Quick Data-retrieving for U-APSD in IEEE802.11e WLAN Networks

Shojiro Takeuchi*, Kaoru Sezaki† and Yasuhiko Yasuda*
*Grad. School of Science and Engineering, Waseda University
Email: shojiro,yasuda@yasuda.comm.waseda.ac.jp Telephone: +81 (3) 5286–3388
‡Center for Spatial Information Science at University of Tokyo
Email: sezaki@iis.u-tokyo.ac.jp Telephone: +81 (3) 5452–6268

Abstract— **The IEEE802.11e defines a MAC protocol, which provides EDCA (enhanced distributed channel access) and HCCA (HCF controlled channel access) to support differentiation service over WLAN (wireless LAN). In IEEE802.11e WLAN, real-time application such as VoIP (Voice over IP) can have more chance to access the WM (wireless medium) than non real-time application. In addition to QoS support in WLAN, power consumption is a critical issue when WLAN is used in handheld devices. For power saving in the use of real-time applications like VoIP under EDCA, U-APSD (Unscheduled Automatic Power Save Delivery) was proposed in [2] and [4]. In fact, it can save power consumption and works well when it is used for bi-directional voice connections generated at constant bit rate. However, when it is used for real-time applications like ON-OFF traffic, buffuring delay at AP (access point) increases. To reduce the buffereing delay, this paper proposes two mechanisms. Simulation results show that they can alleviate buffering delay generated at AP.**

I. Introduction

IEEE802.11 [1] WLAN technology has become very popular because of its advantage in price and bandwidth. Nowadays, WLAN is mainly used for Internet access, but real-time applications like VoIP are identified as promising applications for WLAN. Since real-time applications require distinct specific features, such as delay sensitivity or bandwidth requirement, it is desired to support differentiation services in IEEE802.11. Hence, IEEE 802.11e working group is now discussing new 802.11 MAC protocol, which provides QoS [2]. The 802.11e HCF (hybrid coordination function) provides both a contention-based channel access, called EDCA, and a controlled channel access, referred to as HCCA. In this paper we focus on EDCA.

Power consumption is a critical issue when WLAN is used in handheld devices. In IEEE802.11 [1], PSM (Power Saving Mechanism), which is called LegacyPS, is supported. The general idea of LegacyPS is for STAs to keep their radio off when they do not have to send or receive frames. For details, see [1]. But LegacyPS is not suitable for real-time applications with periodicity such as VoIP because STAs with LegacyPS need to be awake for more than an actual active voice call. Therefore, in IEEE802.11e, a new PSM, which is referred to as U-APSD, is proposed [2], [4]. An AP begins transmissions of downlink frames to an STA operating with U-APSD when it receives an uplink data frame from the STA. In the next section, we explain U-APSD in detail. U-APSD is suitable for real-time applications like VoIP, and especially for bi-directional traffic generated at constant bit rate. However, if U-APSD is used for real-time traffic with silence period, e.g. ON-OFF traffic [5], downlink frames are caused large buffering delay at an AP. This is because an STA may have no uplink frames and an AP cannot start downlink transmissions. Actually, in U-APSD the STA is allowed to send a null-data frame in order to start transmissions of downlink frames buffered at the AP if it has no uplink data frame. However, how often or when the STA sends a null-data frame is not specified in IEEE802.11e [2].

In this paper, we propose two mechanisms to alleviate buffering delay generated when U-APSD is used for ON-OFF real-time traffic. The first one uses periodic transmissions of null-data frames to initiate downlink transmissions during OFF period of an uplink flow and it can be implemented within the range of IEEE802.11e. Actually, in U-APSD, transmissions of periodic uplilnk frames are needed to maintain small delay and desired throughput of real-time traffic. However, periodic transmissions of uplink null-data frames become overhead and cause a waste of the wireless bandwidth. When there is much real-time traffic operating with U-APSD, collisions often occur between uplink null-data frames or between an uplink null-data frame and other traffic. To overcome these problems, we propose the second mechanism. It can allow the AP to periodically start downlink transmissions without the reception of uplink null-data frames. Although it needs a little modifications in U-APSD defined in IEEE802.11e [2], reduction of overhead caused by uplink null-data frames results in that it can accommodate more traffic than the first proposal.

The remainder of this paper is organized as follows. The next section introduces IEEE 802.11e EDCA and U-APSD. In Section 3, our two proposals are presented. Performance evaluations of them are carried in the section 4. Finally in section 5, we present the conclusion.

II. IEEE802.11e

A. EDCA

EDCA provides differentiated and distributed channel access to the WM based on 8 different UPs (User Priorities). The EDCA mechanism defines four ACs (Access Categories) to support differentiated channel access. The mapping from UPs to ACs is defined in [2]. Differentiated channel access is achieved through varying the amount of time a station would

1-4244-0270-0/06/$20.00 (c)2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

sense the channel to be idle and the length of CW (Contention Window) for a backoff. Four ACs use different values of AIFS (arbitration inter frame space), the minimum CW size and the maximum CW size. In this paper, for AC k ($0 \leq k \leq 3$), the minimum CW size is $CW_{min}[k]$, the maximum CW size is $CW_{max}[k]$, and the arbitration inter frame space is $AIFS[k]$. Further the arbitration inter frame space number is $AIFSN[k]$. Since, in EDCA, real-time traffic in one AC has a smaller, $AIFSN[k]$, $CW_{min}[k]$, and $CW_{max}[k]$ than non real-time trafifc in another AC, it can have more opportunities to access the WM earlier than non real-time traffic.

