

A Greedy Reclaiming Scheduler for IEEE 802.11e HCCA Real-Time Networks

Anna Lina Ruscelli
a.ruscelli@sssup.it

Gabriele Cecchetti
g.cecchetti@sssup.it

Giuseppe Lipari
g.lipari@sssup.it

Antonella Mastropaolo
antonella.mastropaolo@gmail.com

ReTiS Lab. - TeCIP
Scuola Superiore S. Anna
Pisa, Italy

ABSTRACT

The IEEE 802.11e standard introduces Quality of Service (QoS) support for wireless local area networks and suggests how to design a tailored HCF Controlled Channel Access (HCCA) scheduler. However the reference scheduling algorithm is suitable to assure service guarantees only for Constant Bit Rate traffic streams, whereas shows its limits for Variable Bit Rate traffic. Despite the numerous alternative schedulers proposed to improve the QoS support for multimedia applications, in the case of VBR traffic satisfactory real-time performance has not been yet achieved.

This paper presents a new scheduling algorithm, *Unused Time Shifting Scheduler* (UTSS). It integrates a mechanism for bandwidth reclaiming into a HCCA real-time scheduler. UTSS assigns the unused portion of each transmission opportunity to the next scheduled traffic stream. Thanks to such feature, traffic variability is absorbed, reducing the waste of resources. The analytical evaluation, corroborated by the simulation results, shows that UTSS is suitable to reduce the delay experienced by VBR traffic streams and to increase the maximum burstiness sustainable by the network.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*Wireless Communication*.

General Terms

Algorithms, Performance.

Keywords

Quality of Service, real-time scheduling algorithms, wireless LAN.

1. INTRODUCTION

The support for multimedia applications (like VoIP, multimedia streaming, online gaming, High Definition TV etc.) is an important challenge in the context of *Wireless Local Area Networks* (WLAN). The reference IEEE 802.11 standard [17] lacks the support of the *Quality of Service* (QoS) as requested by the multimedia applications, for instance, in terms of guaranteed bandwidth and bounded delay, jitter and packet loss. Indeed the earlier release IEEE 802.11b [18] standard was designed for best effort data transmissions. Hence the IEEE 802.11e amendment [19], has been produced to offer QoS support. In particular it introduces service differentiation, as required by multimedia applications. Service differentiation is possible thanks to two new *Medium Access Control* (MAC) functions that improve the functions existing in IEEE 802.11b. *Hybrid coordination function Controlled Channel Access* (HCCA) adds a parameterized QoS at the polling mechanism, whereas *Enhanced Distributed Channel Access* (EDCA) supplies prioritized QoS at the contention-based medium access. However numerous theoretical and simulative studies [1, 4, 5, 8, 14, 35] highlighted that the reference scheduler has good performance only with *Constant Bit Rate* (CBR) traffic whereas it is not suitable to provide temporal guarantees to *Variable Bit Rate* (VBR) traffic. Thus numerous scheduling algorithms have been proposed as alternative to the reference one in order to overcome its poor performance due to the fixed values the HCCA function assigns to the transmission parameters.

In this paper we present a novel scheduler named *Unused Time Shifting Scheduler* (UTSS) which integrates a mechanism for bandwidth reclaiming into a HCCA real-time scheduler. The mechanism assigns the portion of *Transmission Opportunity* (TXOP) unused by polled stations to the next scheduled traffic stream. Therefore a dynamic computation of the current transmission time is introduced. Such dynamic computation lets the scheduler to reassign the unused bandwidth, for dealing with rate variations of VBR traffic. We will show through analytical study and simulation that UTSS improves the provided QoS, being able to assure lower access delay and to absorb the variability of VBR traffic streams, without wasting of resources.

The rest of the paper is organized as follows: in Section 2 we summarize some relevant works that improve the HCCA reference scheduler. In Section 3, the proposed scheduler is described. In Section 4 some properties of the scheduler are analytically evaluated, whereas Section 5 show its performance through simulations. Finally, in Section 6 we draw our conclusions.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MSWiM'11, October 31–November 4, 2011, Miami, Florida, USA.
Copyright 2011 ACM 978-1-4503-0898-4/11/10 ...\$10.00.

