

## Reverse Direction Transmission in Wireless Networks: Review

*Ali Ahmad Milad, Zul Azri Bin Muhamad Noh,  
Abdul Samad Shibghatullah and Mustafa Almahdi Algaet*

Department of Computer System and Communication,  
Faculty of Information and Communication Technology,  
Universiti Teknikal Malaysia Melaka, Hangtuha Jaya,  
76100 Durian Tunggal, Melaka, Malaysia

---

**Abstract:** Reverse direction mechanism is a promising significant development that may lead to promoting the accuracy of TXOP. The transfer, in conventional TXOP operation, is one way direction out of the station which holds the TXOP and which is not applied to some network services using two lane traffic namely VoIP and on-line gaming. Therefore, the conventional TXOP operation enhances only the forward direction transfer, but not the reverse direction transfer. Moreover, reverse direction mechanism makes it possible for the holder of TXOP to reserve unused TXOP time for its receivers which may improve the channel utilization as well as the performance of reverse direction traffic flows. It is well-known that the reverse direction transfer scheme aims mainly to improve the effectiveness and that plays a key role in reducing the overhead and increasing the system throughput. Thus, this paper provides an overview of a research progress in reverse direction transmission scheme over high speed wireless LANs. Moreover, it addresses the reverse direction mechanism that has been proposed for the next generation wireless networks and the ones adopted by IEEE 802.11n standard. Furthermore, it stresses the reverse issues that require to be dealt with in order to bring further progress to the reverse direction transmission.

**Key words:** Reverse direction • Wireless LANs • IEEE802.11 • TXOP

---

### INTRODUCTION

This section of the paper introduces informational background of the IEEE802.11 throughput and delay limits and it provides a detailed discussion about the improvements that have been used over high speed wireless LANs.

**IEEE802.11 Delay and Throughput Limits:** Wireless LANs based on IEEE 802.11 increases everywhere to support many applications using TCP, UDP, HDTV and VOIP. In IEEE802.11a the data rate reaches 54Mbps, IEEE802.11b reaches 11Mbps and the physical data rate in IEEE802.11n may reaches up to 600Mbps to get high efficiency at MAC layer [1-5]. The idea behind this is that the increase in the physical rate can result into increasing the transmission at MAC link and increasing the

throughput, thus, causing an increase in the overhead. Figure 1 describes the relation between the increasing data rate and the overhead at the MAC layer [6].

To achieve a wide performance enhancement or improvement, it is important to find out the major problem which causes MAC inefficiency as displayed in Figure 2 [7]. The theoretic throughput higher limit and the theoretic delay lower limit are present in the IEEE 802.11 protocols. The presence of such limitations assures the augmentation of the data rate without diminishing the overhead [8].

**Enhancements the Performance of MAC Layer:** During transmission, the frame in Distributed Coordination Function (DCF) scheme provides some overheads that can be clarified as: Distributed Inter-Frame Space (DIFS), backoff timing, PHY header and Short Inter-Frame Space

---

**Corresponding Author:** Ali Ahmad Milad, Department of Computer System and Communication,  
Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka,  
Hangtuha Jaya, 76100 Durian Tunggal, Melaka, Malaysia.

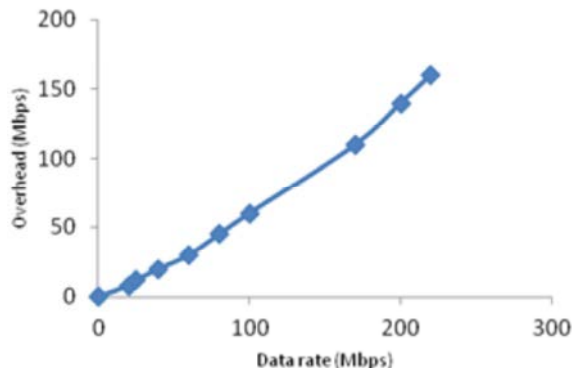


Fig.1: Relation Between Data Rate and Overhead

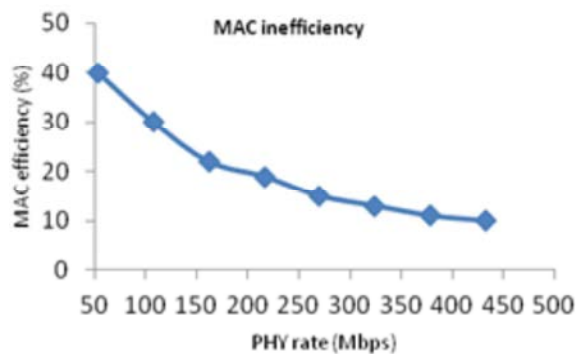


Fig.2: Relation Between Data Rate and MAC Efficiency

(SIFS). All of these types of overheads minimize the time and this can be an influential factor at performance of MAC layer [9]. So, to get high and good performance at MAC layer, it is necessary to mitigate the overheads. Many researchers have introduced the enhancements of MAC layer. In IEEE 802.11e, the data frames are sent in sequential separate ways, adding to that, these data frames were acknowledged by block acknowledgment (BA) after sending block acknowledgment request from the sender (BAR) [10, 11]. Besides, IEEE 802.11n (next generation wireless LANs) provides high throughput mechanism and may reach more than 100 Mbps [5]. There are two possible methods that can be achieved to enhance the performance of next generation wireless LANs: the first one aims to increase the physical data rate in the PHY layer and the second one aims to increase the efficiency of the MAC layer [12].

**Enhancements of IEEE 802.11n:** The IEEE 802.11 Working Group adopts the IEEE 802.11n, which is a modified version of higher throughput and higher speed improvement. The main goal of IEEE 802.11n is to provide a higher throughput rather than higher data rates along with PHY and improvements of medium access control

(MAC) [13]. Moreover, IEEE 802.11n provides many enhancements to reduce the overhead at MAC layer, such as: frame aggregation, Block acknowledgment (BA) and reverse direction transmission (piggybacking) [14]. The Improvements of next generation wireless LANs IEEE802.11n will be discussed in details especially in more concern with the reverse direction transmission.

