IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 17, 2019, accepted October 28, 2019, date of publication October 30, 2019, date of current version November 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2950464

A Target Wake Time Scheduling Scheme for Uplink Multiuser Transmission in IEEE 802.11ax-Based Next Generation WLANs

QINGHUA CHEN[1,2], (Member, IEEE), ZHENGQIU WENG[2], XIAOFENG XU[1], AND GUOLANG CHEN[2]

[1]School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310014, China
[2]Department of Information Technology, Wenzhou Vocational & Technical College, Wenzhou 325035, China

Corresponding author: Guolang Chen (glchenwz@qq.com)

This work was supported in part by the General Foundation of Zhejiang Provincial Education Department under Grant Y201840791, and in part by the Basic Scientific Research Project of Wenzhou Science and Technology Bureau under Grant G20180006.

ABSTRACT IEEE 802.11ax Wireless Local Area Networks (WLANs) introduce Orthogonal Frequency Division Multiple Access (OFDMA) physical layer technology to improve throughput in dense scenarios. In order to save power of battery operated stations (STAs), a novel broadcast Target Wake Time (TWT) operation for negotiating wake time between an access point (AP) and a group of STAs is also proposed by making full use of the new capability of uplink OFDMA-based multiuser transmissions. However, if the wake time of each STA which is determined by the offset and wake interval (listen interval) is not properly scheduled, deteriorated throughput and high power consumption occur because of collisions. In this paper, we take the advantage of uplink multiuser transmission with the novel TWT scheduling to maximize throughput. We first investigate the fundamental relationship between throughput and energy efficiency with several key aspects, such as the number of simultaneously active STAs, the number of eligible random access resource units, and the contention window size. We further derive the formulations of throughput and energy efficiency on the listen interval of each STA. Based on the relationship, a TWT-based sleep/wake-up scheduling scheme (TSS) is proposed to improve the throughput by reducing or even cancelling collisions. Simulation results demonstrate the effectiveness in terms of average throughput and energy efficiency. The TSS also makes a practical step towards a collision-free and deterministic access in future WLANs when cooperating with TWT service period scheduling.

INDEX TERMS IEEE 802.11ax, WLAN, power conservation, target wake time (TWT), orthogonal frequency division multiple access (OFDMA).

I. INTRODUCTION

Recently, Wireless Local Area Networks (WLANs) based on IEEE 802.11 standards [1] have become one of the most popular solutions for wireless internet accessing because of their mobility, flexibility of deployment, and cost efficiency as wireless access requirements increase [2], [3]. WLAN devices are currently being used in diverse environments which are characterized by the existence of a large number of Access Points (APs) and stations (STAs) in geographically limited areas, i.e., dense scenarios. Due to severe system performance degradation caused by collision and inference, a new IEEE 802.11ax [4] amendment is approved as the next generation WLAN, which employs many new technologies [5]–[7] like Orthogonal Frequency Division Multiple Access (OFDMA) based multiuser (MU) transmission, spacial reuse (SR), and Target Wake Time (TWT) [8] to improve performance.

The STAs are usually equipped with power limited batteries to support mobility and portability, and they can power down the transceivers to save power which is called power save mode (PSM), otherwise they will consume a substantial portion of available power [9], [10]. In order to improve existing power management mechanism, the individual TWT agreement which performs well in IEEE 802.11ah [11] WLANs with low traffic requirements and periodic data

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaowen Chu.

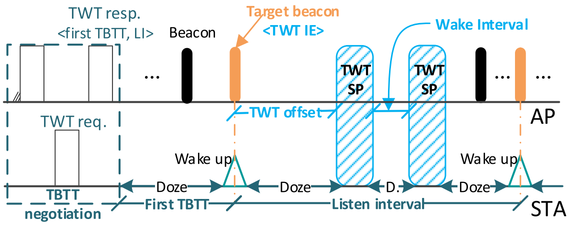


FIGURE 1. An example of TBTT negotiation procedure.

transmissions now is introduced in IEEE 802.11ax. It keeps STAs staying in PSM for a long time to save power by negotiating wake periods.

A key difference from 802.11ah is that 802.11ax has the MU transmission capability, which allows STAs to receive/transmit simultaneously. In other words, TWT requirements from STAs may be sent simultaneously, and transmission time scheduled by AP is shared by more than one TWT requesting STAs. Therefore, a novel broadcast TWT operation is proposed to improve performance by leveraging the new capability of MU transmission in a centralized way. As shown in Fig. 1, a STA in PSM listens to specific beacon frames,i.e., target beacons. A target beacon contains TWT information elements (IEs) which specify the shared TWT service periods (SPs) with TWT offsets and wake intervals. The time to receive the target beacons is negotiated at the beginning through a Target Beacon Transmission Time (TBTT) negotiation procedure by setting the offset (i.e., the first TBTT) and interval of successive TBTTs (i.e., Listen Interval).

In dense scenarios, many STAs may request TBTT negotiation simultaneously during a short period. The AP has to tackle these TWT requests by negotiating with first TBTTs and Listen Intervals (LIs) at the same time. Here, we denote these two parameters as $<f, t>$ for simplicity. The STAs which have received TWT IEs in their target beacons will ask to take part in or change the membership of a broadcast TWT, and wake up to transmit/receive in given TWT SPs simultaneously. Clearly, the same or improper TBTTs of STAs make them wake up simultaneously to receive the same target beacon and request for limited random access resource units (RA-RUs) especially when the number of active STAs is much bigger than the number of available RA-RUs. It will result in deteriorated throughput and higher power consumption. Therefore, there is a tradeoff in setting proper $<f, t>$ tuples.

Furthermore, the scheme for scheduling a proper TWT is out of scope in the IEEE 802.11ax draft 3.0 [4] and is implementation-specific [8]. In order to achieve higher throughput and efficient power consumption in infrastructure basic service sets (IBSSs), we propose a TWT based sleep/wake-up scheduling scheme (TSS) which takes the OFDMA based MU transmission capability into consideration by minimizing contention between STAs in each beacon slot. In the TSS, the AP is able to negotiate $<f, t>$ tuples by tending to use adaptive LIs according to network density and the number of available RA-RU and avoiding alignment of first TBTTs. Aiming at an efficient media access, the TSS equalizes the number of simultaneously active STAs and alleviates average contention in each beacon slot. This paper makes the following contributions.

- Considering the new capability of UL MU transmission, we propose TSS for the AP to arrange different active STAs in each TWT SP. The TSS is designed to decrease contention when the active STAs experience high collision or to increase the number of active STAs when the channel utilization is low.
- We derive the relationship between throughput and EE with the average number of active STAs in a beacon slot, the number of RA-RUs, contention window size, and the values of LIs. An optimization method of getting optimal LIs based on the relationship is proposed to maximize the overall throughput. In addition, we develop a first TBTT scheduling procedure to evenly arrange STAs to wake up in different beacon slots so that the expected throughput is achieved.
- Simulation results demonstrate the accuracy of the formulations and also show a better performance of the TSS on throughput and EE in high density IEEE 802.11ax based WLAN when comparing with other schemes.

The remainder of this paper is organized as follows. In Section II, we briefly review the related works about power management methods. Section III presents the TSS applicable to IEEE 802.11ax infrastructure WLANs. Based on the derived formulation, we propose a scheme to maximize throughput in Section III-B. In Section IV, algorithms of scheduling active STAs in each beacon slot by arranging first TBTTs are proposed. Simulation results are presented in Section V. Finally, conclusion is given in Section VI.

II. RELATED WORKS

In this section, we review the conventional power saving mechanisms and the TWT operation in IEEE 802.11ax, survey their proposals, and identify their issues. The main abbreviations adopted in this paper are listed in Table 1.

A. CONVENTIONAL PSM IN IEEE 802.11 WLANs

Two types of power state is defined in IEEE 802.11, namely, active state and doze state [1]. In active state, a STA is fully powered, and in doze state, the STA shuts down its transceiver to reduce power consumption when it does not transmit or receive. IEEE 802.11 also defines two types of power management mode, i.e., active mode and power save mode (PSM). In active mode, a STA remains awake continuously, while in PSM, the STA is usually in doze state and sometimes enter active state to receive or transmit frames.

PSM allows STAs to enter into doze state by buffering frames destined to these STAs at AP. The AP broadcasts a beacon frame every beacon interval to indicate data availability to all associated STAs [12]. According to IEEE 802.11 specification, the beacon interval is a management

TABLE 1. List of the main abbreviations.

Abbreviations	Description
WLAN	Wireless Local Area Network
HE	High Efficiency
AP	Access Point
STA	Station
MU	Multiuser
PSM	Power Save Mode
TWT	Target Wake Time
TWT SP	TWT Service Period
TBTT	Target Beacon Transmission Time
LI	Listen Interval
OFDMA	Orthogonal Frequency Division Multiple Access
RU	Resource Unit
TF	Trigger Frame
MBA	Multi-STA Block ACK
OCW	OFDMA Contention Window
EE	Energy Efficiency
LCM	Least Common Multiple
GCD	Great Common Divisor

parameter for the AP, and the LI must be a multiple of this beacon interval. A STA wakes up every LI to listen to their target beacons, and, if available, tries to transmit a power saving poll (PS-Poll) frame according to distributed coordination function (DCF) access scheme, i.e., after an idle distributed inter-frame spacing (DIFS) and a random backoff period, a PS-Poll is sent by PS STA before AP forwards the data frames to the STAs [13]. In other words, PS STAs sleep for most time and only wake up periodically to receive the frames buffered at the AP to save power.

Basically, power saving is achieved by minimizing the time for a STA to be awake and maximizing the time in doze mode [14]. The problem of collision in 802.11 WLANs is one of the key issues responsible for prolong awake time, because collisions result in retransmission or even frame loss. Over the years, much work has been done to help improve performance by employing adaptive LIs, coordinating LIs under different constraints, amending Traffic Indication Map (TIM) policy, and so on.

Smart Adaptive PSM proposed in [15] labels each application with a priority with the assistance of a machine learning classifier, which permits only high priority applications affect the client's behavior to switch to active mode, while low priority traffic is optimized for EE. In [16], Saeed and Kolberg presented a novel context-aware network traffic classification approach based on machine learning classifiers, and the classified output traffic is used to optimize the proposed context-aware LI. Gan and Lin [17] proposed a power conservation scheme to optimally schedule the awake time to minimize the number of active STAs in a beacon slot. In [18], Liu *et al.* presented a power saving mechanism with offset LI (OLi) for Machine-to-Machine (M2M) communication in IEEE 802.11ah WLANs. OLi spreads the M2M traffic evenly with calculated offset to alleviate network contention and reduce packet delay by minimizing the maximum number of STAs per beacon interval. Based on the concept of Chinese Remainder Theorem (CRT), Kuo and Chen [13] who were concerned with multicast message transmission in asynchronous ad hoc network presented an asynchronous power saving protocol to generate awake frequencies (i.e., LI). Therefore, all the STAs can wake up simultaneously and receive multicast messages even if clock drift happens.