B. U-APSD

In general, when an AP delivers downlink frames to STAs operating with power savig mode including U-APSD, it has to confirm that they are awake. If they are awake, it can transmit downlink frames to them. Otherwise it has to buffer those frames until they are awake. U-APSD is defined in IEEE802.11e [2], [4] as a rule of downlink frame derivery and is used under EDCA. On the other hand, uplink frames can be transmitted based on EDCA from STAs at any time because the AP is always awake.

The basic idea of U-APSD is to use a specific interval called U-SP (Unscheduled Service Period) for an AP to deliver downlink frames to an STA. Fig.1 shows operations of U-APSD. During U-SP, the AP knows that the STA is awake and is ready to receive downlink frames. The STA using U-APSD can start a U-SP with transmission of an uplink data or null-data frame. An uplink data or null-data frame used to initiate a U-SP is called a trigger frame. After the AP responds with ACK, a U-SP starts and the AP begins to deliver downlink frames destined for the STA under EDCA. The length of U-SP is decided by MAX SP (Service Period) length, which indicates the maximum number of frames the AP can transmit during an U-SP, and it is determined through (re) association procedure between the AP and STA. During a U-SP, the AP has to deliver at least one downlink frame to the STA, but no more than the number specified in MAX SP length. Even though the AP receives uplink frames from the STA after a U-SP starts, they are not treated as trigger frames. If the AP receives a trigger frame from the STA and has no buffered frame destined for the STA, it transmits a null-data frame to the STA becaues the AP has to send at least one frame to it. At the end of a U-SP, the AP informs the STA about the end of the U-SP using EOSP (end of service period) field set to 1 in QoS control field in data frame as shown in Fig.1. On receiving data or null-data frames with EOSP set to 1, the STA can enter into the doze mode and the AP cannot deliver further downlink frames until a new U-SP starts. In addition to the use of EOSP field, if there are more buffered data destined for the STA at the end of a U-SP, the AP sets MoreData field in frame control field in MAC header to 1 and informs the STA of that more frames are buffered as shown in Fig.2. On the receipt of a downlink frame containing EOSP set to 1 and MoreData set to 1, the STA has to send uplink frames to quickly retrieve frames buffered at the AP. However, if there is no uplink frame buffered at the STA, it can send a null-data frame to start a new U-SP. As mentioned earlier, an STA operating with power saving mode can transmit an uplink frame at any time. Therefore, it wakes up before starting a transmission of the uplink frame and it can enter into the doze mode after finishing the transmission. But, in U-APSD, the transmitted uplink frame becomes a trigger frame, and so the STA has to wait for a downlink data or null-data frame and continue to be awake until the downlink frame with EOSP set to 1 is delivered from the AP. This is because the AP recognizes the STA is awake after the AP receives a trigger frame and a U-SP ends after the AP transmits to the STA a data or null-data frame with EOSP set to 1.

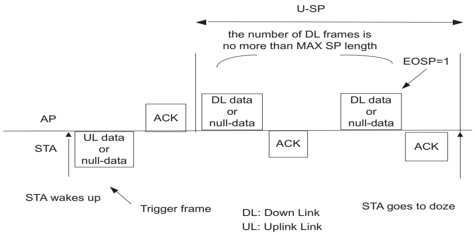

Fig. 1. U-APSD operations

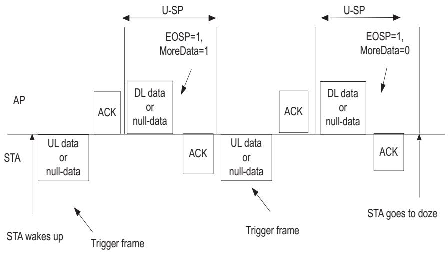

Fig. 2. STA operations on the receipt of EOSP=1 and MoreData=1

Actually, when an STA uses U-APSD for bi-directional traffic generated at constant bit rate, it can efficiently achieve power saving because it can periodically wake up and a U-SP can be constantly initiated by uplink frames. Moreover it can retrieve downlink frames from the AP during the U-SP. Thus, it does not have to be awake for a long period [4]. However, since U-APSD does not carefully consider the property of ON-OFF real-time traffic [5], large buffering delay at an AP will be generated when U-APSD is used for ON-OFF traffic as described in Fig.3. This is due to the fact that in U-APSD a U-SP and transmissions of downlink frames cannot start until the AP receives an uplink trigger frame. During the OFF period of an uplink flow, the STA may not have any uplink frames and cannot start a U-SP. On the other hand, downlink data frames arriving at the AP are buffered and cannot be delivered to the STA. Thus if the OFF period is long, large buffering delay with regard to downlink frames is generated at the AP and furthermore buffered frames may be discarded when its buffer overflows or when they have been buffered over pre-defined period.