2. HCCA SCHEDULING ALGORITHMS

The IEEE 802.11e standard defines a non-mandatory *reference* HCCA scheduler and the guidelines for the computation of protocol parameters by taking into account *QoS Station* (QSTAs) requirements. In particular the reference scheduler computes *Service Interval* (SI), i.e. the polling period, and *Transmission Opportunity* (TXOP), i.e. the transmission duration, as fixed values. SI is computed as a unique value for all admitted QSTAs with the aim to globally meet their temporal service expectations: its value has to be less than the beacon interval, thus each QSTA is polled at least once during the beacon interval, and less than the minimum *Maximum Service Interval* (MSI), thus polling period constraints of all traffic streams are respected. TXOP is computed as the maximum time to transmit at the minimum physical rate Γ_i the total amount of bits that can arrive during SI:

$$N_i = \left\lceil \frac{SI \cdot R_i}{L_i} \right\rceil \quad TXOP_i = \max\left(\frac{N_i \cdot L_i}{\Gamma_i}, \frac{M_i}{\Gamma_i}\right) + O,$$

where M_i is the maximum *MAC Service Data Unit* (MSDU) size, i.e. 2304 bytes, L_i is the nominal MSDU size, R_i is the mean data rate and O is the transmission overhead due to interframe spaces, ACK, CF-POLL.

The admission control test for deciding whether to admit a new stream in the HCCA reference scheduler is:

$$\frac{TXOP_{k+1}}{SI} + \sum_{i=0}^k \frac{TXOP_i}{SI} \leq \frac{T - T_{CP}}{T} \leq 1.$$

where k is the number of admitted streams, $k+1$ indexes the newly admitted stream, T is the beacon interval and T_{CP} is the EDCA duration.

SI and TXOP are recomputed only if a new *Traffic Stream* (TS) arrives. Their values are based on *worst case* conditions; this produces a too stringent admission control test and a not-optimal resource management. Moreover, since SI is the same of all admitted QSTAs and TXOP is globally assigned to a QSTA, all different TSs are polled with the same period and are served with the same computation time. This makes the reference scheduler suitable to serve CBR traffic but unable to efficiently adapt the resource management to VBR TSs, as highlighted by numerous studies and evaluations about the reference scheduler [8, 14, 29, 34].

Many scheduling algorithms alternative to the reference one have been proposed to improve the HCCA QoS provisioning [10, 12, 28, 34] and few of them are focused on real-time support [5], i.e. on temporal guarantees. In the following some real-time schedulers that, at the best of our knowledge, have introduced significant contributions are summarized. They range over different approaches including queue length models, feedback-based schemes, bandwidth reclaiming methods, exploitation of EDCA resources using the IEEE 802.11e *HCCA-EDCA Mixed Mode* (HEMM) mode that permits a QSTA to jointly use both the HCCA and the EDCA MAC mechanisms. In particular we will deeply describe some algorithms based on the concept of deadlines that is used by the proposed UTSS scheduler.

Fair HCF (FHCF) [1] aims to improve the fairness of both CBR and VBR traffic and the delay performances assigning variable TXOPs by means of a mathematical model of the uplink TSs queues length, used to estimate the global packet delay. It distinguishes between the packet queuing delay, influenced by the variations in packet size and data rate, and the waiting time delay, defined as the interval between the packet arrival time and the QSTA polling time.

The *Feedback Based Dynamic Scheduler* (FBDS) [3] deals on a closed loop feedback control to restore the balance about the right

packets queue delivery by bandwidth recovering, limiting the maximum delay. TXOPs are dynamically assigned according to queue length estimation at the beginning of the *Controlled Access Phase* (CAP) through a discrete time model, corrected by the use of the actual queue length information sent by each QSTA; SI remains fixed.

The model of the channel when both HCCA and EDCA modes are used, presented in [22], shows that incrementing the portion of HCCA increases the medium utilization of large WLAN in saturation conditions and the determinism in the channel control. Instead large EDCA networks are affected by growing collisions that degrade their performance.