Besides, researchers proposed many different power saving schemes under different constraints. In [19], Chen *et al.* found a Mobility-aware PSM (M-PSM) scheme to save more energy by employing a smart buffering strategy and an adaptive LI adjustment algorithm according to buffer limit, channel condition and moving direction of STAs. Buffered frames are sent only when the channel condition is well to achieve high rate. The scheme is very suitable for scenarios where STAs is lightly-loaded with delay-insensitive traffic. Zhu *et al.* [20] [21] proposed a time-based power management (TPM) scheme to adjust the idle time and doze time (LI) by adding an idle timer and introducing a power-aware buffer management scheme to accommodate as many PS STAs as possible, under the condition of a fixed amount of memory size at AP.

Si *et al.* [22] utilized the TIM to save power, and proposed a modified TIM setting scheme which allows only a part of STAs containing pending frames at the AP to join the contention after a beacon. The number of contenting STAs is evaluated by the AP to improve throughput and power consumption in saturated WLANs.

The efforts made by the researchers above have greatly improved the PSM in different application scenarios. However, these PSM schemes need to be improved due to the soaring number of STAs in dense WLANs. To alleviate the channel contention, the TWT operations, which use request and reply frames to facilitate channel reservation by employing the next wake time and service periods, are proposed by IEEE 802.11ah. TWT requesting STAs only wake up at scheduled times (TWTs) for scheduled durations (TWT SPs). The TWT operation is now introduced in IEEE 802.11ax with the same characteristic of a large number of STAs.

B. A NOVEL TWT OPERATION IN IEEE 802.11AX WLANs

Obviously, the AP, which is the focus of all UL services and the starting point of all downlink (DL) services, has a global perspective. It is an important breakthrough in IEEE 802.11ax to enhance WLANs' scheduling capability through APs' global vision by adopting TWT mechanisms for power saving. To adapt to the relevant technologies in IEEE 802.11ax, especially the MU capability, IEEE 802.11ax has greatly enhanced the TWT mechanism by introducing a novel broadcast TWT operation which negotiates the first TBTT, LI, TWT start time (offset), and wake interval with a group of STAs simultaneously to reserve services.

In a broadcast TWT operation, a STA negotiates the LI with the AP via a TWT request frame. Next, the AP replies with a TWT respond frame to accept or reject the parameters like LI, the first TBTT, and its ultimate LI. After the TWT request is granted, the STA will switch into doze mode until its timing synchronization function (TSF) matches the first negotiated TBTT. The STA that wakes up at TBTT will join

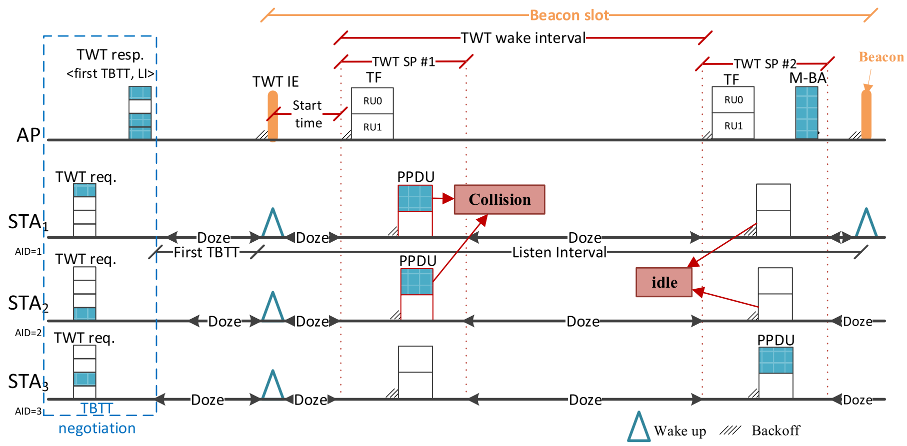


FIGURE 2. An example of a TWT operation in IEEE 802.11ax-based WLANs.

in the broadcast TWT as a member according to the TWT IE which is contained in the target beacon. The STAs join in a same TWT SP perform transmissions simultaneously.

In [8], Nurchis and Bellalta gave the throughput analysis of the TWT operation in 802.11ax WLANs which shew that the TWT operation results in 10 times throughput increase than the scenario in which MU capability is not supported. The TWT mechanism alleviates WLAN power consumption problem [5], [6] and improve throughput. However, there are still many challenging issues for simultaneous transmission.

At the beginning of a TWT SP, AP has to schedule resources properly by TF to receive PS-Poll frames from STAs for DL traffic or to trigger transmission process for UL traffic. When too many STAs wake up simultaneously, AP involves in complex resource assignment to arrange RUs in frequency dimensionality and TWT SPs in time dimensionality. Some STAs are even failed to receive/transmit frames for lacking of resources or meeting with serious contention during their active beacon slots. As a result, the AP has to discard buffered frames according to the aging function in DL traffic or make UL transmission more asynchronously for their unknown transmission requirements and contention mechanism. Meanwhile, large LIs involve the AP and STAs in buffer overflow.

As an example shown in Fig. 2, STA_1, STA_2 and STA_3 request to enter into doze mode. They send their own TWT request frames to the AP after the AP announces available RUs in a TF. In the received TWT response frames, the AP informs these three STAs to wake up in the same beacon transmission time to receive the same first target beacon. In UL traffic, STA_1, STA_2 and STA_3 which have buffered frames wake up only when a satisfied trigger frame(TF)-enabled TWT SPs are broadcasted by the AP. They wake up at the broadcast TWT start time to transmit data with uplink OFDMA-based random access (UORA) procedure. In the first TF-enabled TWT SP (i.e., TWT SP #1), STA_1 and STA_2 randomly select the same RA-RU marked as RU_0 to transmit when their backoff counters reduce to 0. As a result, an unexpected collision happens which leads to the ineffective utilization on both collided RU_0 and idle RU_1, deteriorated throughput, and waste of power. Besides, a retransmission will be scheduled which results in longer delay. STA_1 and STA_2 switch into doze mode again when TWT SP #1 ends. Clearly, improper wake up time may make the the contention based UORA process inefficient, especially in high dense WLANs.

C. THE CURRENT PROGRESS IN TWT

As IEEE 802.11ax amendment is scheduled for publication in 2020, the TWT mechanism will be determined until 2020. Little work related to TWT has been published. In [23], an adaptive UL OFDMA random access grouping scheme for IEEE 802.11ax WLANs is proposed. The performance of both the system and each STA is improved by coordinating the group size according to the UORA process during the TWT SP. However, the proposed scheme does not consider the scheduling on TBTTs.

Besides, many works on PSM above are performed between one AP and one/many STAs via a single channel. we focus on the strategy of determining tuples of $<f, t>$ for followed broadcast TWT SPs with multiple RUs.

In [24], Naik *et al.* gave a performance analysis of UL MU OFDMA in IEEE 802.11ax based on Markov Chain [25], and derived the relationship between the number of STAs and the throughput in saturated traffic. In other words, the throughput during each TWT SP depends on the the number of active STAs, which is closely related with TWT scheduling. Hence, we will investigate the TWT operations to improve performance in dense scenarios.

III. THE SCHEME OF TSS

As shown in Fig. 3, we consider an IEEE 802.11ax based WLAN with one single AP and n TWT scheduled/requesting STAs, denoted by $\mathbb{S} = \{s_i | i = 1, 2, \ldots, n\}$. We employ the new UORA procedure to transmit/receive data in the MAC layer for UL traffic. Besides, we apply the TSS for the AP to

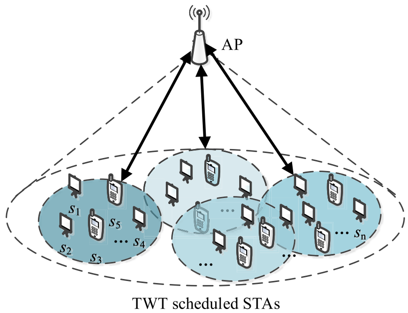


FIGURE 3. An example of application scenario.

make decision on the time when STAs switch into doze mode and when wake up by setting the first TBTTs and LIs. The TSS works as follows:

Step 1 The AP announces available RA-RUs in TFs to gather TWT requests from STAs. Next, n TWT requesting STAs will send request frames with parameter of LIs to the AP simultaneously by using the RA-RUs specified in the TF. We denote the LIs as $\mathbb{T} = \{t_i | i = 1, 2, \ldots, n\}$.

Step 2 After receiving request frames from STAs during a short time, the AP responds with better LIs, denoted by $\mathbb{T}' = \{t'_i | i = 1, 2, \ldots, n\}$, and different first TBTTs, denoted by $\mathbb{F}' = \{f'_i | i = 1, 2, \ldots, n\}$, to the scheduled STAs by employing the LI scheduling procedure proposed in Section III-B, and the first TBTT scheduling procedure in Section IV, respectively. Since the AP is the centralized controller in the IEEE 802.11ax WLAN, it schedules the DL transmissions, including TWT response frames.

Step 3 After receiving their response frames, STAs will switch into doze mode,and wait for their own target beacons. The target beacon transmission times (TBTTs) of each STA are determined by its received parameters of the first TBTT and LI.

Step 4 For UL transmissions, STAs will take part in the broadcast TWTs indicated in the target beacons to transmit data if they have pending frames. After the TWT SP, STAs will switch into doze mode again until their next TBTT or TWT SP begins.

Step 5 In case of buffer overflows or a new STA joins in, the AP sends renewed TWT parameters included in a beacon to notify the STAs to start new doze periods.

Finally, several assumptions are made to simplify the subsequent analysis.

- All the STAs are associated to the AP with equal access priority.
- The PHY layer is assumed to be ideal, while the network is saturated.
- In each TWT SP, the AP utilizes m eligible RUs of the same bandwidth with the same data rate of r.

TABLE 2. List of the main notations.

Notations	Description
n	The number of TWT requesting STAs
m	The number of eligible RA-RUs
$\mathbb{S}$	The set of all TWT requesting STAs
$Sub\mathbb{S}_i$	The i-th subset of $\mathbb{S}$
s_i	The i-th STA where $s_i \in \mathbb{S}$
$\mathbb{T}$	The set of all LIs
$Sub\mathbb{T}_i$	The i-th subset of $\mathbb{T}$
u	The number of subsets of $\mathbb{S}$ or $\mathbb{T}$
f_i	The first TBTT of STA s_i
t_i	The LI of STA s_i
w_i	A TBTT of STA s_i
$[f_i]$, $[f_i]_{t_i}$	All TBTTs of STA s_i
$lst[][]$	Lists used for storing TBTTs
c_i	The list size of $Sub\mathbb{S}_i$
r	Data rate on a RU
L	The constant size of a data frame
T	Beacon interval
τ	TWT SP duration
d_{cl}	Contention level
d_{cv}	Contention variation
d_{cva}	Adjacent contention variation
p_{sb}	The prob. of successful backoff
p_{ru}	The prob. of successfully choosing an idle RA-RU
T_T	The time duration used to transmit one trigger frame
T_B	The time duration used to transmit one beacon frame
T_D	The time duration used to transmit one data frame
T_M	The time duration used to transmit one multiuser block ACK (MBA)
E_t	Power consumption in transmission mode
E_r	Power consumption in reception mode
E_i	Power consumption in idle mode
E_d	Power consumption in doze mode

A. PERFORMANCE METRICS

In this subsection, we introduce three performance metrics. The main notations adopted are listed in Table 2 for convenience of description.