III. Periodic U-SP for quick data retrieving

In order to reduce buffering delay generated at an AP, an STA operating with U-APSD has to periodically start a U-SP. However, even though a U-SP is frequently initiated, there may be no frames buffered at the AP. Thus, energy will be wasted. On the other hand, if a U-SP is rarely initiated, buffering delay

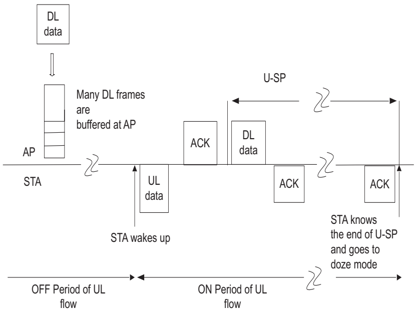

Fig. 3. Large buffering delay in the use of ON-OFF traffic

increases even though energy is saved. In fact, the relation between saving energy and small delay is the trade-off. In the followings, there are two proposals to reduce buffering delay, considering the trade-off. They are focusing on how to and when to periodically start a U-SP in case where U-APSD is used for ON-OFF real-time traffic. And both of them operate based on U-APSD. While the first one can be achieved within the area defined in IEEE802.11e [2], the second one needs a little of modifications with regard to initiation of U-SP but can reduce overhead generated in the first one. Hence it can efficiently utilize the wireless bandwidth.

A. U-APSD with Periodic Transmissions of Trigger Frame

When an STA begins a U-SP to retrieve buffered frames, it needs to transmit a trigger frame (uplink data or null-data frame) as specified in U-APSD [2]. If there is no uplink frame, it has to send a null-data frame as a trigger frame in order to initiate a U-SP. Therefore the basic idea behind the first proposal, which is called U-APSD with Periodic Transmissions of Trigger frame (UPTT), is that an STA maintains a timer, which is called triggering timer, in order to periodically initiate transmissions of a null-data frame as a trigger frame and that if the STA has no uplink frame at the expiration of the triggering timer, it transmits a null-data frame to begin a U-SP. With the use of null-data frames, UPTT is fully compatible with U-APSD defined in [2]. Since the trade-off between saving energy and small delay has to be balanced, an interval (called triggering interval) between consecutive triggering timers has to be carefully choosen. However, it is very difficult for STAs to predict arrivals of downlink packets at an AP and so it is very difficult to completely optimize the triggering interval.

In order to balance the trade-off, the triggering interval can be varied in UPTT. The basic strategy to decide a triggering interval is that when there are buffered frames at the AP, the STA sets the triggering interval to minimum one and frequently sends null-data frames to quickly retrieve them, and that when there is no buffered frame at the AP, the STA sets the triggering interval to larger one to continue the doze mode for a long time and to save energy. Besides when the STA increases the triggering interval, it is incremented in binary exponential fashion like a back-off counter. If T_{tr} denotes a triggering interval, it is incremented like the following: $T_{tr} = 2 \times T_{tr}$. Hence, if there is no buffered frame at the AP, the triggering interval fast becomes a large value, and so the STA can continue to sleep for a long time. Fig. 4 and 5 show examples of frame exchage sequence and triggering timer operations in UPTT. The triggering interval, T_{tr}, can be varied from minimum to maximum value. In the bellow, detailed operations of UPTT are described.

If an STA decides that U-APSD is used for an AC or several ACs, a triggering timer is set to maximum interval when WLAN device starts to work. As shown in Fig.4, at the expiration of a trigger timer, the STA becomes awake and a null-data frame is transmitted. However, if it has buffered uplink data frames, it does not send a null-data frame since those buffered data frames can be trigger frames. Regardless of whether or not a null-data frame is transmitted, the triggering interval is incremented by double but is not beyond the maximum value, and the STA sets the triggering timer and waits for downlink frames. The reason why the triggering timer is set just after the expiration of the timer is the following. If it is set after interpreting EOSP and MoreData fields in received downlink frames, it cannot be set when transmissions of downlink frames are failed. Therefore, in UPTT, the triggering timer is set just after the expiration of the timer, and it is, again, reset after interpreting EOSP and MoreData fields in downlink frames. How to reset the triggering interval when interpreting these fields is shown in the following.

On the receipt of downlink frames from the AP, the STA checks EOSP and MoreData field in MAC header. Since EOSP set to 0 means that U-SP does not end and further downlink frames come from the AP, the STA continues to be awake and waits for downlink frames. In case of EOSP=1 and MoreData=1, the STA transmits a trigger frame to the QAP to quickly retrieve buffered frames. Before transmitting it, the STA confirms whether there are buffered uplink frames. If there is no buffered frame, the STA sends a null-data frame as a trigger frame. Otherwise a buffered frame will be transmitted as a trigger frame. In the example shown in Fig.4, a null-data frame is sent as a trigger frame. And then the STA clears a pending timer and resets the timer with the triggering interval set to minimum one. This is because it is better to retrive frames buffered at the AP as soon as possible. In case of EOSP=1 and MoreData=0, the STA can enter into the doze mode if there is no uplink frame. However, if the received frame is a data frame, the STA sets the triggering interval to minimum one and resets the triggering timer as shown in Fig.4. This is due to the fact that data frames of a real-time flows periodically arrive at the AP. On the other hand, if the received frame is a null-data frame (i.e., even though the AP receives a trigger frame and a U-SP starts, the AP has no buffered data destined for the corresponding STA. In this case, the AP sends a null-data frame with EOSP=1 and MoreData=0 to end the U-SP.), the STA can go to doze mode and continue to sleep until a triggering timer expires or it has uplink data frame. As mentioned earlier, STAs can begin uplink data transmissions at any time in U-APSD. Therefore, when an uplink data frame comes into the transmission buffer at an STA and can be transmitted as a trigger frame, the triggering timer is cancelled and just reset to T_{tr}, as presented in Fig.5. In this case, T_{tr} is not updated and is the same as the value which was used by