**Frame Aggregation:** Frame aggregation is defined as a function which allows combining several frames into a single large frame for transfer. It is carried out by using two available methods: aggregate MAC protocol service unit (A-MSDU) and aggregate MAC protocol data unit (A-MPDU). Moreover, an MSDU is mainly distinguished from an MPDU in that whereas the former is correspondent to the information that is imported to or exported from the upper part of the MAC sub layer, respectively, from or to the higher layers, the latter depends on the information transferred by the lower part of the MAC from or to the PHY. Aggregate exchange sequences are facilitated by a protocol acknowledging multiple MPDUs with a single block ACK. This is made as a response to a block acknowledgment request (BAR). Another key to achieving such improvement is that the 802.11n specifies is the bidirectional data transfer method over a single TXOP. This is called a reverse direction. This characteristic allows the transfer of data frames including aggregates in both directions in one TXOP. Currently, allocating the sender STA with a TXOP may inform surrounding STAs about the duration of the wireless medium engagement. The results this approximation of channel provides, however, cannot always be accurate and the transfer failure is often sooner. In other cases, contended STAs, therefore, assume that the channel is still occupied. Moreover, with reverse direction, any available packets addressed to the sender for the remaining TXOP time are allowed to be sent by the initial receiver STA. This, in particular, suits TCP well as it permits a TCP link to piggyback TCP ACK collection onto TCP data transfer. The long network allocation vector (long-NAV) is another progress that develops scheduling as long as the station that includes a TXOP may set up a longer NAV value for the intention to save multiple PPDU. Another compulsory characteristic is phased coexistence operation (PCO) which mainly aims to protect stations either through 20 MHz or 40 MHz channel spectrum at the same time. Lastly, the reduced IFS (RIFS) is suggested to permit a time interval of 2  $\mu$ s between multiple PPDU, which is much shorter than SIFS as identified in the legacy standards.

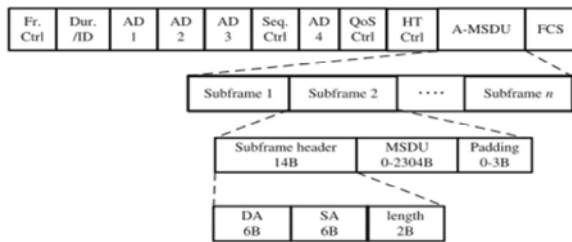


Fig. 3: Shows the frame of A-MSDU

**A-MSDU Aggregation:** The A-MSDU (or MSDU aggregation) objective is mainly to permit sending multiple MSDUs to the same receiver concatenated in a single MPDU. Absolutely, this leads to enhancing the effectiveness of the MAC layer, mainly when many small MSDUs, such as TCP acknowledgments are included. This function of enhancement for A-MSDU within the 802.11n is proved to be compulsory at the receiver's side. To form an A-MSDU, a layer at the top of the MAC receives and buffers multiple packets (MSDUs). The A-MSDU is fulfilled in one of these two cases: having the size of the waiting packets reach the maximal A-MSDU threshold or in the second case when the maximal delay of the oldest packet reaches a pre-assigned value. Moreover, its maximum length can be either 3839 or 7935 bytes; this is 256 bytes shorter than the maximum PHY PSDU length (4095 or 8191 bytes, respectively), as the expected space is preserved for future status or controlling information. The size can be noticed in the high throughput (HT) capabilities element which is advertised from an HT STA in order to affirm its HT status. Although the maximal delay can be set to an independent value for every AC, it is usually set to 1  $\mu$ s for all ACs. When constructing an A-MSDU, certain limitations can be detected:

- The same TID value must be acquired by all MSDUs.
- Lifetime of the A-MSDU should be appropriate to the maximum lifetime of its constituent elements.
- The destination address (DA) and sender address (SA) parameter values in the sub frame header must correspond to the same receiver address (RA) and transmitter address (TA) in the MAC header.

Broadcasting or multicasting, therefore, is prevented. However, the main weakness of employing A-MSDU is represented by the under error-prone channels. Furthermore, the entire A-MSDU must be retransferred, by combining all MSDUs into a single MPDU with a single sequence number, for any sub frames that are

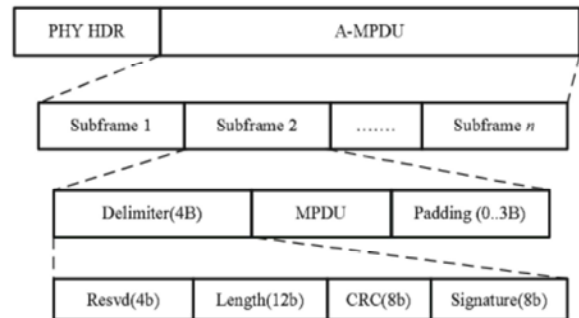


Fig. 4: Shows the frame of A-MPDU

corrupted. This case is dealt with in [15], where further frame structures or optimum frame sizes have been suggested to achieve accuracy under noisy channels.