1) CONTENTION LEVEL

In order to analyze the number of active STAs in each beacon slot, we put forward the definition of contention level to describe the congestion (i.e., network density) at each beacon slot.

Definition 1 (Contention level d_{cl}): Contention level is defined as the number of TWT requesting STAs that wake up simultaneously at a TBTT.

$$d_{cl}(k) = \sum_{i=1}^{n} g_i(k). \tag{1}$$

Here, the function of $g_i(k)$ denotes whether the STA s_i will wake up at the k-th beacon slot or not. If $g_i(k) = 1$, it represents s_i will wake up at the k-th beacon slot, while $g_i(k) = 0$ means it stays in doze mode.

We denote the parameters $<f, t>$ of s_i as $<f_i, t_i>$. Both f_i and t_i are multiples of a beacon interval, i.e., f_i and t_i are positive integers. Then, s_i will wake up at beacon slot $f_i, f_i + t_i, f_i + 2t_i, \ldots$, i.e., $f_i + m_i \cdot t_i$, where m_i is a non-negative integer. Hence, the function of $g_i(k)$ can be calculated as follows:

$$g_i(k) = \begin{cases} 1, & \text{if } k \pmod{t_i} = f_i, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where s_i is expected to wake up at the k-th beacon slot, if $k \pmod{t_i} = f_i$.

Definition 2 (Contention variation d_{cv}): Contention variation is defined as the range of contention levels which is used to describe maximum volatility of contention levels during the whole beacon cycle.

$$d_{cv} = \max\{d_{cl}(k) \mid k \in \Omega\} - \min\{d_{cl}(k) \mid k \in \Omega\}, \quad (3)$$

where $\Omega = \{1, 2, \ldots, c\}$, and the notation c represents the least common multiple (LCM) of all LIs. Note that c is also the beacon cycle.

Definition 3 (Adjacent contention variation d_{cva}): Adjacent contention variation is defined as the range of contention levels which is used to describe volatility of contention levels between adjacent beacon slots.

$$d_{cva} = \sqrt{\frac{\sum_{k=1}^{c}(d_{cl}(k) - d_{cl}(k-1))^2}{c}}, \quad (4)$$

where $d_{cl}(0) = 0$.

Definition 4 (Average contention variation $\overline{d_{cv}}$): Average contention variation, used to describe average volatility of contention levels in the whole beacon cycle, is calculated as follows

$$\overline{d_{cv}} = \sqrt{\sum_{k=1}^{c} \frac{(d_{cl}(k) - \overline{d_{cl}})^2}{c}}. \quad (5)$$

Here, $\overline{d_{cl}}$ is the average value of all $d_{cl}(k)$, $k \in \Omega$. Thus, we have $\overline{d_{cl}} = \frac{1}{c}\sum_{k=1}^{c} d_{cl}(k)$. Besides, $\overline{d_{cl}}$ also represents the average number of active STAs in each beacon slot, and it can be calculated as follows

$$\overline{d_{cl}} = \sum_{i=1}^{n} \frac{1}{t_i}. \quad (6)$$

Note also that $\frac{1}{t_i}$ represents the wake frequency of s_i.

2) THROUGHPUT

Since the DL MU OFDMA is based on centralized schedule-based transmission, the throughput is invariant to network capability in the saturated WLAN. Besides, the EE is optimal when only one STA wakes up in a beacon slot while others keep dozing. Hence, we only analyze on the UL performance of the IEEE 802.11ax MAC.

In case of simultaneous UL transmission, the AP first sends a TF to the STAs assigning them corresponding RU grants. Then, STAs may operate in two different TF-based access schemes. The first scheme is Uplink TF-based Multiuser Access (UTMA), in which the STA transmits directly on an allocated RU. The UTMA transmission is also scheduled by the AP so that the throughput and EE are similar with DL MU transmission. The second scheme is the Uplink OFDMA-based Random Access (UORA) mechanism, in which STAs contend for channel access on the assigned set of RUs where a STA selects a paricular RU uniformly. Suppose packet errors occur only when multiple STAs transmit at the same time in the same RU, i.e., the channel condition is ideal. Hence, the successful probability of transmission depends on the number of contending STAs, the number of available RA-RUs and the probability of successful backoff. While the probability of successful backoff depends on the OFDMA contention window (OCW), the number of RA-RUs, and successful probability of choosing an idle RA-RU.

The probability of successful backoff, denoted as p_{sb}, can be obtained from the following equation derived from [24], [26]:

$$p_{sb} = \frac{2}{1 + \frac{W}{m} + (1 - p_{ru})\frac{W}{m}\sum_{j=0}^{bs-1} 2^j (1 - p_{ru})^j}. \quad (7)$$

Here, m denotes the number of RA-RUs, W denotes the minimum OCW which equals minimum OCW + 1.

The notation bs, the maximum number of backoff stage, is got from the following equation,

$$bs = \log_2 \frac{OCW_{max} + 1}{OCW_{min} + 1}, \quad (8)$$

where OCW_{min} and OCW_{max} denote the minimum and maximum OCW size given by the AP, respectively.

The p_{ru}, the successful probability of choosing an idle RA-RU that is not selected by the other STAs, can be calculated as follows,

$$p_{ru} = (1 - \frac{p_{sb}}{m})^{\overline{d_{cl}} - 1}, \quad (9)$$

where $\overline{d_{cl}}$ is the average number of contending users in each TWT SP.

Hence, the average throughput on m RA-RUs is $\overline{d_{cl}} \cdot p_{sb} \cdot p_{ru} \cdot r$, where r denotes the data rate on a RU. According to the procedure of TWT SP #1 in Fig. 2, we have the total effective throughput on m RA-RUs from the following equation,

$$\Theta = \overline{d_{cl}} \cdot p_{sb} \cdot p_{ru} \cdot r \cdot \frac{T_D}{T_T + T_D + T_M}. \quad (10)$$

Here, T_T, T_D and T_M denote the time durations used to transmit one TF, data frame and multi-STA block ACK (MBA) frame, respectively. Note that the inter frame apace (IFS) is ignored here.

3) ENERGY EFFICIENCY (EE)

We only consider the power consumption of the network interface, which depends on the PSM: active or doze mode. The active mode is further sub-divided into the following three states: transmission, reception and idle mode. Assume that $\overline{E}$ is the average energy consumed, E_t, E_r, E_i, and E_d represent transmission mode power, reception mode power, idle mode power, and doze mode power, respectively. P_t, P_r, P_i, and P_d denote the probabilities in transmission, reception, idle, and doze mode. Therefore, the following equation is obtained:

$$\overline{E} = P_t \cdot E_t + P_r \cdot E_r + P_i \cdot E_i + P_d \cdot E_d. \quad (11)$$

The probabilities of transmission, reception, doze, and idle mode are calculated as follows:

$$P_t = \frac{\tau_t}{T}, \tag{12}$$

$$P_r = \frac{\tau_r}{T}, \tag{13}$$

$$P_d = \frac{\tau_d}{T}, \tag{14}$$

$$P_i = 1 - P_r - P_t - P_d. \tag{15}$$

Here, T denotes the beacon interval, and τ_t, τ_r, and τ_d represent the average time spent in transmission, reception and doze mode during a beacon interval, respectively.

The average number of data frames transmitted no matter successfully or not during a TWT SP, denoted by N_t, can be got as follows:

$$N_t = \frac{\Theta \cdot \tau}{p_{ru} L}, \tag{16}$$

where L is the size of a single data frame, and τ represents the duration of a TWT SP which is bigger than the minimum duration (i.e., $256\mu s$ in IEEE 802.11ax [4]) but less than $T - T_B$. Here, the notation T_B represents the time used to transmit a beacon frame.

Then, we can easily get the total time τ_t used to transmit frames as

$$\tau_t = N_t \cdot T_D, \tag{17}$$

and the average time τ_r used to receive frames during a beacon interval is

$$\tau_r = N_t \cdot p_{ru}(T_T + T_M) + N_t \cdot (1 - p_{ru})T_T + \overline{d_{cl}} \cdot T_B. \tag{18}$$

In the equation, $N_t \cdot p_{ru}$ denotes the number of data frames successfully received by the AP. MBA frames will be sent back to the corresponding STAs for these data frames. While $N_t \cdot (1 - p_{ru})$ represents the number of failed transmission so that no MBA frame will be received. Moreover, $\overline{d_{cl}}$ STAs wake up at each TBTT in average to receive the beacon frame.

During a beacon interval, about $(n - \overline{d_{cl}})$ STAs stay in doze mode, and $\overline{d_{cl}}$ STAs do not transmit frames outside the TWT SP τ. We further derive the average time in doze mode, denoted as τ_d, from the following equation,

$$\tau_d = (n - \overline{d_{cl}}) \cdot T + \overline{d_{cl}} \cdot (T - T_B - \tau). \tag{19}$$

Hence, the average average time in idle mode, denoted as τ_i, can be obtained as follows,

$$\tau_i = n \cdot T - \tau_t - \tau_r - \tau_d. \tag{20}$$

Next, we get the average energy consumption based on (11) as

$$\overline{E} = \frac{\tau_t \cdot E_t + \tau_r \cdot E_r + \tau_i \cdot E_i + \tau_d \cdot E_d}{T}. \tag{21}$$

Finally, we have the energy efficiency (EE) formulation as

$$EE = \Theta / \overline{E}. \tag{22}$$

B. THE OPTIMIZATION PROBLEM

The LIs (i.e. wake frequencies) of each STA will greatly impact the active number of STAs in each beacon slot. Hence, we apply the developed equation of (10) to optimize the overall throughput by controlling LIs. Specially speaking, given n, m, OCW_{min}, and OCW_{max}, we are aiming at achieving the maximum throughput by finding optimal $\overline{d_{cl}}$, which is determined by LIs, namely

$$\max \Theta \tag{23}$$

$$w.r.t.\ t'_1, t'_2, \cdots, t'_n$$

$$s.t.\ \overline{d'_{cl}} = \sum_{i=1}^{n} \frac{1}{t'_i}, \tag{23a}$$

$$t'_i \le t'_j, \text{ if } t_i \le t_j, \tag{23b}$$

$$i, j \in 1, 2, \cdots, n, \tag{23c}$$

where t'_i denotes the optimal LI assigned to STA s_i. Now, we explain the constraints in throughput maximization problem in (23). Since the average number of active STAs in a beacon slot greatly impacts the throughput, it is reasonable to get optimal contention level $\overline{d_{cl}}$ to maximize throughput. As in Equation (6), we have $\overline{d_{cl}} = \sum_{i=1}^{n} \frac{1}{t_i}$. Constraint (23a) and (23b) aim to make the assigned LI of each STA to be in proportion to its original LI requested, given optimal $\overline{d_{cl}}$. In other words, a bigger LI will be assigned to a STA, if it requests for a bigger one in its TWT request frame.