In [32] the economic model used to efficiently manage elastic traffic over EDCA and HCCA functions shows how the CW_{min} parameter, the CSMA/CA scheme, and the RTS/CTS procedure affect channel congestion and throughput. Moreover it finds the optimal value of the HCCA-EDCA ratio by means of optimization techniques.

The *Adaptively Tuned HCF* (AT-HCF) algorithm [23] dynamically tunes the HCCA and the EDCA durations to the different type of traffic, until they converge to the optimal values to improve the throughput of the overall system.

The *Overboost* local node scheduler [31] exploits EDCA bandwidth to integrate the HCCA activity to limit the delay experienced by the traffic streams waiting for the next HCCA polling time and improves the network performance. Before the contention period begins, Overboost moves the TSs traffic exceeding the assigned HCCA TXOP transmission time from the HCCA queue to the higher priority Access Category EDCA queue. This scheduler can collaborate with any type of centralized HCCA scheduling algorithm that continues to manage admission control, scheduling parameters, and polling list.

The use of deadlines introduces timing constraints and their management is tailored to meet temporal requirements. The *Scheduling Estimated Transmission Time - Earliest Due Date* (SETT-EDD) [13] algorithm uses a token bucket scheme of time units or TXOP timer to vary their TXOP over time according to the node requirements. SI of each node is computed taking into account its traffic profile, while *Earliest Deadline First* (EDF) [26] determines the polling order. Variable TXOP and SI enhance the scheduler flexibility and lead to significant reduction in average transmission delay and packet loss ratio.

In [9] a timer-based scheduler computes the transmissions deadlines as the smallest between the downlink and uplink ones, and then schedules traffic streams according to EDF algorithm.

Real-Time HCCA (RTH) scheduler [7] manages TXOP as critical section, which cannot be interrupted by higher priority flows, by using *Stack Resource Policy* (SRP) algorithm [2]. Then an EDF-based algorithm schedules TS transmissions. It is composed by two activities: the offline one executes admission control, computes transmission parameters and scheduling timetable, and the online one schedules traffic streams transmissions.

Adaptive Resource Reservation Over WLANs (ARROW) [33] dynamically computes each TXOP by taking into account the actual different buffered TSs data at the beginning of the polling. Moreover, MSI is bounded in order to ensure no deadline miss and delay requirements. Finally *Earliest Due Date* (EDD) [21] deadlines scheduling manages the QSTAs polling list.

The *Application-Aware Adaptive HCCA Scheduler* [20], derived from ARROW, distinguishes uplink and downlink schedulers, while the *Earliest Deadline First* (EDF) algorithm [26] defines the QSTAs polling order taking into account the stations deadlines. The uplink scheduler assigns each QSTA a minimum and a maximum

SI , adapted to the application and network conditions and to the buffered traffic.

Wireless Capacity Based Scheduler (WCBS) [6] dynamically updates QSTA polling list using EDF algorithm and uses static and dynamic parameters to adapt the transmissions scheduling to the TSs characteristics. During the admission control a pair of static parameters, that do not change during normal conditions, are assigned to each TS_i taking into account its *Traffic Specification* (TSPEC_{*i*}): the budget Q_i , i.e. the maximum transmission time during a period ($TXOP_i$), and P_i , the service interval (SI_i). The ratio $U_i = Q_i/P_i$ is the utilization factor of the stream, i.e. its bandwidth. Therefore the admission control test is defined as:

$$\frac{TXOP_{k+1}}{SI_{k+1}} + \sum_{i=0}^k \frac{TXOP_i}{SI_i} \leq \frac{T - T_{CP}}{T} \leq 1. \quad (1)$$

Instead, the dynamic parameters characterize each TS_i during the scheduling phase: the remaining time c_i assigned to TS_i during the next transmission, the absolute deadline d_i before the budget Q_i has to be exhausted, the next time p_i when TS_i will be polled if it has no more data to transfer or it has exhausted its $TXOP$, and the stream state (*transmitting, active, polling, idle*).