**A-MPDU Aggregation:** The major aim of the concept of A-MPDU aggregation is to connect multiple MPDU sub frames with a single leading PHY header. The only aspect in which it differs from A-MSDU aggregation is the function of A-MPDU since it commences after the MAC header encapsulation process. Because of this, the A-MSDU restriction of aggregating frames with matching TIDs is not considered as a factor with A-MPDUs. However, it is necessary to deal with all the MPDUs within an A-MPDU using the same receiver address. Besides, the lack of waiting/holding time needed in forming an A-MPDU makes the number of MPDUs which should be added totally dependent on the number of packets which already exist in the transmission queue. Furthermore, the maximum length obtained by an A-MPDU (the maximum length of the PSDU probably to be received) is 65,535 bytes. However, it is possible to be further limited depending on the capabilities of the STA existing in the HT capabilities element. Moreover, the maximum number of sub frames that it may acquire is 64 and this is due to the reason that a block ACK bitmap field is 128 bytes in length, where mapping each frame is performed using two bytes. It is also important to notice that these two bytes are needed for acknowledging up to 16 fragments. However, since the A-MPDUs prevent fragmentation, these additional bits become excessive. Therefore, a new variant has been carried out, known as compressed block ACK with a bitmap field of eight bytes long. Lastly, the size of each subframe is confined to 4095 bytes as the length of a PPDU cannot surpass the 5.46-ms time limit; this can spring out of the maximum length divided by the lowest PHY rate, which is 6 Mb/s and is the highest duration of an MPDU in 802.11a [16].

**Block Acknowledgment:** A Block ACK method is proposed in the IEEE802.11e specification [20]; in (BTA) block transmission and acknowledgment scheme and therefore a block of data frames send it to the destination in sequence and between each one and the data frame there is a SIFS period of time. After sending the block of data, the sender starts sending the (BAR) block acknowledgment request to ask about frame which has been received by the receiver. Then, the (BA) block acknowledgment is sent back to the sender including information about the frame received as illustrated in Figure 6. When the block acknowledgement is received correctly, the sender is supposed to delay the DIFS and back off mechanisms earlier than sensing the channel and all other stations should wait until the block acknowledgment transmission finishes. If many stations have started transmitting the data with BAR at the same time, the collisions will occur. Moreover, when each station sends the data with BAR, the receiver should wait to detect the collision first and then send the block acknowledgment.

**Reverse Direction:** Two types of stations are identified in reverse direction operation: RD initiator and RD responder. RD initiator which is the station holding the TXOP that permits sending the Reverse Direction Grant (RDG) to the RD responder. Furthermore, RDG is marked in the 802.11n header and sent together with the data frame to the RD responder. When receiving the data frame with RDG, the RD responder responds with RDG acknowledgement, mainly if it has data that needs to be sent or without RDG when no data is available to be sent to the RD initiator. Then, the RD initiator needs to wait for the transfer from the RD responder [17-19], when the acknowledgement is marked with RDG. Figure 5.

**Reverse Direction Transmission Approaches:** The 802.11n RD protocol mainly aims to accurately exchange the data between two 802.11 devices during a TXOP by restricting the necessity for either device to have access to a further data exchange. Before the RED protocol, each uni-directional data transfer needs the initiating station to attract (and possibly reserve time on) a contention-based RF medium. With RD, the other stations are essentially allowed to send information back once the transmitting station has attained a TXOP. Therefore, two roles need be identified: RD initiator and RD responder. The RD initiator sends its permission to the RD responder due to

a Reverse Direction Grant (RDG) in the RDG/More PLCP protocol data unit (PPDU) field of the HT Control field in the MAC frame as illustrated in Figure 7. This bit is adopted by the RD initiator to grant access (RDG) to the RD responder and by the RD responder to prove if it can send more frames immediately after the one it has just received (More PPDU)[21].

**Reverse Direction (RD) Exchange Sequence:  
An RD Exchange Sequence Includes:**

- The transfer of the PPDU by the TXOP holder including an RD grant (the RDG PPDU) as indicated by the PPDU consists of one or more high throughput control (+HTC) MPDUs in which the RDG/More PPDU subfield is put into 1. The STA transmitting this PPDU is called the RD initiator. The process of carrying out the rules for an RD initiator is performed only during a single RD exchange sequence (e.g. after the transfer of an RDG PPDU till the end of the last PPDU in the RD exchange sequence).
- The transfer of one or more PPDUs (the RD response burst) by the STA is dealt with in the MPDUs of the RDG PPDU. The first (or only) PPDU of the RD response burst includes at most one immediate Block ACK or ACK response frame this process is called piggybacking. The last (or only) PPDU of the RD response burst includes any MPDUs requiring an immediate Block ACK or ACK response. The RD responder is the STA that transfers the RD response burst. Furthermore, It is only during a single RD exchange sequence that the rules for an RD responder can be applied, i.e. following the reception of an RDG PPDU and up to the transfer of the PPDU by the RD responder in which the RDG/More PPDU subfield is put into 0.
- The transfer of the PPDU by the RD initiator containing an immediate Block ACK or ACK MPDU (the RD initiator final PPDU), If so needed by the last PPDU of the RD response burst.

**NOTE:** The RD initiator may contain multiple RD exchange sequences within a single TXOP. Each RD exchange sequence within a single TXOP that can be dealt with by different recipient and it is probable that any single recipient includes more than one RDG within a single TXOP.