Note that p_{sb} in (7) and p_{ru} in (9) are interrelated, we can apply binary search to find the solution of $\overline{d_{cl}}$ for the maximum throughput. Then, we get optimal LI of each STA which is in proportion to its requested LI as follows,

$$t'_i = \max\{\mathbb{R}(t_i \cdot \overline{d_{cl}} / \overline{d'_{cl}}), 1\}, \tag{24}$$

where $\mathbb{R}(\cdot)$ denotes the rounding function to get an integer, and $i \in \Omega$. If the solution of LI is 0, the smallest LI 1 is allocated to the STA.

The changing trend of the total throughput with respect to $\overline{d_{cl}}$ will help us find the optimal $\overline{d_{cl}}$ easily and quickly. However, it is not easy for the AP to schedule the exact number of simultaneously active STAs during all beacon slots. In the next section, we will propose an algorithm on how to arrange the number of active STAs in each beacon slot approaching $\overline{d_{cl}}$ by scheduling first TBTT of each STA. Hence, we introduce the performance metrics like d_{cv}, $\overline{d_{cv}}$ and d_{cva} in Section III-A.1 to explain the deviation between the expected maximum throughput and its actual throughput. That is, higher variance in contention level leads to a deteriorated performance near the maximum throughput.

IV. THE FIRST TBTT SCHEDULING ALGORITHM

In this section, the algorithm of scheduling the first TBTTs to arrange active STAs in each beacon slot is presented. The awake time of STA s_i is a nonnegative residue class of a modulo t_i, i.e., $[f_i]_{t_i} = \{w_i | w_i \equiv f_i \pmod{t_i}\} = \{f_i + m_i \cdot t_i | m_i \in \mathbb{Z}, m_i \ge 0\}$, and the set of all nonnegative integers congruent to a modulo t_i. Note that, $[f_i]_{t_i}$ is a set of TBTTs,

$\mathbb{Z}$ is the ring of integer. For it is cumbersome, we use notation $[f_i]$ to represent $[f_i]_{t_i}$ in the sequel.

Lemma 1: Given $[f_i]$, $[f_j]$, if $GCD(t_i, t_j) = 1$, we have $[f_i] \cap [f_j] \neq \emptyset$, i.e., STA s_i and STA s_j are expected to wake up simultaneously. Here, $GCD(\cdot)$ denotes greatest common divisor.

Proof: Based on Chinese Remainder Theorem (CRT), given $[f_i]$, $[f_j]$, if $GCD(t_i, t_j) = 1$, there exists $x \in [f_i] \cap [f_j]$. Moreover x is unique modulo $t_i \cdot t_j$, i.e., STA s_i and STA s_j with LIs of relatively prime are bound to wake up at the same beacon slot x. □

Lemma 2: Given $[f_i]$, $[f_j]$, if $t_j = \gamma \cdot t_i$, $f_j \notin [f_i]$, then for all elements $w_i \in [f_i]$ and $w_j \in [f_j]$, we have $w_i \neq w_j$,where γ is a positive integer, i.e., STA s_i and STA s_j never wake up simultaneously.

Proof: From the definition of $[f]$, we have $[f_i] = \{w_i | w_i \equiv f_i \pmod{t_i}\} = \{f_i + m_i \cdot t_i | m_i \in \mathbb{Z}, m_i \geq 0\}$, and $[f_j] = \{w_j | w_j \equiv f_j \pmod{t_j}\} = \{f_j + m_j \cdot t_j | m_j \in \mathbb{Z}, m_j \geq 0\} = \{f_j + m_j \cdot \gamma \cdot t_i | m_j \in \mathbb{Z}, m_j \geq 0\} = \{f_j + m_j \cdot \gamma \cdot t_i | m_j \cdot \gamma \in \mathbb{Z}, m_j \cdot \gamma \geq 0\} \subseteq [f_j]_{t_i}$. Hence, $[f_j]_{t_j} \subseteq [f_j]_{t_i}$. Since $f_j \notin [f_i]_{t_i}$, we have $[f_i]_{t_i} \cap [f_j]_{t_i} = \emptyset$. It can be concluded that $[f_i]_{t_i} \cap [f_j]_{t_j} = \emptyset$. Therefore, for all elements $w_i \in [f_i]$ and $w_j \in [f_j]$, we have $w_i \neq w_j$, i.e., STA s_i and STA s_j never wake up simultaneously in any beacon slot. □

Lemma 3: Given $t_j > t_i$, $t_i \nmid t_j$, and $GCD(t_j, t_i) > 1$, then $\exists w_i \in [f_i]$, $w_j \in [f_j]$, s.t. $w_i = w_j$, i.e., STA s_i and STA s_j may wake up simultaneously.

Proof: Let $x = LCM(t_j, t_i)$, $y = GCD(t_j, t_i) = t_j \cdot \frac{t_i}{x}$, and suppose $f_i = f$, $d_i = t_j - t_i > 0$, where $0 \leq f < t_i$, and $LCM(\cdot)$ denotes least common multiple. Define $f_j = f + d_i < t_j$, $m_j = \frac{x}{t_j} - 1 > 0$, and $m_i = \frac{x}{t_i} - 1 > 0$. Then, we have $w_i = f_i + m_i \cdot t_i = f + (\frac{x}{t_i} - 1) \cdot t_i = f - t_i + x = f + t_j - t_i + (x - t_j) = f_j + (\frac{x}{t_j} - 1) \cdot t_j = f_j + m_j \cdot t_j = w_j$.Thus, STA s_i and STA s_j will wake up simultaneously at beacon slot w_i. □

Lemma 4: Given $t_j > t_i$, $t_i \nmid t_j$, and $GCD(t_j, t_i) > 1$, then $\exists [f_i]$, $[f_j]$, $\forall w_i \in [f_i]$, $w_j \in [f_j]$, s.t. $w_i \neq w_j$, i.e., STA s_i and STA s_j will never wake up simultaneously.

Proof: Let $x = LCM(t_j, t_i)$, $y = GCD(t_j, t_i) = t_j \cdot \frac{t_i}{x}$, and suppose $f_i = 0, f_j = 1$. Then in a beacon cycle x, there are $w_i = f_i + m_i \cdot t_i = m_i \cdot t_i$ and $w_j = f_j + m_j \cdot t_j = 1 + m_j \cdot t_j$, where $0 \leq m_i < \frac{x}{t_i}$ and $0 \leq m_j < \frac{x}{t_j}$. Obviously, if $m_i = m_j = 0$, then $w_i \neq w_j$. So we assume that $0 < m_i < \frac{x}{t_i}$ and $0 < m_j < \frac{x}{t_j}$. Clearly, $w_i \pmod{y} = m_i \cdot t_i \pmod{y} = 0$. Since $y = GCD(t_j, t_i) > 1$, we have $w_j \pmod{y} = (m_j \cdot t_j + 1) \pmod{y} = 1$. Then, $w_i \pmod{y} \neq w_j \pmod{y}$, i.e., $w_i \neq w_j$. Therefore, STA s_i and STA s_j will never wake up simultaneously, if $f_i = 0, f_j = 1$. The conclusion is also true if $f_i = d$, $f_j = d + 1$, where $0 \leq d < t_i$. □

Based on the lemmas above, the procedure of determining the first TBTTs of all STAs aiming at minimizing contention variation has three steps, named inter-grouping, intra-grouping and drift-grouping, respectively.

Step 1 *Inter-grouping:* LIs with different properties are divided into different subsets, while STAs with LIs in different subsets fall into different groups accordingly. In light of Lemma 2, for any couples of LIs (t_i, t_j) where $t_i | t_j$, we group them into the same subset of LIs, since their corresponding STAs never wake up simultaneously with different first TBTTs.
On basis of Lemma 1, we divide all the LIs which are relatively primes into two different subsets, since these two corresponding STAs wake up simultaneously at least once during each beacon cycle. According to Lemma. 3, LIs with the properties of $t_j \nmid t_i$ but $GCD(t_j, t_i) > 1$, we divide them into two different subsets, because the corresponding STAs may wake up simultaneously during a beacon cycle.
As a result, STAs with LI in a same subset can be scheduled together to wake up asynchronously all the time.

Step 2 *Intra-grouping:* STAs in a same subset are scheduled by the procedure of intra-grouping. After Step 1, STAs with LIs in multiples are grouped into one subset. Based on Lemma 2, STAs in the same subset never wake up simultaneously under the condition of $t_j > f_j > f_i$ in each beacon cycle. Hence, it is easy to control the number of active STAs in a beacon slot.

Step 3 *Drift-grouping:* According to Lemma 4, STAs in different subsets may wake up asynchronously by setting specific first TBTT. Hence, we make the start index of the last list of each arrangement in Step 2 different from each other via the algorithm of drift-grouping.

Next, we present above steps one by one in detail.

1) INTER-GROUPING ALGORITHM

Algorithm 1 illustrates the procedure of inter-grouping on dividing LIs into multiple subsets. Here, we take input $\mathbb{T} = \{16, 8, 18, 9, 3, 27, 6, 2, 4, 12, 9, 6, 1\}$ for example. It is easy to get a LI set in an ascending order, i.e., $\mathbb{T}_o = \{1, 2, 3, 4, 6, 6, 8, 9, 9, 12, 16, 18, 27\}$, and the distinct LI set $\mathbb{T}_d = \{1, 2, 3, 4, 6, 8, 9, 12, 16, 18, 27\}$. The smallest element $1 \in \mathbb{T}_d$ is 1st to be arranged into a new generated subset named $Sub\mathbb{T}_1$. Later, the 2nd element 2 falls into $Sub\mathbb{T}_1$, since it is a multiple or a divisor of all elements in $Sub\mathbb{T}_1$. And then, the 3rd element 3 falls into $Sub\mathbb{T}_2$ because $2 \nmid 3$, and so on in a similar manner. Division subsets we have got are listed as follows: $Sub\mathbb{T}_1 = \{1, 2, 4, 8, 16\}$, $Sub\mathbb{T}_2 = \{3, 6, 12\}$, $Sub\mathbb{T}_3 = \{9, 18\}$, and $Sub\mathbb{T}_4 = \{27\}$. Finally, STAs with corresponding LIs are divided into four subsets after inter-grouping. We have $Sub\mathbb{S}_1 = \{s_1, s_2, s_8, s_9, s_{13}\}$, $Sub\mathbb{S}_2 = \{s_5, s_7, s_{10}, s_{12}\}$, $Sub\mathbb{S}_3 = \{s_3, s_4, s_{11}\}$, and $Sub\mathbb{S}_4 = \{s_6\}$. Finally, all the STAs in $Sub\mathbb{S}_1$ with LIs in $Sub\mathbb{T}_1$ are scheduled in the same lists with the intra beacon cycle $c_1 = 16$, while STAs with LIs in different subsets, such as $18 \in Sub\mathbb{T}_3$ and $27 \in Sub\mathbb{T}_4$, are scheduled in two different process of intra-grouping.