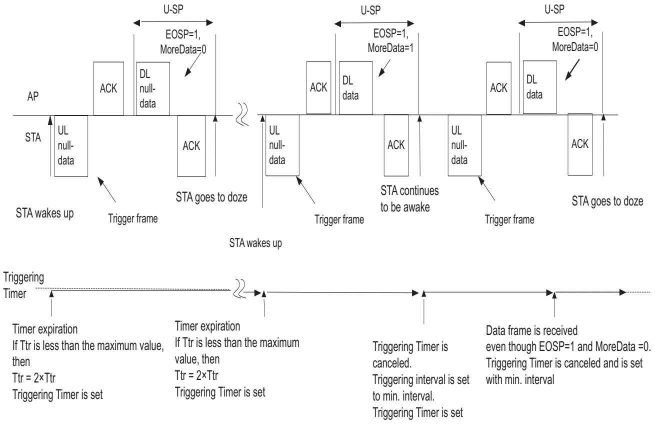

Fig. 4. UPTT operations

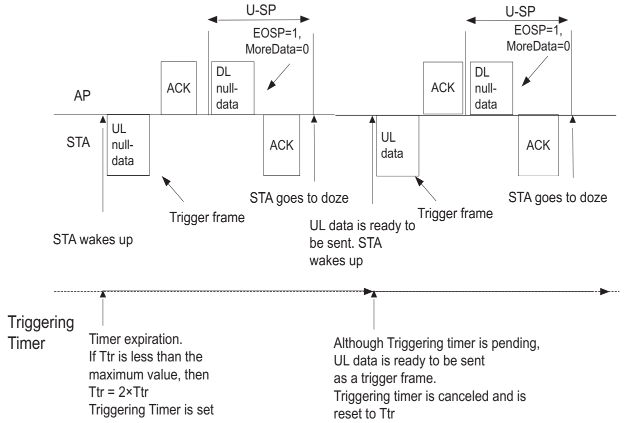

Fig. 5. Timer cancelation in UPTT when sending an uplink data frame

the cancelled triggering timer.

Since, in UPTT, an STA maintains a triggering timer to periodically transmit a null-data frame as a trigger frame, it can quickly retrieve data frames buffered at an AP even though U-APSD is used for ON-OFF traffic. Furthermore, since the triggering interval is increased in binary exponential manner when there is no frame buffered at the AP, energy is efficiently saved. Besides, a null-data frame is transmitted only when the STA has no uplink frame, and thus even if UPTT is used for bi-directional real-time traffic generated at constant bit rate, many null-data frames do not have to be transmitted. Therefore it is applicable for bi-directional traffic generated at constant bit rate. However, a null-data and the corresponding ACK frame really consumes the wireless bandwidth even though UPTT is fully compatible with and periodic transmissios of null-data frames are necessary in U-APSD defined in IEEE802.11e [2]. Moreover, since null-data frames may contend with other transmissions of a data or management frame and may be suffered from retransmissions due to channel error, overhead generated by transmissions of null-data frames will be increased.

B. U-APSD with Timer-based Wake-up Rule

Compared to UPTT, in the second proposal which is referred to as U-APSD with Timer-based Wake-up Rule (UTWR), an STA does not have to send a null-data frame to start a U-SP. Hence UTWR can remove overhead generated due to uplink null-data frames. As similar to UPTT, UTWR operates based on U-APSD. Moreover a triggering timer is also used in UTWR in order to reduce buffering delay and also for energy-saving the triggering interval is increased in binary exponential manner when there is no buffered frame at the AP. However, in UTWR, both the STA and AP maintain triggering timers in order to quit uplink transmissions of null-data frames. That is, UTWR enables the STA and AP to periodically start an U-SP without transmissions of uplink null-data frames and they use the same rule to determine a triggering interval. In addition to that a U-SP is initiated at the expiration of the triggering timer, as similar to U-APSD and UPTT, it also starts when the AP receives uplink data frames as trigger frames. As for uplilnk data transmissions, STAs operating with UTWR can send uplink data frames at any time. In the following, UTWR operations both in STA and AP are explained in detail.