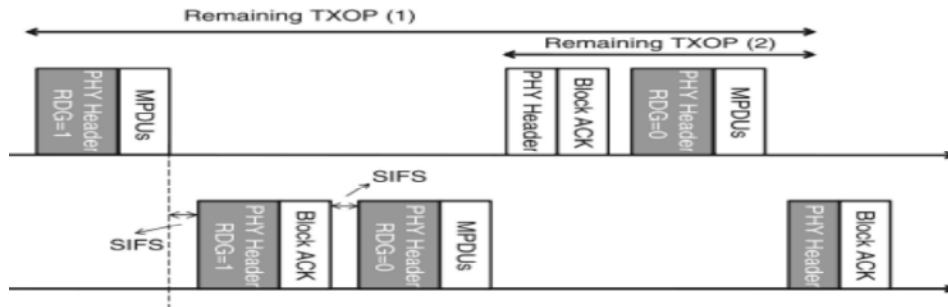


Fig. 5: Reverse direction

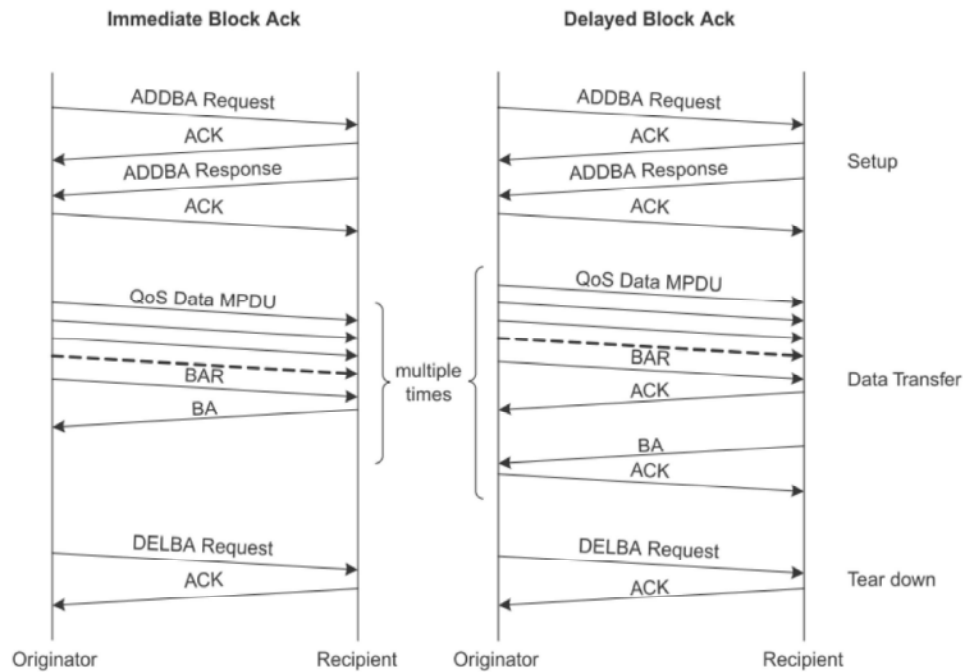


Fig. 6: Immediate and delayed block acknowledgment

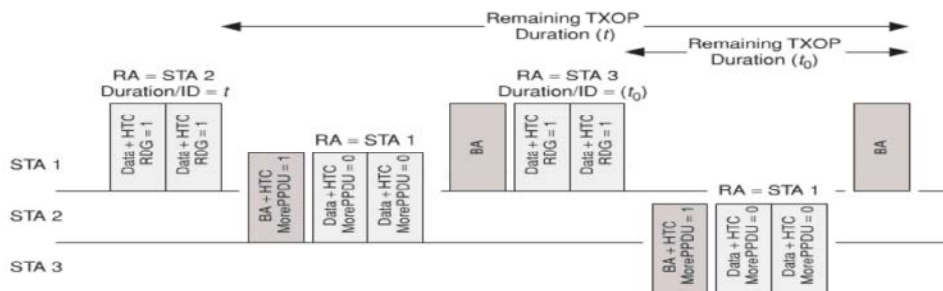


Fig. 7: Reverse direction exchange

**Supports for RD:** Support of the RD characteristic is optional for an HT STA in the sense that an RDG is never demanded to be provided by a TXOP holder and the grant is never required to be used by a STA receiving an RDG. Support of the RD characteristic such as an RD responder is identified due to the RD Responder subfield

of the HT Extended Capabilities field of the HT Capabilities element. The STA will put the RD Responder subfield into 1 in frames so that it is transmitted with a value of HT Capabilities element if dot11RD Responder Option Implemented is true. Otherwise, the STA will put the RD Responder subfield into 0 [22].

**Rules for RD Initiator:** An RDG can exist provided that the MPDU which carries the grant or even each MPDU carrying the grant in an A-MPDU does not correspond to one of these conditions listed below:

- The QoS data MPDU with the ACK Policy field is carried out at any value except power save multi poll (PSMP) ACK (i.e. including Implicit Block ACK Request), or
- The Block ACK Request is relevant to the HT-immediate Block ACK agreement, or
- The MPDU does not require straight response (e.g. Block ACK under an HT-immediate Block ACK agreement, or Action No ACK).

Within a PSMP sequence, an RDG will not be dealt with.

**NOTE 1:** These regulations are meant to guarantee that an RDG is conveyed in the PPDU that requires neither an immediate response nor an immediate Block ACK or ACK response.

**NOTE 2:** It is not necessary that the RD initiator ought to treat the RD Responder field of a potential responder before taking the decision to send the PPDU to that STA in which the RDG/More PPDU subfield is put into 1.

**NOTE 3:** The RD initiator is needed to identify the +HTC Support field of a potential responder before taking the decision of sending the PPDU to that STA in which the RDG/More PPDU subfield is put into 1, Table 1.

Transfer of a +HTC frame by an RD initiator with the RDG/More PPDU subfield put into 1 (either transferred as a non-A-MPDU frame or within an A-MPDU) proves that the duration mentioned by the Duration/ID field is valid for the RD response burst and RD initiator final PPDU (if present).

The RD initiator that sets the RDG/More PPDU field put into 1 in a +HTC frame will have the access category (AC) constraint subfield to 1 in that frame if the TXOP is accomplished due to the EDCA channel access mechanism and will otherwise put it into 0, as illustrated in Table 2.

The RD initiator will not exchange a +HTC frame with the RDG/More PPDU subfield put into 1 that needs a response MPDU except these below:

- ACK
- Compressed Block ACK

Table 1: RDG/More PPDU subfield values

Value	Role of transmitting STA	Interpretation of value
0	RD initiator	No reverse grant
	RD responder	The PPDU carrying the frame is the last transmission by the RD responder
1	RD initiator	An RDG is present, as defined by the duration/ID field
	RD responder	The PPDU carrying the frames is followed by another PPDU

Table 2: AC Constraint subfield values

Value	Description
0	The response to a reverse direction grant (RDG) may contain data frames from any TID.
1	The response to an RDG may contain data frames only from the same AC as the last data frame received from the RD initiator.