After the process of inter-grouping, the division subsets of LIs obtained have following properties.

i) Let $Sub\mathbb{T} \subseteq \mathbb{T}$, then $Sub\mathbb{T} \neq \emptyset$.
ii) Let $Sub\mathbb{T}_i, Sub\mathbb{T}_j \subseteq \mathbb{T}$, then $Sub\mathbb{T}_i \cap Sub\mathbb{T}_j = \emptyset$.

Algorithm 1 Inter-Grouping Algorithm

Input: $\mathbb{T} = \{t_1, t_2, \ldots, t_n\}$, $\mathbb{S} = \{s_1, s_2, \ldots, s_n\}$
Output: Subsets of $\mathbb{T}$: $Sub\mathbb{T}_1, Sub\mathbb{T}_2, \ldots, Sub\mathbb{T}_u$, and subsets of $\mathbb{S}$: $Sub\mathbb{S}_1, Sub\mathbb{S}_2, \ldots, Sub\mathbb{S}_u$

$\mathbb{T}_d \leftarrow \{\text{distinct } t_i | t_i \in \mathbb{T}\}$, $\mathbb{T}_o \leftarrow$ ascending $\mathbb{T}_d$, $u \leftarrow 0$;
while $\mathbb{T}_o \neq \varnothing$ **do**
  $t \leftarrow$ the first element in $\mathbb{T}_o$;
  for $i = 1 : u$ **do**
    $t_j \leftarrow$ the last element in $Sub\mathbb{T}_i$;
    if $t_j | t$ **then**
      $Sub\mathbb{T} \leftarrow Sub\mathbb{T}_i$;
      break ;
    end
  end
  if $Sub\mathbb{T} == \varnothing$ **then**
    $u \leftarrow u + 1$;
  end
  $Sub\mathbb{T}_i \leftarrow Sub\mathbb{T} \cup \{t\}$;
  $\mathbb{T}_o \leftarrow \mathbb{T}_o - \{t\}$;
  $Sub\mathbb{T}_i \leftarrow \varnothing$;
end
Split all STA $s_i \in \mathbb{S}$ into $Sub\mathbb{S}_k$, if $t_i \in Sub\mathbb{T}_k$;

iii) Let $Sub\mathbb{T} \subseteq \mathbb{T}$, then $LCM\{t_i | t_i \in Sub\mathbb{T}\} = max\{t_i | t_i \in Sub\mathbb{T}\}$, i.e., the maximum value of each subset is its intra beacon cycle.
iv) Let $t_i \in Sub\mathbb{T} \subseteq \mathbb{T}$, if t_i is a prime, then $t_i = min\{t_i | t_i \in Sub\mathbb{T}\}$.
v) Let $t_i, t_j \in Sub\mathbb{T} \subseteq \mathbb{T}$, if $j > i$, then $t_j > t_i$.
vi) Let $t_i, t_j \in Sub\mathbb{T} \subseteq \mathbb{T}$, if $t_j > t_i$, then $t_i \mid t_j$.

Proposition 1: The division result of inter-grouping procedure is optimal, depicting by the number of subsets, which follow the above six properties.

Proof: Assume, to get a contradiction, that there are no subsets can be separated and grouped into other subsets. $\forall A \subseteq \mathbb{T}, B \subseteq \mathbb{T}$, supposing B can be resolved, let $B \rightarrow B'$, $A \rightarrow A'$, and b_r is the last element that will fall into subset A' after many steps. $A' = \{a_i | i = 1, 2, \ldots .n\}$, $B' = \{b_r'\}$, $a_1' < b_r \in B' < a_d'$ and u is the number of subsets of T. Clearly, $B' = B' - \{b_r\} = \varnothing$, $A' = A' \cup \{b_r\}$, $u' = u - 1$, and $b_r = m_1 \cdot a_1 = m_2 \cdot a_2 = \ldots = m_{d-1} \cdot a_{d-1}$. According the process of inter-grouping, b_r is divided into A before a_{d+1}, which means the subset B never exists. □

2) INTRA-GROUPING ALGORITHM

After getting subset division result, we schedule the exact TBTTs of each STA in each subset via Algorithm 2. We take LIs in the k-th subset $\{2^i | i \in \mathbb{Z}, i \geq 0\}$ as an example, which is derived from [17], i.e., the LIs used in [17] is a specific subset of $\mathbb{T}$ in this paper. For example, LIs requested by 11 STAs are 4, 8, 4, 2, 4, 8, 4, 4, 8, 4, and 16. As shown in Fig. 4, we draw lists to illustrate our scheduling. The horizontal line represents the time line in beacon interval. The STAs which occupy the 1st column will wake up at the 1st beacon transmission time, i.e, STAs s_4, s_5, and s_2 in the 1st column are scheduled to wake up at the 1st TBTT. STAs s_1, s_7, and s_6 in the 2nd column will wake up at the 2nd TBTT, and so on.

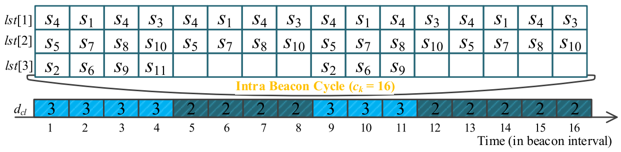


FIGURE 4. An example for TWT intra-grouping ($c_k = 16$), where $n = 11$, $t_4 = 2$, $t_1 = t_3 = t_5 = t_7 = t_8 = t_{10} = 4$, $t_2 = t_6 = t_9 = 8$, and $t_{11} = 16$.

According to the Algorithm 2, the 1st vacant list ($lst[1]$) generated takes intra beacon cycle (max value in $Sub\mathbb{T}_k$, i.e., $c_k = 16$) as its size. The STA with the smallest LI is first to be handled to occupy the 1st vacant unit to wake up at corresponding TBTT. Hence, STA s_4 which has the smallest LI of 2 is scheduled to wake up at TBTTs of 1, 3, 5, ..., 15, and it occupies the columns marked 1, 3, 5, ..., 15 in the 1st list. Next, STA s_1 with LI of 4 is arranged. If $lst[1]$ is full, a new list ($lst[2]$) will be generated. The remaining STAs are arranged in order in the same way. The figure shows the final schedule result.

After scheduling, the STAs that wake up in the same slot content for limited resources so that frequency division (FD) and time division (TD) hybrid TWT SP scheduling algorithms can be applied to arrange an appropriate MU transmission.

Algorithm 2 Intra-Grouping Algorithm

Input: $Sub\mathbb{T}_k$, $Sub\mathbb{S}_k$
Output: First TBTTs $f_1, f_2, \ldots$ for all STAs in $Sub\mathbb{S}_k$

$ln \leftarrow 1$, $c_k \leftarrow$ the greatest number in $Sub\mathbb{T}_k$;
$lst[ln][c_k] \leftarrow \varnothing$;
for $i = 1 : length(SubS_k)$ **do**
  Find the first vacant unit f_i in $lst[ln][c_k]$;
  $lst[ln][j] \leftarrow i$-th element in $Sub\mathbb{T}_k$, for all $j == f_i + m_i * t_i$, where $j \leq c_k$, and m_i is a nonnegative integer;
  if $lst[ln][c_k]$ *is full and* $i \neq length(Sub\mathbb{S}_k)$ **then**
    $ln \leftarrow ln + 1$;
    Generate a new vacant list $lst[ln][c_k]$;
  end
end

Proposition 2: Intra-grouping procedure is optimal in the terms of contention variation of each subset.

Proof: Since the vacant unit appears only in the last list after the procedure of intra-grouping, it can be easily concluded that the average contention level $\overline{d_{cl}} = \sum_{i=1}^{n} \frac{1}{t_i}$. Thus, the maximum contention level $\hat{d_{cl}}$ is $\lceil \sum_{i=1}^{n} \frac{1}{t_i} \rceil$ and the minimum contention level $\check{d_{cl}}$ is $\lfloor \sum_{i=1}^{n} \frac{1}{t_i} \rfloor$. And we have contention variation $d_{cv} = \hat{d_{cl}} - \check{d_{cl}} \leq 1$. Hence, the intra-grouping is optimal in terms of contention variation. □

Fig. 5 shows an example of combination both inter-grouping and intra-grouping to explain the contention level.

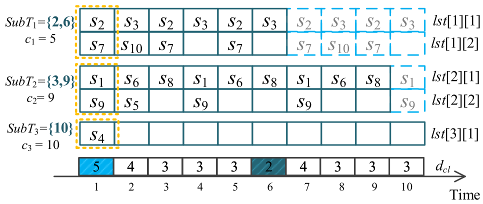


FIGURE 5. An example for TWT inter-grouping and intra-grouping, where $u = 3, c_1 = 6, c_1 = 9, c_3 = 10, n = 10, t_2 = t_3 = t_7 = 2, t_1 = t_6 = t_8 = t_9 = 3, t_{10} = 6, t_5 = 9$, and $t_4 = 10$.

It is observed that the maximum contention level $\hat{d_{cl}} = 5 \leq \sum_{i=1}^{u} \hat{d_{cl}}^{(i)}$, the minimum contention level $\check{d_{cl}} = 2 \geq \sum_{i=1}^{u} \check{d_{cl}}^{(i)}$, and contention variation $d_{cv} = 3 \leq \sum_{i=1}^{u} d_{cv}^{(i)}$ in the schedule. Here, u denotes the number of division subsets of $\mathbb{S}$, and the superscripts i of $\hat{d_{cl}}^{(i)}$, $\check{d_{cl}}^{(i)}$, and $d_{cv}^{(i)}$ are all represent the i-th subset of $\mathbb{S}$. Clearly, the combination of inter-grouping and intra-grouping makes the contention variation more serious than individual intra-grouping so that drift-grouping is proposed consequently.

3) DRIFT-GROUPING ALGORITHM

The smallest element in each LI subset is first to be scheduled in the process of intra-grouping, and collisions caused by the STAs from other subsets are first put forward. In Fig. 5, the LIs of s_1 and s_2 are both the smallest element in their subsets which are scheduled to wake up first at TBTT 1. Hence, collisions caused from s_1 and s_2 first appear. Consequently, the maximum contention level appears in the 1st TBTT of each list after intra-grouping. Therefore, different initial indexes for the last lists of intra-grouping on different subsets, named drift-grouping, can be applied to generate different contention level distribution.

Only the initial index for the last list of each inter-group with vacant units needs to be drifted, since the foregoing list is full with STAs. Figs. 5 and 6 illustrate the difference between the schedule result before and after drift-grouping, respectively. In Fig. 6, the new maximum contention level $\hat{d_{cl}}' = 4$, while the new minimum contention level $\check{d_{cl}}' = 2$. Hence, the contention variation $d'_{cv} = 2 < d_{cv}$, which means the procedure of drift-grouping can be applied to alleviate the contention variation.

Obviously, drift-grouping keeps the same intra beacon cycle, but make the number of simultaneously active STAs more evenly distributed comparing with the situation in Fig. 5. Thus, contention variation is decreased and volatility of resource requirements is reduced. However, it is complicated to find the optimal drift index in time. Therefore, we employ a random methodology for drift-grouping to achieve time efficiency.