Fig.6 and 7 show examples of frame exchage sequence and triggering timer operations. In UTWR, the AP maintains a triggering timer for an STA operating with UTWR and the STA also maintains a triggering timer. The triggering timers set to maximum interval start to work at a time pre-determined by both the AP and STA through the (re) association procedure. And also they are set when the AP sends and the STA receives a data frame or null-data frame with EOSP=1 and MoreData=0 as shown in Fig.6. In case where the AP sends a data frame with EOSP=1 and MoreData=0, the triggering interval at the AP is set to minimum one and it sets its triggering timer after sending the data frame because downlink real-time data periodically arrives at the AP. On the receipt of the data frame, the STA accordingly sets its triggering interval to minumum one and sets its triggering timer. On the other hand, in case of null-data frame with EOSP=1 and MoreData=0, triggering intervals both in the AP and the STA are incremented by double and the triggering timers are set as presented in Fig.6. In fact, there is a lag between time when the AP sends the data frame and time when the STA completes the transmissions of the ACK frame Therefore, the triggering timer in the AP should be set considering the lag. Furthermore, when the STA receives a data or null-data frame with EOSP=1 and MoreData=0, it may fail to transmit an ACK frame to the AP and go to the doze mode. In this case, the STA cannot receive the frame retransmitted by the AP. This phenomenon also occurs in U-APSD [2] and UPTT. However, in UTWR, triggering timers can be successfully set both at the AP and the STA since the AP considers the lag and sets its triggering timer. Of course, if the frame is retransmitted by the AP, its triggering timer is accordingly reset. On the expiration of trigger timer, the STA wakes up and also the AP knows that the corresponding STA wakes up, and if the AP has no buffered frame destined for the corresponding STA, it sends a null-data frame with EOSP=1 and MoreData=0 to the STA and both AP and STA increment the triggering interval by double as shown in Fig.6. Sending a null-data frame to the STA is for ending a U-SP. This operation is the same as one used when in U-APSD and UPTT the AP receives a trigger frame and there is no buffered frame destined for the STA. On the other hand, if there are buffered frames at the AP,

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2006 proceedings.

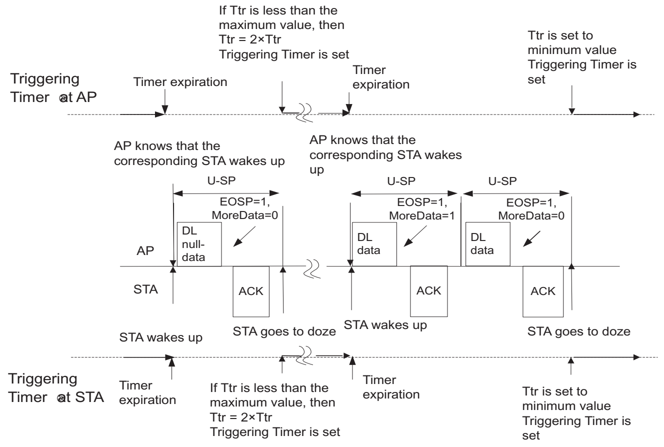
If Ttr is less than the maximum value, then Ttr = 2×Ttr Triggering Timer is set
Ttr is set to minimum value Triggering Timer is set
Triggering Timer at AP
Timer expiration
Timer expiration
AP knows that the corresponding STA wakes up
AP knows that the corresponding STA wakes up
U-SP
U-SP
U-SP
EOSP=1, MoreData=0
EOSP=1, MoreData=1
EOSP=1, MoreData=0
AP
DL null-data
DL data
DL data
STA
ACK
ACK
ACK
STA wakes up
STA goes to doze
STA wakes up
STA goes to doze
Triggering Timer at STA
Timer expiration
If Ttr is less than the maximum value, then Ttr = 2×Ttr Triggering Timer is set
Timer expiration
Ttr is set to minimum value Triggering Timer is set

Fig. 6. UTWR operations

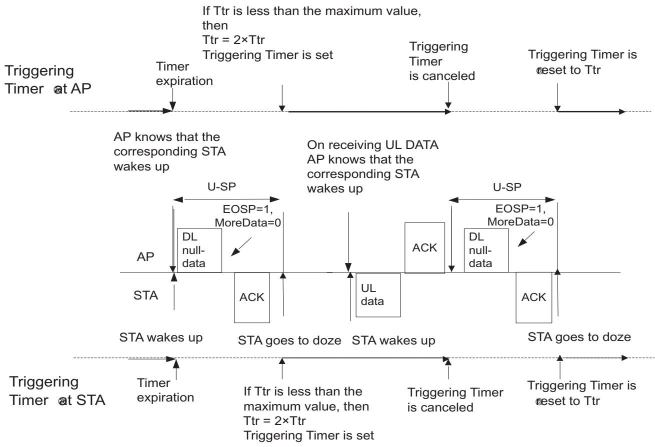
If Ttr is less than the maximum value, then Ttr = 2×Ttr Triggering Timer is set
Triggering Timer is canceled
Triggering Timer is reset to Ttr
Triggering Timer at AP
Timer expiration
AP knows that the corresponding STA wakes up
On receiving UL DATA AP knows that the corresponding STA wakes up
U-SP
U-SP
EOSP=1, MoreData=0
EOSP=1, MoreData=0
AP
DL null-data
ACK
DL null-data
STA
ACK
UL data
ACK
STA wakes up
STA goes to doze
STA wakes up
STA goes to doze
Triggering Timer at STA
Timer expiration
If Ttr is less than the maximum value, then Ttr = 2×Ttr Triggering Timer is set
Triggering Timer is canceled
Triggering Timer is reset to Ttr