Subject to TXOP restrictions, after transferring an RDG PPDU, the RD initiator can exchange its next PPDU as follows:

**Normal Continuation:** The RD initiator can transfer its next PPDU a minimum of a SIFS after getting a response PPDU that satisfies one of the above conditions:

- It should include one or more accurately received +HTC frames with the RDG/More PPDU subfield put into 0, or
- It should include one or more accurately received frames that are able to carry the HT Control field but they should not consist of an HT Control field, or
- It should include a correctly received frame that needs a straight response.

**Error Recovery:** When the carrier sense (CS) mechanism proves that the medium is idle at the TxPIFS slot boundary, the RD initiator can transfer its next PPDU (this transfer is a process of the current TXOP).

**NOTE 1:** The RD initiator is responsible for recovering Error of the RDG mechanism

**NOTE 2:** If the response is corrupted after exchanging a PPDU including an RDG, therefore the state of the RDG/More PPDU subfield is unknown and the RD initiator of the RD transmission is prevented to exchange after a SIFS interval. Transfer may happen at a PIFS interval after reassertion of (CS).

**NOTE 3:** After transferring the PPDU needing a response but not including an RDG, the state of the RDG/More PPDU subfield, but the response does not impact the behaviour of the RD initiator.

The STA that exchanges a QoS +CF-ACK data frame can also contain an RDG in that frame in condition that:

- It is a non-A-MPDU frame and
- The target of the +CF-ACK is equal to the Address 1 field of the frame.

**NOTE:** The RD initiator may exchange a CF-End frame following to the rules for TXOP truncation and according RD transfer sequences. An RD responder never exchanges a CF-End.

**Rules for RD Responder:** After the reception of the RDG PPDU, an RD responder will transfer the initial PPDU of the RD response burst a SIFS. PPDUs in a response burst are separated by SIFS or RIFS.

**NOTE:** The transfer of a response by the RD responder does not provide a new channel access but a progress of the RD initiator's TXOP. When responding to an RDG, an RD responder forgets about the NAV.

#### The Recipient of an RDG Can Reduce the RDG By:

- Denying exchanging any frames coming after the RDG PPDU, when no response is otherwise needed, or
- Exchanging a control response frame with the RDG/More PPDU subfield put into 0, or
- Exchanging a control response frame that includes no HT Control field.

An RD responder may transmit a +CF-ACK non-A-MPDU frame in response to a non-A-MPDU QoS Data +HTC MPDU that has the ACK Policy field set to Normal ACK and the RDG/More PPDU subfield set to 1.

The RD responder will make sure that its PPDU transfer(s) and any expected responses fit completely within the remaining TXOP duration, as revealed in the Duration/ID field of MPDUs within the RDG PPDU.

An RD responder will not transfer an MPDU (either individually or aggregated within an A-MPDU), except one of these mentioned below:

- ACK
- Compressed Block ACK
- Compressed Block ACK Request (BAR)
- QoS data
- Management

The RD responder shall transfer data frames of only the same AC, as the last frame is received from the RD initiator when the AC Constraint subfield is put into 1. Moreover, for a Block ACK Request or Block ACK frame, the AC is identified when treating the TID field. For a management frame, the AC is AC\_VO. The RD initiator will not be able to exchange a +HTC MPDU with the RDG/More PPDU subfield put into 1 from which the AC is difficult to be figured out. Furthermore, the RD responder can exchange data frames of any TID, if the AC Constraint subfield is put into 0.

During an RDG, the RD responder will not be able to transfer any frames with an Address 1 field that is not comparable to the MAC address of the RD initiator.

When an RDG PPDU also requires an immediate Block ACK response, the Block ACK response frame will be included in the first PPDU of the response and this process is called piggybacking.

An HT Control field treating the RDG/More PPDU subfield put into 1 will exist in every MPDU within the PPDU and be able to deal with the HT Control field when a PPDU is not the final PPDU of a response burst.

The last PPDU of a response burst will contain the RDG/More PPDU subfield put into 0 in all +HTC MPDUs existed in that PPDU.

The RD responder will not put the RDG/More PPDU subfield into 1 in any MPDU in a PPDU that has an MPDU that needs a straight response.

**NOTE:** The RD responder has to wait for either another RDG or its own TXOP before retrying the transfer, if the RD responder exchanges a PPDU that estimates a transfer by the RD initiator after SIFS and such transfer is unnoticed.

The RD responder will not be able to transfer any more PPDUs within the current response burst, after exchanging the PPDU containing one or more +HTC MPDUs in which the RDG/More PPDU subfield is put into 0.

**NOTE:** When the RD-capable STA that is not the TXOP holder receives the PPDU that does not have an indication of the RDG, no difference can be noticed in its response in comparison with the STA that is not RD-capable.

In [23] a new scheme of simple reverse direction data transfer that is capable of achieving sufficient reduction of the bottleneck overhead and permission of the data