After above three steps in the first TBTT scheduling scheme, we have $\hat{d_{cl}} \leq \lfloor \overline{d_{cl}} \rfloor + u$, $\check{d_{cl}} \geq \max\{\lfloor \overline{d_{cl}} \rfloor - u + 1, 0\}$, and $d_{cv} \leq 2u - 1$, where $\hat{d_{cl}}$ and $\check{d_{cl}}$ represent the maximum and minimum contention level, respectively. Clearly, the number of division subset in the inter-grouping algorithm impacts the contention variation.

TABLE 3. List of the main parameter settings.

Parameter	Value
Channel Width	20 MHz
Multi-User Capabilities	OFDMA
RU size	26-tone
MPDU size	2000 Bytes
Data rate r	11.8 Mbps
Beacon Interval T	100 ms
T_B	100 μs
T_T	100 μs
T_M	40 μs
OCW$_{min}$	7
OCW$_{max}$	31
TWT mode	Trigger-enabled unannounced TWT
TWT SP τ	30 ms
Transmission power	1000 mW
Reception power	600 mW
Idle mode power	300 mW
Doze mode power	150 mW

V. PERFORMANCE EVALUATION

In this section, we first use simulation programs written in MATLAB to illustrate the accuracy of the analytical formulations on throughput and energy efficiency (EE). The performance of the proposed first TBTT scheduling algorithm is also evaluated in terms of contention level and contention variation. Furthermore, we compare the schemes of first come first service (FCFS), random (RND), EPCS [17], and TSS in terms of: overall throughput (Θ), and EE, and analyze the manage overheads.

A. SIMULATION SETUP

We suppose that transmitted frames will be successfully received, which means the PHY channel is ideal. Furthermore, it is assumed that, as a high-density network, each device always has a packet available for sending, which means it is working in a saturated area. The values of the parameters used to implement both the analytical formulations and the simulations are listed in Table 3.

The impact factors on throughput include the number of STA n, the number of RU-RUs m, the mean of LIs $\bar{t}$, the OCW$_{min}$, and the OCW$_{max}$. Besides, the proposed TSS is compared with the following schemes.

- TSS(without LI scheduling) scheme. The AP accepts all the LIs requested by the STAs, while the first TBTTs are scheduled by the proposed TSS scheme.
- RND scheme. The LIs requested by the STAs are all accepted, while the first TBTT suggested by the AP is randomly selected between 1 and its corresponding LI.
- FCFS scheme. The first TBTT of each STA is assigned with the same value of 1, since all the STAs request to doze during a short time.
- EPCS scheme. In [17], C. Gan and Y. Lin proposed a power conservation scheme (EPCS) to optimally schedule the awake times to minimize the possibility of PS-poll contention by using specified LIs and optimal first TBTTs.

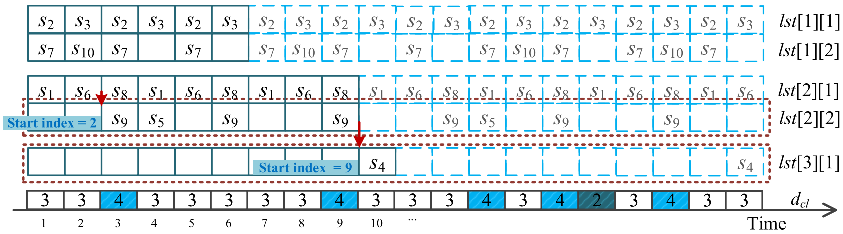
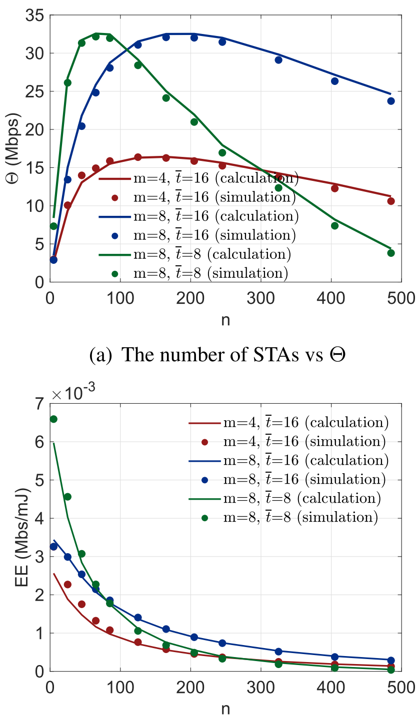


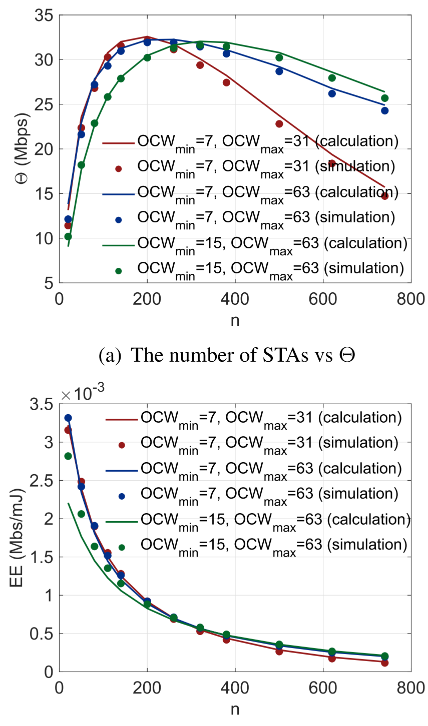
FIGURE 6. An example of the first TBTT scheduling after drift-grouping, where $u = 3, c_1 = 6, c_2 = 9, c_3 = 10, n = 10, t_2 = t_3 = t_7 = 2, t_1 = t_6 = t_8 = t_9 = 3, t_{10} = 6, t_5 = 9$, and $t_4 = 10$.



(a) The number of STAs vs Θ

(b) The number of STAs vs EE

FIGURE 7. Comparison on throughput and EE, where $OCW_{min} = 7$, $OCW_{max} = 31$ in default, and LIs follow the normal distribution of $N(\bar{t}, 3)$.



(a) The number of STAs vs Θ

(b) The number of STAs vs EE

FIGURE 8. Comparison on throughput and EE, where $m = 8, \bar{t} = 16$ in default, and LIs follow the normal distribution of $N(\bar{t}, 3)$.

In the sequel, we denote TSS #1 as the TSS scheme without LI scheduling, and denote TSS #2 as the TSS scheme with LI scheduling.

B. THE ACCURACY OF MODELING

To confirm the accuracy of the derived formulations, the simulation attempts to imitate, as closely as possible, the actual transmission procedure based on UORA of each STA. The LIs used in the simulation follow the normal distribution of $N(\bar{t}, 3)$. Note that $\bar{t}$ and 3 are the mean and variance of the normal distribution, respectively.

As shown in Figs. 7 and 8, the analytical formulations are compared to the simulation results to validate the accuracy of the derived formulations. The figures presents the validation for the performances of throughput and EE when compared to the simulation results. As shown, the values of the analytical prediction results employing the formulations of (10) and (22) are accurate and very close to those of the simulation results. Furthermore, we perform another simulation in which the LIs follow the uniform distribution with the mean value of $\bar{t} = 16$ in default. We find that the simulation results are very similar to Fig. 7 and 8, so the figures and analytical details are omitted here.

In Figs. 7 and 8, the horizontal axis represents the number of STAs n, and the vertical axis represents the corresponding metrics. We can observe that the number of STAs is increased, throughput begins to be increased. However, the throughput starts decreasing when n continues to increase. It is because there is a increase in the channel utilization ratio with the increasing n at first, but the continuous increase in number of STAs leads to increasing collisions and channel failures when the number of STAs exceeds the peak, resulting in a decrease in throughput. Meanwhile, overall EE decreases as the number of STAs increases.

The values of the performance have been tested when $m = 4, 8$ and $\bar{t} = 8, 16$. Fig. 7 shows that the performance metrics are affected by the values of m and $\bar{t}$. By inspecting in Fig. 7(a), we can see that the more available RA-RUs the more number of STAs it can serve, and thus higher throughput is achieved. Besides, it is the same situation as $\bar{t}$ changes. Because the average number of STAs contending on each RU decreases when $\bar{t}$ increases. In Fig. 7(b), we can observe that EE decreases with the increasing number of STAs for the overall power consumption in the network changes much faster than throughput.

In Fig. 8, we realize that the maximum throughput is approximately the same no matter what the parameters of OCW_{min} and OCW_{max} are. Besides, the bigger value of OCW size, the more STAs the AP can serve. This means that the more number of STAs can compensate the larger backoff counter so that higher throughput is achieved.

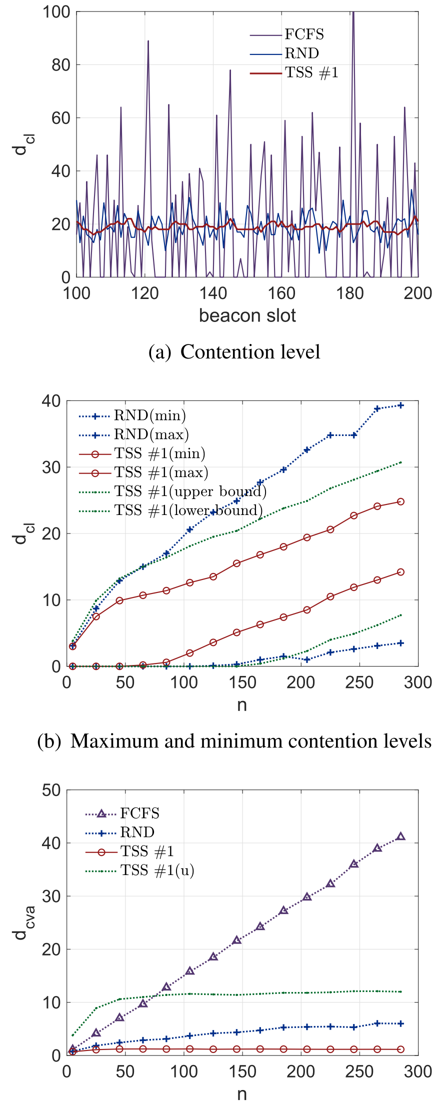
C. CONTENTION LEVEL ANALYSIS

In order to analyze the number of active STAs which has a significant impact on throughput, we compare the difference between FCFS, RND, and TSS #1 for first TBTT scheduling in terms of contention level. In FCFS, the first TBTT depends on when the STA requests to save power, while in RND scheme the first TBTT is set randomly.

As shown in Fig. 9(a), the vertical axis represents the number of STAs wake at a same TBTT, and horizontal axis represents the time in beacon interval. We can see that the number of active STAs, i.e., d_{cl}, fluctuate greatly between neighboring beacon slots in FCFS, while d_{cl} keeps relatively stable in TSS #1. As large variance in contention level leads to a deteriorated throughput near the peak, it is reasonable to use first TBTT scheduling scheme in TSS to control the number of active STAs in a sound range to achieve optimal throughput during each beacon slot.