To the best of our knowledge the first mechanism using bandwidth reclaiming in IEEE 802.11e is [24] where open-loop strategies, based on the use of TSPEC, and closed-loop strategies, that use the information about the queue sent by stations to the QAP, are compared. In particular, two different *max-min fairness* algorithms are proposed with the aim reduce the delay produced by HCCA in the case of VBR traffic by recovering spare resources through a proportional controller. A weighted proportional function assigns an addition amount of resources proportionally to the traffic class and to the buffer length. *MAXMin Fair-Adaptive* (MMF-A) assumes fixed SI, whereas *MAXMin Fair-Adaptive with Rescheduling* (MMF-AR) considers dynamic SI values. The non linearities of the system that can affect the mentioned works are overcome by [30] with an optimal controller, based on Model Predictive Control [11], that dynamically assigns resources in order to empty buffers of the stations and reduce packet loss. Moreover in [27] both cited proportional controller as optimal controller algorithms are applied to multi-class traffic with different priorities and to heterogenous traffic in order to find the resource assignment tailored to maximize the throughput, reducing the packet loss experienced by the different types of traffic.

In [25] the authors propose a bandwidth reclaiming scheme for the IEEE 802.11 PCF function with *Weighted Round Robin* (WRR) scheduling policy. It determines when the unused transmission time can be either used to advance the next polling opportunity or assigned to the Contention Period. Moreover the proposed algorithm modifies the WRR polling list in order to put the stations with higher probability of generating unused time at the end of the list. This rearrangement aims to reduce the number of reclaimed stations but it makes the solution not extensible to HCCA function, where the polling order is strictly related to real-time guarantees.

3. UNUSED TIME SHIFTING SCHEDULER

Despite of many QoS scheduling algorithms enhancing the reference one, satisfactory performance for VBR traffic with real-time guarantees has not yet been achieved. For instance, EDF-based algorithms are well performing in the case of CBR traffic streams, whereas they are not able to follow the variations of VBR traffic. Due to the traffic variability when the instant data rate drops down, a QSTA transmits data for an amount of time shorter than the as-

signed $TXOP$. In this case the *QoS Access Point* (QAP), listening the idle channel for a time longer than a *Short Interframe Space* (SIFS), assumes the control of the medium and polls the next station. Therefore the unused time is lost.

Unused Time Shifting Scheduler (UTSS) aims to face off this problem by reclaiming the unused transmission time and provides a shortcut to have instantaneous dynamic $TXOP$, without modify the admission control (see Section 4).

UTSS integrates the scheduling rules of WCBS scheduler illustrated in Section 2 by keeping the same admission control and scheduling algorithm of WCBS. In Fig 1, the Admission Control of the WCBS scheduler tests the feasibility condition, then it computes both the static and dynamic parameters and finally the newly admitted TS is enqueued in the EDF-ordered list.

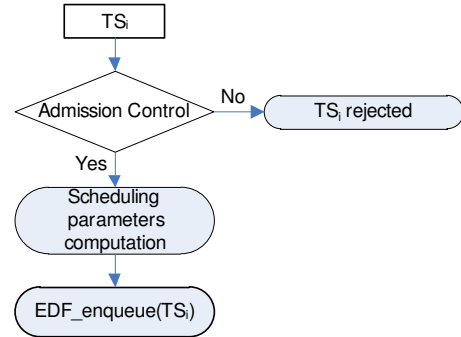


Figure 1: Admission control test.

UTSS adds a further scheduling rule during the computation of the dynamic parameters by modifying the $TXOP$ if some unused time is available. Fig 2 describes the relationship between WCBS and UTSS. The scheduler, after the extraction of the next

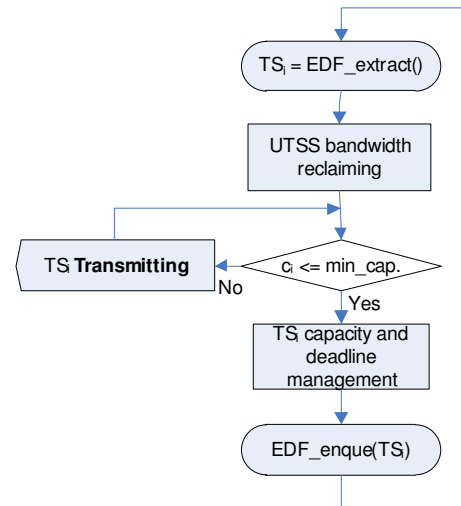


Figure 2: EDF + UTSS scheduling cycle.