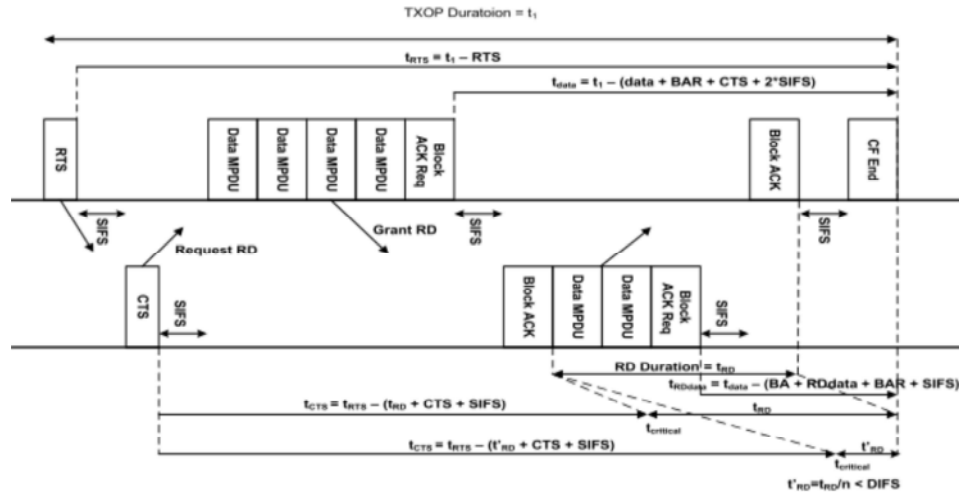


Fig. 8: Data service in RD protocol

packets flow in both directions was suggested. The Initiator sends the RTS which contains NAV duration of TXOP. According to the receipt of RTS, the responder checks whether it has any packet to send to the initiator. And when it has, it determines the duration required for the reverse direction data transfer. Under the normal conditions, the CTS from responder must include the duration in the RTS minus, the sum of CTS and SIFS duration. Regarding the reverse direction, the responder also subtracts the value of reverse data duration from the received duration in the RTS and sends the CTS with modified duration to the initiator. The initiator checks the duration in the CTS to insure the equality of the difference existing between the duration in the RTS and the CTS in one hand and the CTS and SIFS duration. However, when it is not equal, then the initiator can realise that responder has packet to exchange. Furthermore, as displayed in Fig. 8 below, the initiator can provide use to the responder by the piggybacking data MPDU or BA Request (BAR) MPDU, when this duration is less than the remaining TXOP.

In [24], the authors identified the value of reverse direction data flow characteristic of the emerging IEEE 802.11n WLAN standard and studied the benefit of using it in terms of its performance and clarified the way this merit springs out. The authors also deduced that RDF diminishes the number of contentions by a factor of 1.5 to 2 and thus decreases the number of collisions within WLAN during EDCA phase. Besides, it diminishes the MAC overhead associated with contention by itself.

**The RD Protocol Exchange is Summarized as Follows:**

- The RD initiator (the TXOP holder) sends the PPDU which includes the RDG and that needs an immediate response.
- The RD responder will respond with one or more PPDUs. The first or only PPDU will include one or more ACKs or Block ACK frames (piggyback).
- The last or only PPDU exchanged by the RD responder in the burst will include any packets that need a straight ACK or Block ACK response. The last PPDU will have the RDP/More PPDU subfield.

**Bidirectional Transmission:**

**Piggyback Scheme:** When the data packets are transferred in duplex directions from part A to part B and the data arrives at B, instead of sending a control frame from B to A, receiver B waits until the network layer at B sends the next packet to A and the acknowledgment is attached in the data frame from B to A using the field of acknowledgment in the data frame header. Therefore, the acknowledgment gets a free ride in the data frame and this technique is known as piggybacking. One of the most well-known advantages of piggyback scheme is improving the efficiency, which plays the role in reducing the overhead and increasing the system throughput.

**Overview of Piggyback:** When the receiver station has a frame to the sender station and allows sending the data frame with ACK to the sender, this process is called a piggyback scheme. So the acknowledgment gets a free



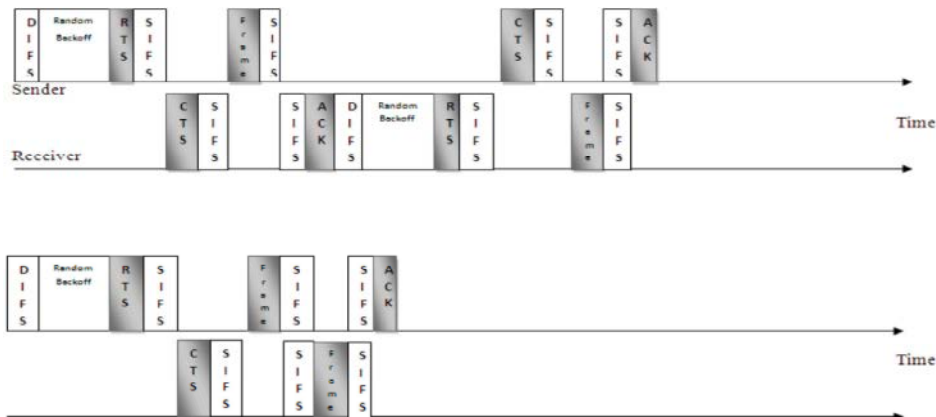


Fig. 9: The overhead with/without the piggyback scheme

ride in the data frame and takes few bits and this is a distinct ACK. Therefore, each frame requires an ACK header and data frame (checksum). This means that the piggyback merely relies on the receiver since the fewer the frames are sent, the fewer the frames that arrive and this is dependent the way the receiver is organized. However, the emergence of the piggyback scheme has posed some cases of complexity. For example, the question of how long the data link layer is supposed to wait until the packet is transmitted to piggyback the ACK is still a posing a complex case. As far as we know, the link layer usually waits for a certain period of time and if this waiting period consumed by the data link layer is longer than the sender timeout period, it is expected that the frame will be retransmitted. Therefore, it is assumed that the data link layer must wait for fixed time like Ad hoc scheme such as number of milliseconds. On the other hand, one of the benefits of the piggyback is that the piggyback frame does not need to rivalry the channel again in a case when the receiver has a frame to send to the sender. This is because it does not need to be in the front of the queue but the nearest frame to the destination at the sender. In [25] the researcher shows the overhead with/without piggyback mechanism in case when the frame belong to the receiver is sent to the sender after receiving a frame as completion of the channel. After this a CTS frame time, an RTS frame time, two SIFS times, a DIFS time and a random back off is required to be done by it. Otherwise, in a case when the frame is possible to be piggybacked by the receiver to the sender along with the acknowledgment, the ACK is sent by the sender as a way of acknowledging the piggybacked frame after reducing the SIFS time and the overhead has been already accomplishe (Figure 9).

