about 75 Mbps. It can be easily seen that this method does not ensure effective use of HT physical layer capabilities. Even OFDM physical layer, in some cases, cannot be effectively utilised.

When block acknowledge is used, TUL is almost 4 times as high as for basic method. Using longest frames, we can get TUL of 434 Mbps, while with 1500-bytes frames – about 283 Mpbs. We can therefore assume that block acknowledge allows effectively utilise OFDM layer capabilities, however, for HT layer it is not sufficient.

Unlike expected, A-MSDU aggregation for the longest frames not only does not bring advantages, but it can even make network achievements worse – when $L_{\max} = 3839$ bytes, TUL is even lower than for basic method. It is caused by larger overhead resulting from aggregation, but, despite aggregation, only a single maximum-size frame (2304 bytes) can be sent. However, for shorter frames, e.g., 1500-bytes long, TUL is twice as high as for basic method. When $L_{\max}$ increases to 7935 bytes, network performance is much better, but still below the capabilities of block acknowledge. Performance does not practically depend on payload size.

A-MSDU aggregation, however, shows high efficiency for shorter frames – TUL is about 135 to 140 Mbps for 48-bytes frames and 173 to 174 Mbps for 256-bytes ones. Similar results for both basic method and block acknowledge are much below these numbers. Thus, we can say that A-MSDU aggregation allows increase network efficiency while transmitting short frames. Nevertheless, it does not ensure effective usage of HT physical layer capabilities, despite they correspond to each other as both are defined in 802.11n standard.

A-MPDU aggregation shows the best performance of all considered transmission methods – TUL for 256-bytes frames exceeds 700 Mbps, while for the longest ones reaches over 3 Gbps. Even for the shortest frames this method is most effective. We can therefore say that A-MPDU aggregation allows effectively utilise transmission rates defined for HT physical layer. It could possibly allow effectively utilise future solutions with even higher transmission rates.

Calculated TUL values for HT physical layer (compatible with IEEE 802.11n standard) are collected in Table 3.

The results are slightly worse than those for OFDM, because physical layer preamble and header are longer than in OFDM. It increases protocol overhead and decreases its efficiency. Nevertheless, the relations between individual results are the same as for OFDM physical layer.

Table 3
Throughput upper limit for HT physical layer [Mbps]

Frame length [B]	Basic	BlockAck	A-MSDU (4k)	A-MSDU (8k)	A-MPDU
2304	106.85	363.58	101.55	167.89	2844.16
1500	69.57	236.71	132.23	167.18	2820.89
256	11.87	40.40	157.97	158.72	722.16
48	2.23	7.57	124.83	126.99	135.40

5. Summary and conclusions

Presented results show that the modifications of data link layer in IEEE 802.11 standard were really necessary. In fact, without presented enhancements, the protocol efficiency would be lower and lower for each new physical layer. Thus, effective throughput would not rise as high as expected, because it would still be limited to a relatively low value that would make physical layer utilisation ineffective, and its deployment – useless. With the new frame exchange methods, protocol performance is much better, and the effective throughput observed by a used is also higher.

It must be noticed, however, that the presented calculations are done for perfect conditions that are far from the real network operating conditions. Nevertheless, such conditions are not impossible, e.g., in a small home network. It would be, however, interesting how the frame exchange procedures that have been analysed in the paper behave in a real network. Such results can be achieved using computer simulations, but still more accurate results could be obtained in an experimental network. However, the results may depend on network hardware and software used for tests. For example, today's access points and network adapters compatible with 802.11n draft standard typically do not allow reach transmission rates higher than 300 Mbps, which is only a half of what is defined in the standard. Another fact is that protocol parameters like block size, number of aggregated frames and so on might be limited by some network equipment. Therefore, presented results should only be considered as a reference point for further research that takes into account the influence of real network environment and network devices properties. Nevertheless, they still show how much depends on the protocol architecture and the cross-layer optimization of the network layers.

REFERENCES

[1] IEEE Std 802.11™-2007, *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, New York, 2007.

[2] IEEE Std 802.15.3™-2003: *IEEE Standard for Information Technology – Telecommunications and Information Exchange*

between Systems – Local and Metropolitan Area Networks – Specific Requirements. Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs), IEEE, New York, 2003.

[3] *Specification of the Bluetooth System. Core Version 3.0 + HS*, Bluetooth SIG, 2009.

[4] Y. Xiao and J. Rosdahl, "Throughput and delay limits of IEEE 802.11", *IEEE Commun. Lett.* 6 (8), 355–357 (2002).

[5] D. Qiao, S. Choi, and K.G. Shin, "Goodput analysis and link adaptation for IEEE 802.11a wireless LANs", *IEEE Trans. Mobile Comput.* 1 (4), 278–292 (2002).

[6] IEEE Std 802.11n^{TM}-2009, *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. Amendment 5: Enhancements for Higher Throughput*, IEEE, New York, 2009.

[7] W.A. Beech, D.E. Nielsen, and J. Taylor, *AX.25 Link Access Protocol for Amateur Packet Radio*, Tucson Amateur Packet Radio Corporation, Richardson, TX, 1998.

[8] A.S. Tanenbaum, *Computer Networks*, Prentice Hall, Upper Saddle River, 2003.