TS_i from the EDT queue, performs the UTSS bandwidth reclaiming mechanism that affects the assigned $TXOP_i$. Then, if there is enough capacity to transmit, TS_i is scheduled for transmission. Otherwise WCBS operates replenishing the capacity and, if necessary, postponing the deadline. Therefore, while WCBS uses a constant $TXOP$ assigned during the admission control phase, UTSS can make this parameter variable each time TS is scheduled for

transmission. In particular, each time a QSTA does not use the full allocated $TXOP$, UTSS assigns this unused time to the next scheduled TS extracted from the EDF queue during the current CAP. We adopt the following notation:

$t_{end} = t_p + TXOP$: ending time of $TXOP$, when it is completely exhausted (t_p is the polling time);

T_{spare} : spare time of $TXOP$, computed as difference between t_{end} and the time when QSTA has actually finished its transmission.

T_{spare} is computed every time QAP polls a QSTA and $T_{spare} \neq 0$ when a station ends its transmission before t_{end} . Current T_{spare} is then added to the assigned $TXOP_i$ of the next polled station $QSTA_i$ that will receive a new $TXOP_i^l$ computed as follow:

$$TXOP_i^l = \begin{cases} TXOP_i & \text{if } T_{spare} = 0 \\ TXOP_i + T_{spare} & \text{if } T_{spare} > 0 \end{cases}$$

Fig. 3 better details UTSS. After the extraction from the EDF

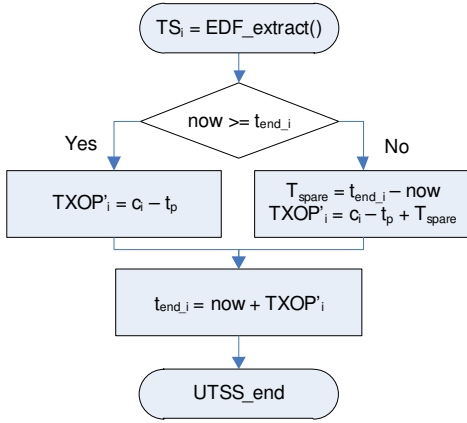


Figure 3: UTSS mechanism.

queue of the next TS to transmit, UTSS verifies if some spare time from previous transmissions is available. In such case, it adds T_{spare} to the $TXOP$ computed by WCBS ($TXOP_i = c_i - t_p$). Then the scheduler updates the estimated transmission ending time.

Hence next scheduled traffic streams can transmit their data not only exploiting the correspondent $TXOP$ (assigned during the admission control phase) or the remaining capacity, if they have been already served, but also by using the surplus T_{spare} . Fig. 4 shows a simplified scheduling example where WCBS and UTSS behaviors are compared. In WCBS the unused bandwidth of TS_2 is wasted, whereas with the help of UTSS TS_3 transmits its data during also the unused time of previously polled TSs.

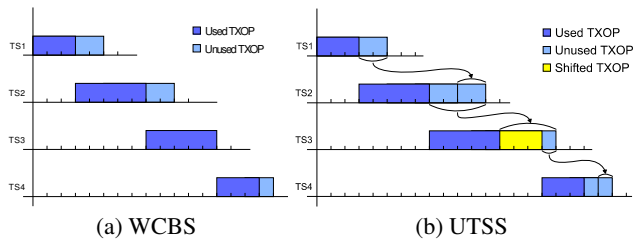


Figure 4: Scheduling example.

In this way it is possible to reclaim all the unused $TXOP$ portions. The total amount of T_{spare} can satisfy the temporal requirements of a highly variable traffic with a temporary load greater than the mean value used to calculate $TXOP$. Hence T_{spare} could be useful to absorb traffic peaks which characterize highly variable bit rate applications.

4. SCHEDULING ANALYSIS

This section analyzes the UTSS algorithm in order to highlight its effects on the centralized HCCA scheduler. The objective is to assess whether UTSS changes the admission control feasibility condition. We investigate from the analytical point of view how the added UTSS mechanism impacts on the admission control performed by the overall scheduler. Moreover, we check if the accumulation of this unused time can rise or not deadline miss. Then an upper bound of the maximum acceptable T_{spare} is computed. Finally to study the behavior of the overall scheduler in presence of bursts of traffic, an expression of the maximum tolerable burstiness is provided.