In IEEE 802.11 WLAN data frame can increase the channel efficiency in a wireless. But the piggyback will decrease the channel efficiency for frame transmission delay. In Global control information the stations have a low transmission rate and in control frames. So the piggyback deals with the problem concerning the low physical transmission rate and evaluation of the effect of this problem with respect to the average frame transmission delay and the channel utilization. Therefore, the purpose of proposing the delay-based piggyback scheme by the authors was to mitigate the piggyback problem [26]. And as revealed by the same authors, the piggyback led to decreasing the channel efficiency while increasing the transmission delay for one station with low physical transmission rate were present.

[27] investigated the piggyback scheme when the data frame was transmitted in two cases for the station. If the frame was corrupted, so whole the process would restart, or the data frame would be received successfully. The researcher also showed that the wireless station turn on piggyback scheme when the packet is less than 1100bytes. Moreover, the throughput is very low because the overhead is high and the data rate is better than the piggyback because the time missed in retransmitting the “data+Acknowledgment” frames is reduced.

[28], the performance of the piggyback requests is already evaluated, so the ratio of requests for the piggyback and the influence on the delay were insistences for different traffic. With the large number of users, the performance increased by using the piggyback requests and the web traffic model showed that large number of bandwidths can piggyback on previous packets.

[29] defined the piggyback problem that when the control frame is piggybacked the channel efficiency is decreased and at the same time, the station has a low physical rate. The researchers also showed that and evaluated the channel utilization with and without the piggyback frame and they proved that utilization of channel means the ratio of the total frame transmission time to the super frame length. So they solved the piggyback problem as the low physical rate by calculating the delay of the piggyback scheme and proposed the delay base piggyback scheme and found that the piggyback was decreasing the channel efficiency while increasing the frame transmission delay even in one station physical rate. However, when the physical transmission rate increases. In this method channel efficiency and the delay efficiency are also increased. Therefore, if the piggyback practices well, the channel efficiency and the delay transmission will increase.

But the proposed algorithm is expected to decrease the average frame transmission delay and the channel utilization is estimated to be about 24% and 25%, respectively if there is one station which has low a physical transmission rate.

[30] Showed that the piggyback mechanism reduces the delay for uplink and downlink packets and the packets loss probability for uplink traffic and downlink traffic for the case of backoff method and piggyback method. For downlink data, packets do not occur for piggyback method and the loss packets of downlink data packet for the backoff is not small. Moreover, the piggyback method reduces the energy consumption significantly and there are no losses of downlink data packets.

[31] Investigated QoS data frames and their related usage rules to increase the channel efficiency. A CF-boll used to grant the channel to QSTA and piggybacked in QoS the data frame to increase the channel efficiency. However, the channel efficiency may be decreased by CF-boll piggyback problem when QSTA associated in QBSS uses the low physical rate. The CF-boll piggyback scheme varies between 24 and 36Mbps depending on the traffic load.

[32] Proposed a study in which the impacts of channel access, bandwidth and piggyback scheme on the performance were examined and it was observed that the bandwidth utilization can be improved if the bandwidth for random channel access is properly configured according to the channel access parameters, piggyback scheme and the traffic of network. The requests for

piggyback can be used to improve the bandwidth efficiency, but it is possible to increase the delay of channel access. The bandwidth efficiency with piggyback requests can be saturated more quickly with the increased number of SSS than that without piggyback requests.

#### **Related Study (Types of ACKs):**

**Cumulative Acknowledgment:** This refers to the stance in which each acknowledgment confirms correct reception of all bytes up to the ACK number.

#### **Acknowledgment Only Segment and Piggybacking:**

Based on this type of acknowledgments, the indication of the ACK is figured out through an ACK field in the TCP header. Therefore, in acknowledging that the bytes have been received correctly, the receiver can also create or develop an ACK segment which carries only the header that contains the ACK number. Another way of acknowledging the correct received bytes is sending the ACK in data segment and when the ACK travels in this data segment, the process is usually known as piggybacking.

**Delayed Acknowledgment:** This feature indicates that the TCP receiver has two different options: either it generates the ACK and receives a segment or it delays / postpones the ACK for a short period of time. In the case of following the second option which is delaying the ACK, the receiver will be able to acknowledge two segments at one time and reduces the ACK traffic. However, in case of a long time-delay of the ACK can cause a timeout and retransmission at the sender. Therefore, the recommended span of time during which the receiver delays the ACK should not be more than 500ms.

**Duplicate ACK:** This feature points at the case in which the segment gets lost in traffic and the segment following it arrives safely at the receiver. It is possible for the receiver to receive the data in a sequent series or number just beyond the range expected to be received. In such a case, the receiver performs the processes of buffering the incoming bytes and re-generating the ACK for the bytes which have been received so far in a sequence. Thus, re-generating the same ACK number can lead ACK duplication phenomenon because the sender receives the same ACK several times. However, for the original TCP, ACK duplication is normally ignored by the sender [33].