In Fig. 9(b), the notation of TSS #1 (upper bound) in the figure denotes the upper bound of contention level of the TSS #1 scheme, while the TSS #1 (lower bound) denotes the lower bound. It is observed that the average contention level increases with the increase of the number of STAs. The contention variance d_{cv} (i.e., $\check{d}_{cl} - \hat{d}_{cl}$) of TSS #1 is minimal since its $\hat{d}_{cl}$ is minimal and $\check{d}_{cl}$ is maximal. In Fig. 9(c), TSS #1 (u) represents the number of LI subsets. From the figure, we find that contention variance d_{cv} and adjacent contention variance d_{cva} of TSS #1 are much better than FCFS and RND schemes when the number of STAs becomes much bigger than the number of LI values. In other words, the RU requirement changes a lot in FCFS and RND. This leads to severe collision in some slots, while some RUs may be unused in other slots. Thus, the resource scheduling becomes inefficient in the TWT SP during the whole beacon cycle.

We also investigate resource requirement in terms of LI range using $n = 100$, $m = 8$, $OCW_{min} = 7$, $OCW_{max} = 31$, and LIs follow normal distribution of $N(\bar{t}, 3)$. As illustrated in Fig. 10(b), it is observed that the TSS #1 is approaching RND and has little strength with the increase on LI range.



(a) Contention level

(b) Maximum and minimum contention levels

(c) Adjacent contention variation

FIGURE 9. Comparison between schemes of FCFS, RND and TSS #1, where $m = 8$, $\bar{t} = 16$, and LIs follow normal distribution of $N(16, 3)$.

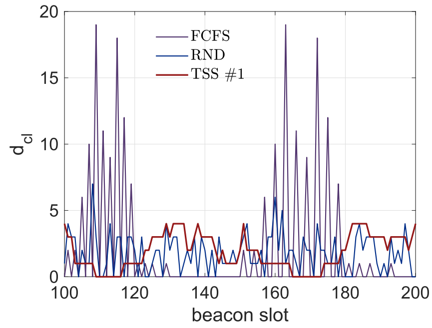
This is because the number of LI subsets generated by Algorithm 1 becomes closer to the number of STAs so that drift-grouping plays a great role in the whole procedure of the first TBTT scheduling. Namely, it indicates that RND becomes a specific case of TSS #1 when the range of LIs is much bigger than the number of STAs.

Finally, we perform another simulation that LIs are uniformly distributed in range of $[1, 31]$. It also shows TSS #1 outperforms FCFS and RND in terms of contention level. Simulation results are similar with Fig. 9, and they are omitted here.

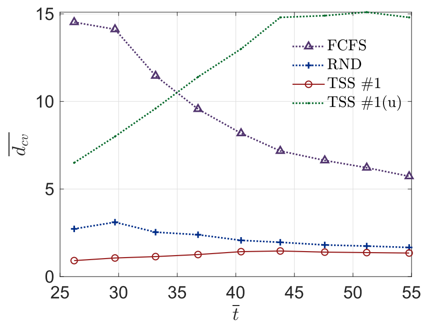
D. PERFORMANCE COMPARISON

In this part, we evaluate throughput and EE between FCFS, RND, EPCS, and TSS schemes.

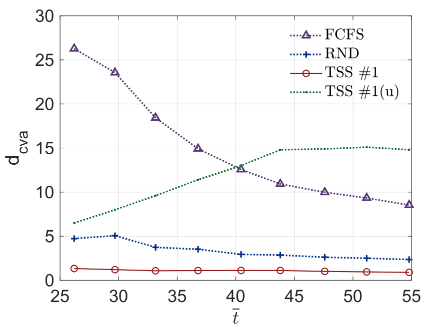
Figs. 11(a) illustrates the effect of n on throughput. The throughput of FCFS, RND, EPCS, and TSS #1 starts to



(a) Contention level ($\bar{t} = 55$)

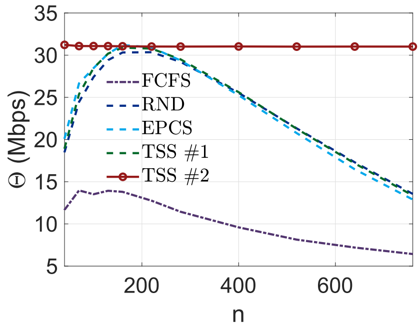


(b) Contention Variation

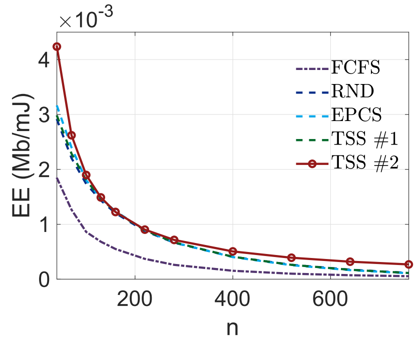


(c) Adjacent contention variation

FIGURE 10. **Comparison between schemes of FCFS, RND and TSS #1, where $n = 100$, $m = 8$, and LIs follow normal distribution of $N(\bar{t}, 3)$.**



(a) The number of STAs vs Θ



(b) The number of STAs vs EE

FIGURE 11. **Comparison between FCFS, RND, EPCS, and TSS, where $\bar{t} = 16$, $m = 8$, $OCW_{min} = 7$, and $OCW_{max} = 31$ in default.**

decrease due to severe collisions and retransmissions when the throughput exceeds the peak value, while it almost keeps the maximum throughput in TSS #2. It also can be observed that EPCS, TSS #1 and RND has a great strength on throughput than FCFS since the active STAs in each beacon slot are more evenly allocated, while in FCFS, the number of active STAs are very different from slot to slot which result in inefficient utilization on RA-RUs.

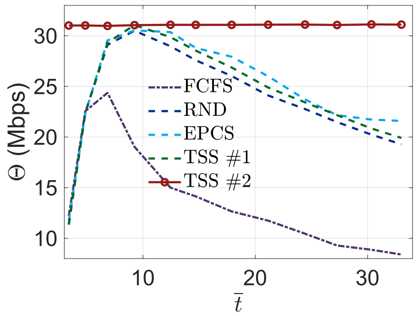
Besides, the throughput of TSS #1 is approaching RND, and has a little strength for the LIs used in the simulation mostly are not multiples. If we arrange LIs in (24) in the same LI sub-group, TSS outperforms RND in both throughput and EE greatly. In the other hand, EPCS has a little strength than TSS #1. The reason is that the LIs assigned to STAs in EPCS are all in one subset, i.e., the LIS assigned to each STA are multiples. The contention variance is minimal after first TBTT scheduling, and a higher throughput is achieved. In the other words, we have to refer to contention level and contention variance to achieve higher throughput.

From Fig. 11(b), it is observed that EE is decreasing with the increase of n for all schemes. The major reason is that the total power consumption increases when operating with more STAs. Furthermore, TSS #2 outperforms other schemes in EE when n exceeds 200.

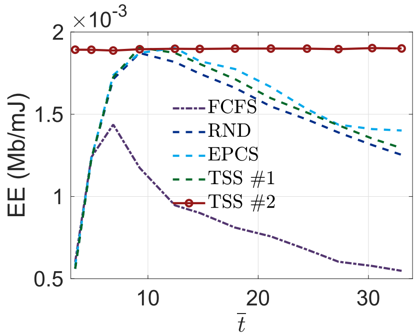
In Fig. 12(a), the throughput starts to decrease when the mean value of LIs exceeds 10 , because the number of active STAs in each beacon slot decreases with the increasing of LIs which results in RA-RUs are not fully used. In addition, the throughput of TSS #1 is worse than EPCS. The reason is that the number of active STAs in each slot is more evenly distributed in EPCS, and less collisions lead to better throughput in some slots. By contrary, TSS #1 outperforms FCFS and RND on throughput. From Fig. 12(b), the trends of EE are very similar with the throughput. As more energy will be saved when LI becomes bigger, the energy consumed in FCFS, RND, EPCS, and TSS #1 decreases. However, throughput becomes the key factor impacts EE for it changes greatly. While in TSS #2, LIs are coordinated to get better throughput which leads to stable power consumption. Hence, the EE keeps stable in TSS #2.

E. KEY PARAMETERS AND OVERHEADS ANALYSIS

From observations on the performance between TSS #1 and #2, we can find that the choice of LIs has a great

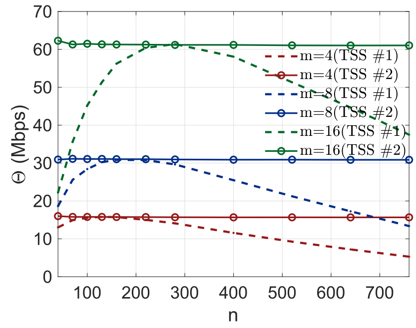


(a) Mean value of LIs vs Θ

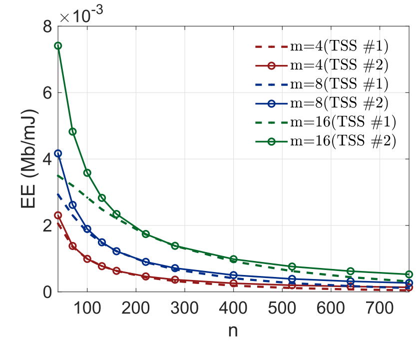


(b) Mean value of LIs vs EE

FIGURE 12. Comparison between FCFS, RND, EPCS, and TSS, where $n = 100$, $m = 8$, $OCW_{min} = 7$, and $OCW_{max} = 31$ in default.



(a) The number of STAs vs Θ



(b) The number of STAs vs EE

FIGURE 13. The impact of n with different number of RA-RUs, where $\bar{t} = 16$, $OCW_{min} = 7$, and $OCW_{max} = 31$.

impact on throughput, and it is reasonable to get better LIs to serve as many STAs as possible according to network capability. Based on the media access mechanism of UORA, we use an optimal number of active STAs to get better LIs. We demonstrate the performance of the LI scheduling algorithm in Figs. 13 and 14. By analyzing the impact of m and OCW, the throughput is greatly improved in TSS #2, while keeping power consumption being efficiently used no matter how many STAs compete in the network.

1) THE IMPACT OF m

Fig. 13 depicts the performance metrics in terms of n against m. The dashed line represents TSS #1, while the solid line represents TSS #2. We observe that TSS #2 outperforms TSS #1 greatly on throughput in Fig. 13(a). Besides, the more available RA-RUs the higher throughput it can achieve. From Fig. 13(b), EE in TSS #2 is a little more efficient than TSS #1 when n exceeds about 200. The reason is that the average wake interval of each STA increases so that the power consumption decreases. In addition, EE is improved with higher m, because more available RA-RUs means more chance for STAs to transmit and less probability of retransmissions.