Fig. 7. Timer cancelation in UTWR when sending an uplink data frame

data delivery operations are processed and there is no need to send a null-data frame. In case where the AP sends a data frame with EOSP=1 and MoreData=1, it immediately starts a new U-SP in order to quickly deliver buffered frames to the corresponding STA as presented in Fig.6. And the STA also knows the new U-SP begins on the receipt of data frame with EOSP=1 and MoreData=1, and continues to be awake. Since in UTWR a new U-SP can be initiated without uplink data or null-data frames, transmissions of those frames needed in U-APSD and UPTT is eliminated. As mentioned earlier, STAs can transmit uplink frames at any time. Therefore if the AP receives an uplink frame from the STA as a trigger frame, a new U-SP is initiated and the timer is cancelled as shown in Fig.7. When the U-SP is started by the uplink frame and the AP has no buffered frame destined for the STA, the AP sends and the STA receives a downlink null-data frame with EOSP=1 and MoreData=0. In this case, they do not increment their triggering intervals by double but they just reset their triggering timer using the current value of T_{tr} as shown in Fig.7. Resetting the current value of T_{tr} is for avoiding a situation where triggering intervals are fast increased to a larger value when there are many uplilnk transmissions from the STA.

UTWR removes overhead of uplink transmissions of null-data frames. Hence the wireless bandwidth is more efficiently used in UTWR than in UPTT. Besides even if UTWR is used for bi-directional real-time traffic generated at constant bit rate, it does not make STAs awake for a long time because in UTWR a U-SP can be also started by uplink data frames and a triggering timer is cancelled when uplink data frames are transmitted.

IV. Performance Evaluation

We performed simulations using NS2 [7]. In these simulations, UPTT and UTWR are evaluated and compared to U-APSD without periodic initiation of a U-SP, which is referred to as OnlyMore. STAs operating with OnlyMore send uplink null-data frames as trigger frames only if they receive downlink frames with EOSP=1 and MoreData=1 and they do not have any uplink data frames. This means that they do not start a U-SP periodically unlike UPTT and UTWR. In all OnlyMore, UPTT and UTWR, MAX SP length is set to 2. In UPTT and UTWR, mimimum and maximum triggering intervals are set to 6.25ms and 100ms, respectively. We assume that both AP and STA operate with IEEE802.11b[3], basic rate is 2.0Mbps, and data rate is 11.0Mbps. For each AC, we have the following parameters [2]: $CW_{max}[0] = 1023$, $CW_{min}[0] = 31$; $CW_{max}[1] = 1023$, $CW_{min}[1] = 31$; $CW_{max}[2] = 31$, $CW_{min}[2] = 15$; $CW_{max}[3] = 15$, $CW_{min}[3] = 7$; $AIFSN[0] = 7$, $AIFSN[1] = 3$, $AIFSN[2] = 2$, $AIFSN[3] = 2$. Each AC operates with the drop-tail algorithm and can accommodate 100 packets. Two simulation scenarios are considered. One is performed under only voice traffic (Scenario1) and another is done under voice and Data traffic configurations (Scenario2). In each Scenario, simulations are performed through varying the number of voice STAs and in Scenario2 there are 6 STAs with uplink Data traffic. Voice STAs operate with one of power saving mechanisms (OnlyMore, UPTT or UTWR) but STAs with uplink Data traffic work with active mode (i.e. they are always awake). Each voice flow is bi-directional ON-OFF traffic and both busy and idle intervals follow exponential distribution with mean 352ms and 650ms, respectively [5]. During busy period, voice traffic is generated at a constant interval, 20 ms and has a fixed payload size of 208 bytes. Its rate is 83.2Kbps. Data traffic has a fixed payload size of 750 bytes. It is UDP traffic and is generated by a constant interval, 10 ms. In each simulation run, simulation runs for 60s. Simulation traffic is generated in the interval from 0s to 30s. We use simulation outputs in the interval from 30s to 60s in each simulation. We evaluate packet dropping probability of voice traffic both in Scenario1 and 2. As for delay evaluation, average delay of downlink voice traffic is considered in both scenarios. In Scenario2, average throughput of Data traffic is also evaluated. Furthermore, energy efficiency is evaluated in both scenarios. The efficacy of power consumption can be often evaluated only by using the amount of comsumed energy. However, power consumption is closely related to how much STAs transmit or receive traffic. If a protocol can promote high throughput, power consumption of the STA on which it is implemented will be large. Therefore, we use *energy goodput* described in the below to evaluate power efficiency [6]:

$$energy\ goodput = \frac{total\ bits\ transmitted}{total\ energy\ consumed}$$

where the total bits transmitted are calculated for application layer packets only. The unit of energy goodput is Kbits/J.