## REFERENCES

1. Abichar, Z. and J. Chang, 2013. Group-based medium access control for IEEE 802.11 n wireless lans.
2. Akhmetov, D., 2006. 802.11 n: Performance results of reverse direction data flow. In: Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on. IEEE: pp: 1-3.
3. Anwar, S., O. Mohamed, S. Shamala and A.H. Nor, Frame aggregation in wireless networks: Techniques and issues. IETE Technical Review, pp: 28.
4. Committee, S., 2005. Wireless lan medium access control (mac) and physical layer (phy) specifications: Amendment 8: Medium access control (mac) quality of service enhancements. IEEE Computer Society.
5. Group, I.W., 2006. IEEE 802.11n/d1.0 draft amendment to standard for information technology-telecommunications and information exchange between systems-local and metropolitan networks-specific requirements-part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications: Enhancements for higher throughput. Local and Metropolitan Networks, Specific Requirements, Part, pp: 11.
6. Hassan, M. and R. Jain, 2004. High performance tcp/ip networking. Pearson Prentice Hall.
7. He, J., K. Yang, K. Guild and H.H. Chen, 2008. On bandwidth request mechanism with piggyback in fixed IEEE 802.16 networks. Wireless Communications, IEEE Transactions on, 7(12): 5238-5243.
8. Ketchum, J., S. Nanda, R. Walton, S. Howard, M. Wallace, B. Bjerke, I. Medvedev, S. Abraham, A. Meylan and S. Surineni, 2005. System description and operating principles for high throughput enhancements to 802.11. QUALCOMM Inc.
9. Lee, H.J. and J.H. Kim, 2006. A optimal cf-poll piggyback scheme in IEEE 802.11 e hcca. In: Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference. IEEE: pp: 6.
10. Lee, H.J., J.H. Kim and S.H. Cho, 2007. A delay-based piggyback scheme in IEEE 802.11. In: Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE. IEEE: pp: 447-451.
11. Lee, H.J., J.H. Kim and S. Cho, 2007. A novel piggyback selection scheme in IEEE 802.11 e hcca. In: Communications, 2007. ICC'07. IEEE International Conference on. IEEE: pp: 4529-4534.
12. Lee, T.H., Y.W. Kuo, Y.W. Huang and Y.H. Liu, 2010. To piggyback or not to piggyback acknowledgments? In: Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st. IEEE: pp: 1-5.
13. Li, T., Q. Ni, D. Malone, D. Leith, Y. Xiao and T. Turletti, 2009. Aggregation with fragment retransmission for very high-speed wlans. IEEE/ACM Transactions on Networking (TON), 17(2): 591-604.
14. Li, T., Q. Ni, T. Turletti and Y. Xiao, 2005. Performance analysis of the IEEE 802.11 e block ack scheme in a noisy channel. In: Broadband Networks, 2005. BroadNets 2005. 2nd International Conference on. IEEE: pp: 511-517.
15. Lim, W.S. and Y.J. Suh, 2010. Achieving per-station fairness in IEEE 802.11 wireless lans. In: World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a. IEEE: pp: 1-9.
16. Lin, Y. and V.W. Wong, 2006. Wsn01-1: Frame aggregation and optimal frame size adaptation for IEEE 802.11 n wlans. In: Global Telecommunications Conference, 2006. GLOBECOM'06. IEEE. IEEE: pp: 1-6.
17. Milad, A.A., Z.A.B.M. Noh, A.S. Shibghatullah, S. Sahib, R. Ahmad and M.A. Algaet, 2013. Transmission control protocol performance comparison using piggyback scheme in wlans. Journal of Computer Science, 9(8): 967.
18. Mujtaba, S.A., 2005. Tgn sync proposal technical specification. IEEE 802.11-04/0889r7.
19. Ozdemir, M., D. Gu, A.B. McDonald and J. Zhang, 2006. Enhancing mac performance with a reverse direction protocol for high-capacity wireless lans. In: Vehicular Technology Conference, 2006. VTC-2006 Fall. 2006 IEEE 64th. IEEE: pp: 1-5.
20. Park, J.S., T.O. Kim, K.J. Kim and B.D. Choi, 2009. Performance analysis of IEEE 802.15.4 non-beacon mode where downlink data packets are transmitted by piggyback method. In: Communications Workshops, 2009. ICC Workshops 2009. IEEE International Conference on. IEEE: pp: 1-6.
21. Pries, R., D. Staehle and D. Marsico, 2007. Performance evaluation of piggyback requests in IEEE 802.16. In: Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th. IEEE: pp: 1892-1896.

22. Rasheed, M.M., O. Ghazali and R. Budiarto, 2012. Fast detection of stealth and slow scanning worms in transmission control protocol. *Journal of Applied Sciences*, 12: 2156-2163.
23. Saeed, R.A. and S. Khatun, 2005. Design of microstrip antenna for wlan. *Journal of Applied Sciences*, 5(1): 47-51.
24. Saif, A., M. Othman, S. Subramaniam and N.A.W.A. Hamid, 2012. An enhanced a-msdu frame aggregation scheme for 802.11 n wireless networks. *Wireless Personal Communications*, 66(4): 683-706.
25. Singh, M. and B. Edwards, 2004. System description and operating principles for high throughput enhancements to 802.11. *IEEE 802.11-04-0886-00-000n*, Aug.
26. Skordoulis, D., Q. Ni, H.H. Chen, A.P. Stephens, C. Liu and A. Jamalipour, 2008. Ieee 802.11 n mac frame aggregation mechanisms for next-generation high-throughput w lans. *Wireless Communications, IEEE*, 15(1): 40-47.
27. Wang, C.Y. and H.Y. Wei, 2009. Ieee 802.11 n mac enhancement and performance evaluation. *Mobile Networks and Applications*, 14(6): 760-771.
28. Xiao, Y., 2004. Packing mechanisms for the ieee 802.11 n wireless lans. In: *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE. IEEE*: pp: 3275-3279.
29. Xiao, Y., 2005. Ieee 802.11 n: Enhancements for higher throughput in wireless lans. *Wireless Communications, IEEE*, 12(6): 82-91.
30. Xiao, Y., 2005. Ieee 802.11 performance enhancement via concatenation and piggyback mechanisms. *Wireless Communications, IEEE Transactions on*, 4(5): 2182-2192.
31. Xiao, Y. and J. Rosdahl, 2002. Throughput and delay limits of ieee 802.11. *Communications Letters, IEEE*, 6(8): 355-357.
32. Xiao, Y. and J. Rosdahl, 2003. Performance analysis and enhancement for the current and future ieee 802.11 mac protocols. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(2): 6-19.