2) THE IMPACT OF CONTENTION WINDOW

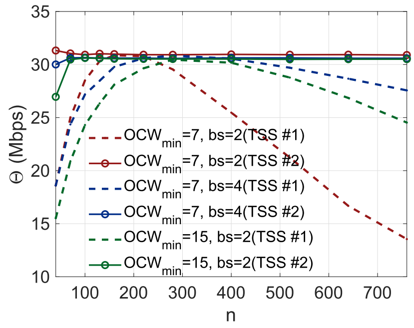
Fig. 14(a) plots the throughput and Fig. 14(b) plots the EE with different contention window parameters, where abscissa represents the number of STA n. OCW_{min} is the minimum contention window, smaller OCW_{min} means a less backoff time. When n is small, a less backoff time gives more chance to transmit so that throughput is better. However, as shown in Fig. 14(a), TSS #1 with the bigger OCW_{min} starts to outperform the smaller one since STAs collide less frequently when n is large. In the other hand, OCW_{min} and OCW_{max} mostly do not have impact on the throughput of TSS #2 since the throughput is optimized based on the OCW_{min} and OCW_{max}. Finally, we observe that EE is improved a little in highly dense WLAN from Fig. 14(b).

3) ANALYSIS ON MANAGEMENT OVERHEADS

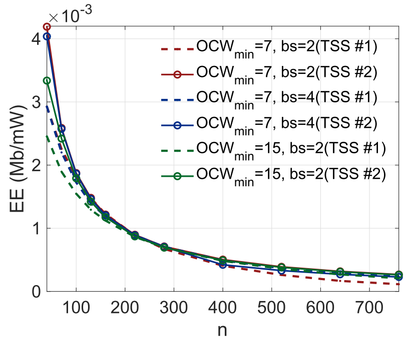
The setup phase of a TWT session requires the exchange of several messages. In [8], Nurchis and Bellatra evaluated the management overheads of both individual and broadcast TWT agreements. TWT management overhead only implies a small fraction of channel airtime even if management packets are transmitted at the lowest available transmission rate, roughly represent about 6 percent of the airtime.

In TSS, we employed the TWT operations directly from IEEE 802.11ax amendment which is backward compatible with legacy WLANs. Besides, we do not need to change the frame structure in the TSS. That is, we do not introduce any other management overheads.

When the AP receives the request frames from STAs, it utilize TSS to calculate the benefitting parameters of first TBTTs and LIs for each STA, which occupies extra computing resource. The time complexity is mainly attributed to: 1) finding the optimal value of p_{sb} and p_{ru} to get the maximum throughput, and 2) the first TBTT scheduling procedure. The former one depending on the throughput precision is



(a) The number of STAs vs Θ



(b) The number of STAs vs EE

FIGURE 14. The impact of n on the performance metrics with various OCW_{min} and OCW_{max}, where $m = 8$ and $\bar{t} = 16$.

$O(lg_2(\frac{1}{a}))$, and the latter one is $O(n)$. Here, a is the precision. However, the STAs do not need to add any computing resources or additional hardware.

A preliminary version of the work was reported in [27].

VI. CONCLUSION

Power saving is a critical issue for portable devices to have a longer run-time. Considering the new technologies introduced in IEEE 802.11ax, we have investigated the broadcast TWT operation with OFDMA based multiuser transmission. Besides, we have proposed a TWT based sleep/wake-up time scheduling scheme known as TSS for the AP to improve overall throughput. It is illustrated the throughput greatly depends on the number of active STAs in each beacon slot. The proposed TSS scheme enables the number of active STAs to be uniformly distributed and optimal by setting the the offset (i.e., first TBTTs) and listening intervals. The proposed TSS scheme makes a practical step towards a collision-free and deterministic access in future WLANs when available resource units are no less than the awake stations in each TWT service period or cooperating with TWT service period scheduling. In the future, QoS requirements like delay, different scenarios such as overlapping WLANs, and TWT service period scheduling will be investigated.

REFERENCES

[1] *Enhancements for Very High Throughput for Operation in Bands Below 6 GHz*, IEEE Standard 802.11ac: Amendment to IEEE Standard 802.11, IEEE Computer Society LAN MAN Standards Committee, Dec. 2016, pp. 1–3534.

[2] Cisco Ltd. (2017). *Cisco Visual Networking Index: Forecast and Methodology, 2015–2020*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

[3] Cisco Ltd. (2017). *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[4] *IEEE Proposed TGax Draft Specification*, IEEE Standard P802.11ax/D3.0, Jun. 2018.

[5] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on IEEE 802.11ax high efficiency WLANs," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 197–216, 1st Quart., 2018.

[6] D.-J. Deng, Y.-P. Lin, X. Yang, J. Zhu, Y.-B. Li, J. Luo, and K.-C. Che, "IEEE 802.11ax: Highly efficient WLANs for intelligent information infrastructure," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 52–59, Dec. 2017.

[7] Q. Qu, B. Li, M. Yang, Z. Yan, A. Yang, J. Yu, M. Gan, Y. Li, X. Yang, O. Aboul-Magd, E. Au, D.-J. Deng, and K.-C. Chen, "Survey and performance evaluation of the upcoming next generation WLAN standard—IEEE 802.11ax," Jun. 2018, *arXiv:1806.05908*. [Online]. Available: https://arxiv.org/abs/1806.05908

[8] M. Nurchis and B. Bellalta, "Target wake time: Scheduled access in IEEE 802.11ax WLANs," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 142–150, Mar. 2019.

[9] S.-L. Tsao and C.-H. Huang, "A survey of energy efficient MAC protocols for IEEE 802.11 WLAN," *Comput. Commun.*, vol. 34, no. 1, pp. 54–67, Jan. 2011.

[10] M. F. Tuysuz, "Towards providing optimal energy—Efficiency and throughput for IEEE 802.11 WLANs," *Int. J. Commun. Syst.*, vol. 31, no. 13, pp. 3597–3621, Jun. 2018.

[11] *IEEE Computer Society LAN MAN Standards Committee, Sub 1 GHz License Exempt Operation*, IEEE Standard 802.11ah: Amendment to IEEE Standards 802.11, May 2017, pp. 1–594.

[12] S.-J. Luan, "Power mangement strategy for IEEE 802.11 based wireless local area networks," Ph.D. dissertation, School Comput. Sci. Technol., Zhejiang Univ. Technol., Zhejiang, China, 2015.

[13] Y.-C. Kuo and C.-N. Chen, "CRT-MAC: A power-saving multicast protocol in the asynchronous ad hoc networks," in *Proc. IEEE Int. Conf. Sensor Netw., Ubiquitous Trustworthy Comput.*, Jun. 2008, pp. 332–337.

[14] Y.-H. Zhu and V. C. M. Leung, "Efficient power management for infrastructure IEEE 802.11 WLANs," *IEEE Trans. Wireless Commun.*, vol. 9, no. 7, pp. 2196–2205, Jul. 2010.

[15] A. J. Pyles, X. Qi, G. Zhou, M. Keally, and X. Liu, "SAPSM: Smart adaptive 802.11 PSM for smartphones," in *Proc. ACM Conf. Ubiquitous Comput. (UbiComp)*, New York, NY, USA, 2012, pp. 11–20, doi: 10.1145/2370216.2370219.

[16] A. Saeed and M. Kolberg, "Towards optimizing WLANs power saving: Novel context-aware network traffic classification based on a machine learning approach," *IEEE Access*, vol. 7, pp. 3122–3135, Dec. 2018.

[17] C. H. Gan and Y. B. Lin, "An effective power conservation scheme for IEEE 802.11 wireless networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1920–1929, May 2009.

[18] R. P. Liu, G. J. Sutton, and I. B. Collings, "WLAN power save with offset listen interval for machine-to-machine communications," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2552–2562, May 2014.

[19] X. Chen, S. Jin, and D. Qiao, "M-PSM: Mobility-aware power save mode for IEEE 802.11 WLANs," in *Proc. Int. Conf. Distrib. Comput. Syst.*, Jun. 2011, pp. 77–86.

[20] Y.-H. Zhu, H.-C. Lu, and V. C. M. Leung, "Access point buffer management for power saving in IEEE 802.11 WLANs," *IEEE Trans. Netw. Service Manage.*, vol. 9, no. 4, pp. 437–486, Dec. 2012.

[21] Y.-H. Zhu, S.-J. Luan, V. C. M. Leung, and K.-K. Chi, "Enhancing timer-based power management to support delay-intolerant uplink traffic in infrastructure IEEE 802.11 WLANs," *IEEE Trans. Veh. Technol.*, vol. 64, no. 1, pp. 386–399, Jan. 2015.

[22] P.-B. Si, H. Ji, F. R. Yu, and G.-X. Yue, "IEEE 802.11 DCF PSM model and a novel downlink access scheme," in *Proc. Wireless Commun. Netw. Conf.*, Apr. 2008, pp. 1397–1401.

[23] J. Bai, H. Fang, J. Suh, O. Aboul-Magd, E. Au, and X. Wang, "Adaptive uplink OFDMA random access grouping scheme for ultra-dense networks in IEEE 802.11ax," in *Proc. IEEE/CIC Int. Conf. Commun. China*, Aug. 2018, pp. 34–39.

[24] G. Naik, S. Bhattarai, and J.-M. J. Park, "Performance analysis of uplink multi-user OFDMA in IEEE 802.11ax," in *Proc. IEEE Int. Conf. Commun.*, May 2018, pp. 1–6.
[25] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.
[26] T. Uwai, T. Miyamoto, Y. Nagao, L. Lanante, Jr., M. Kurosaki, and H. Ochi, "Adaptive backoff mechanism for OFDMA random access with finite service period in IEEE 802.11ax," in *Proc. IEEE Conf. Standards Commun. Netw.*, Nov. 2016, pp. 1–6.
[27] Q. Chen, G. Liang, and Z. Weng, "A target wake time based power conservation scheme for maximizing throughput in IEEE 802.11ax WLANs," in *Proc. IEEE Int. Conf. Parallel Distrib. Syst.*, Dec. 2019, pp. 1–8.

QINGHUA CHEN received the B.S. degree in computer science and technology from Tianjin University, Tianjin, China, in July 2005, and the M.S. degree in computer software and theory from Shanghai Jiao Tong University, Shanghai, China, in January 2008. She is currently pursuing the Ph.D. degree in control science and engineering with the Zhejiang University of Technology, and also an Associate Professor with the Wenzhou Vocational and Technical College, Zhejiang, China. Her current research interests include wireless local area networks and artificial intelligence.

ZHENGQIU WENG received the B.S. degree in computer science and technology from the Beijing Armored Force Engineering Institute, Beijing, China, in July 2003, and the M.S. degree in software engineering from the Beijing Institute of Technology, Beijing, in July 2005. She is currently pursuing the Ph.D. degree in computer science and technology with the Zhejiang University of Technology, and also an Associate Professor with the Wenzhou Vocational and Technical College, Zhejiang, China. Her current research interests include networks security and big data technologies.

XIAOFENG XU received the B.Eng. degree in electronics and information technology from the Ningbo University of Technology, Ningbo, China, in July 2017. He is currently pursuing the M.Eng. degree in computer science and technology with the Zhejiang University of Technology. His current research interest includes the next generation wireless LANs.

GUOLANG CHEN received the B.S. degree in economic information management and computer application from Zhengzhou University, Zhengzhou, China, in July 1998, and the M.S. degree in computer technology from Hangzhou Dianzi University, Hangzhou, China, in November 2007. He is currently an Associate Professor with the Wenzhou Vocational and Technical College, Zhejiang, China. His current research interests include network technology and E-commerce.