A. Simulation results in Scenario1

Fig.8 and 9 show packet dropping probability of voice traffic and average delay of downlink voice traffic, respectively. It is obvious that the dropping probability in OnlyMore increases when the number of voice STAs increases and that average delay of downlink voice traffic in OnlyMore is extremely large. This is due to the fact that when there is no uplink data frame during OFF period of uplink voice traffic, STAs with OnlyMore cannot start a U-SP. Consequently many downlink voice packets are buffered at the AP and are discarded. And also large buffering delay is generated at the AP. When the buffer for downlink voice packets overflows, they are discarded. Therefore, when the number of voice traffic increases, delay of downlink voice traffic is not so increased even though dropping probability is much increased. On the other hand, since UPTT and UTWR can periodically start a U-SP, delay of downlik voice traffic is much smaller and dropping probability is much lower than those in OnlyMore. However, since sending null-data frames to start a U-SP in UPTT is really overhead and a waste of the wireless bandwidth, dropping probability in UPTT becomes high when the number of voice STAs is large. Accordingly delay of downlink voice traffic in UPTT is increased when there are many voice STAs. Meanwhile, STAs with UTWR can periodically start a U-SP without transmissions of uplink null-data frames and UTWR can get rid of overhead generated by transmissions of them in UPTT. Hence, in UTWR, low dropping probability is maintained and small delay of downlink voice traffic is also achieved even when the number of voice STAs is large. Fig.10 presents a comparison of *energy goodput*. In all mechanisms, *energy goodput* declines when the number of voice STAs is increased. This is due to the fact that when the number of voice STAs is large, an STA has to be awake for long time because of contentions with other voice STAs and so it consumes more energy. Since STAs with OnlyMore do not have to often wake up, OnlyMore has the highest *energy goodput* when the number of voice STAs is small. However, delay of downlink traffic is extremely high. Therefore the quality of voice traffic must be really bad. Besides, when the number of STAs is increased, *energy goodput* is significantly decreased because the buffer of downlink voice traffic overflows and downlink voice packets cannot be delivered to them. On the contrary, since UPTT and UTWR can deliver more downlink voice traffic when the number of voice STAs is high, *energy goodput* in both mechanisms is higher than OnlyMore in case there are many voice STAs. Moreover, compared to UPTT, UTWR can reduce overhead and deliver more downlink voice traffic because it does not use uplink null-data frames as trigger frames, and further it can accommodate more voice traffic than UPTT. Thus, UTWR has higher *energy goodput* than UPTT, especially when there are large number of voice STAs.

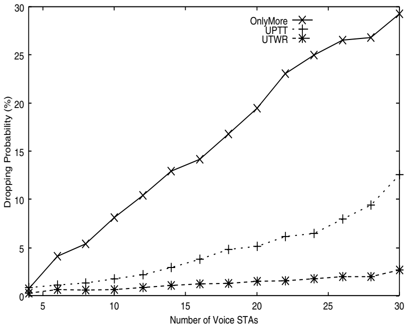

Fig. 8. Dropping probability of voice traffic

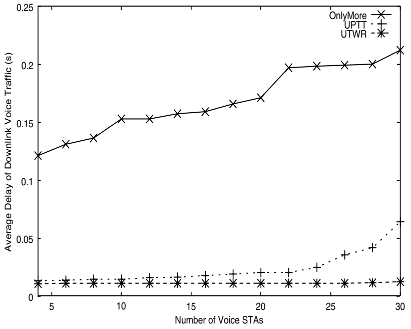

Fig. 9. Average delay of downlink voice traffic

B. Simulation results in Scenario2

Fig.11 and 12 present dropping probability of voice traffic and average delay of downlink voice traffic in Scenario2. Compared to those in Scenario1, dropping probability in all mechanisms in Scenario2 is slightly higher. This is because data traffic disturbs voice traffic under contention-based access even though voice traffic has more opportunities to access the WM. As similar to the results in Scenario1, dropping probability in OnlyMore is really high and average delay of downlink voice traffic in OnlyMore is very large. In OnlyMore, a U-SP cannot be started even when downlink voice packets arrive at the AP. Thus large buffering delay occurs and they are discarded at the AP. On the contrary, since STAs operating with either UPTT or UTWR can periodically start a U-SP and retrieve frames buffered at the AP, delay of downlink voice traffic is extremely small. However, UPTT uses transmissions of uplink null-data frames to start a U-SP and thus these uplink null-data frames cause a waste of the wireless bandwidth. Therefore, when there are large number of voice STAs, dropping probability in UPTT is more increased and also delay of downlink voice traffic is larger than UTWR. Fig.13 shows

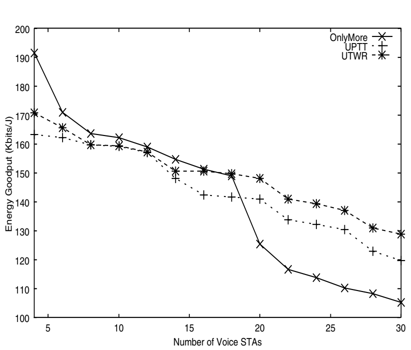

Fig. 10. Comparison of energy goodput

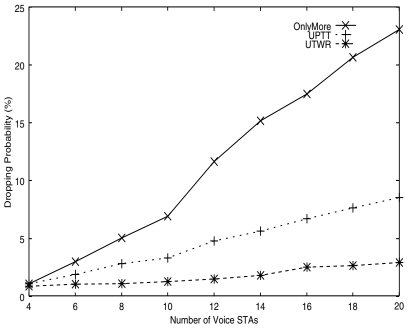

Fig. 11. Dropping probability of voice traffic in Scenario2

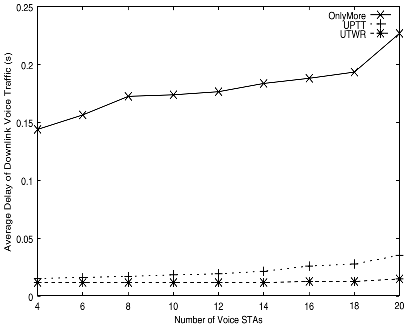

Fig. 12. Average delay of downlink voice traffic in Scenario2

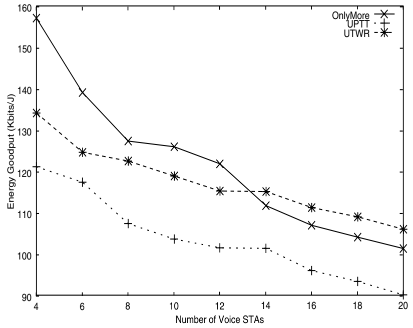

Fig. 13. Comparison of energy goodput in Scenario2

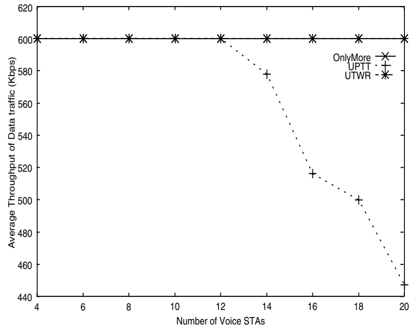

Fig. 14. Average throughput of Data traffic in Scenario2

a comparison of *energy goodput*. When the number of voice STAs is small, *energy goodput* in OnlyMore is the highest among all mechanisms because STAs with OnlyMore do not wake up periodically. However, since it has much large delay of downlink voice traffic, quality of voice traffic is not so good. Furthermore, when the number of voice STAs is increased, *energy goodput* in OnlyMore significantly decreases because a lot of downlink voice packets are discarded at the AP and they are not delivered to voice STAs. On the other hand, UPTT and UTWR can reduce overhead and deliver more downlink voice traffic even when the number of voice STAs is large. Therefore, *energy goodput* in both schemes is not so different from OnlyMore when there are many voice STAs. Moreover, since in UTWR uplink null-data frames are not used as trigger frames, more downlink frames are delivered than UPTT and thus *energy goodput* in UTWR is higher than UPTT. Finally, Fig.14 shows average throughput of Data traffic. Throughput of Data traffic is maintained in OnlyMore. This is because downlink voice traffic is infrequently retrieved from voice STAs and thus Data traffic can have much chance to access the WM. When the number of voice STAs increases, throughput of Data traffic in UPTT declines. Since in UPTT the wireless bandwidth is consumed by transmissions of null-data frames, bandwidth is not sufficiently allocated to Data traffic. On the other hand, in UTWR, throughput of Data traffic is maintained. Since UTWR does not use transmissions of uplink null-data frames to initiate a U-SP, UTWR can efficiently utilize the wireless bandwidth. Hence, even when the number of voice STAs is large, it can maintain throughput of Data traffic and also achieve small dropping probability of voice traffic.

V. Conclusion

When an STA uses U-APSD provided in IEEE802.11e [2] for real-time traffic with silence period, it is suffered from large delay of downlink traffic. To solve this problemm this paper proposed two mechanisms, UPTT and UTWR. From simulations results of voice ON-OFF traffic, it was obvious that both of them achieved reducing buffering delay and could accommodate more voice traffic. Furthermore, comparing UTWR with UPTT, since UTWR could remove overhead generated in UPTT, it could efficiently utilize the wireless bandwidth and thus could accommodate more voice traffic and also Data traffic. Since U-APSD operates under EDCA, a scheduling algorithm implemented in AP influence on power consumption of STAs. Therefore in future research we will work on it to reduce time period for which they are awake.

Acknowledgment

The authors would like to thank Mika Kasslin and Toshiaki Jozawa from Nokia Research Center Finland and Japan respectively for their valuable advices.

References

[1] *"Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications"*, IEEE Std 802.11 -1999 (Reaff 2003).

[2] *"Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements"*, IEEE Std. 802.11e/D8.0, Feb. 2004.

[3] *"Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band"*, IEEE Std. 802.11b-1999, 1999.

[4] Y. Chen, N. Smavatkul and S. Emeott, *"Power Management for VoIP over IEEE 802.11 WLAN"*, in Proc. of IEEE WCNC 2004, vol.5, no.1, March 2004.

[5] K. SRIRAM and W. Whitt, *"Characterizing Superposition Arrival Process in Packet Multiplexes for Voice and Data"*, IEEE Journal on Selected Areas in Communications, Vol.4, No.6, Sep. 1986.

[6] S. Takeuchi, K. Sezaki and Y. Yasuda, *"An Improved Power Saving Mechanism for MAC Protocol in Ad Hoc Networks"*, in Proc. of IEEE Globecom 2004, Nov. 2004.

[7] *"http://www.isi.edu/nsnam/ns/index.html"*