THEOREM 1. *The UTSS mechanism does not affect the validity of the admission control feasibility test.*

PROOF. We write Eq. 1 as follow:

$$T_{CAP} + T_{CP} = \sum_{i=0}^k T_i + T_{CP} \leq H$$

where T_{CAP} is the time assigned during the hyperperiod to HCCA, T_{CP} , is the time assigned to EDCA, and T_i is the $TXOP$ of each $QSTA_i$.

Without loss of generality we can assume that, $QSTA_1$ does not use its whole $TXOP_1$, thus its unused time $T_{spare_1} \neq 0$. We can highlight that, when UTSS reclaims unused time and assigns T_{spare} , the following relationship holds:

$$\begin{aligned} \sum_{i=0}^k T_i^l &= T_{eff_1} + T_2 + T_{spare_1} + \dots + T_k \\ &= T_{eff_1} + T_2 + T_1 - T_{eff_1} + \dots + T_k \\ &= \sum_{i=0}^k T_i \leq H - T_{CP}. \end{aligned}$$

where T_i^l is the new T_i computed taking into account T_{spare} , and T_{eff_1} is the used portion of $TXOP_1$. \square

THEOREM 2. *The UTSS mechanism of T_{spare} assignment does not raise deadline miss.*

PROOF. We distinguish two cases: 1) assignment of T_{spare} during a CAP, and 2) assignment of T_{spare} derived from the last polled QSTA of a CAP.

Case 1 During a CAP, the assignment of T_{spare} does not miss the deadline of the polled QSTAs. In fact the MAC scheduling algorithms manage the temporal sequence of QSTAs access to the medium by the use of fixed interframe space intervals (SIFS and PCF Interframe Space PIFS), that set the waiting time between consecutive polling of QSTAs listed in the polling queue. These strict rules avoid the presence of idle blocking time between polled stations transmissions, as in general it could happen in real-time systems, where the addition of a further slot of transmission time could jeopardize the real-time behavior of the next process to execute, raising a deadline miss. Instead, due to the IEEE 802.11e MAC scheduling rules, the use of T_{spare} only anticipates the transmission of a polled QSTA, without impacting on the next polled QSTA behavior, as shown in Fig. 4.

Case 2 When T_{spare} is derived from the last polled QSTA during a CAP, see Fig. 5a, two additional cases exist. If a T_{spare} recovered from the transmission of the last QSTA in the current CAP is shifted to the subsequent *Contention Period* (CP), the beginning of the following CAP is not affected since in this case there is only a redistribution of time portions between *Contention Free Period* (CFP) and CP of the same hyperperiod, see Fig. 5b.

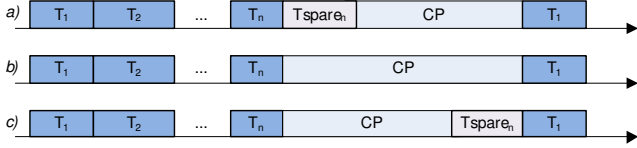


Figure 5: Example of T_{spare} propagation.

Instead, if we choose to propagate the T_{spare} allocation to the next CAP, by assigning the transmission time of the last polled QSTA to the first QSTA at the beginning of the next CAP, there is only an early polling of this QSTA of the same time interval, see Fig. 5c, whereas all other QSTAs are polled by respecting their Delay Bounds, thus without deadline miss. Moreover this choice increases the algorithm fairness by handling all QSTAs, included the first one of the new CAP, in the same way. \square

In order to avoid unpredictable growing of T_{spare} , especially due to accumulation of a large number of unused portion of *TXOPs* when the T_{spare} propagation across consecutive CAP is enabled, we can try to set an upper bound Θ suitable to meet the deadlines.

PROPOSITION 1. *The upper bound Θ of T_{spare} suitable to avoid deadline miss is equal to $\Theta = d_i - t_{end_i} + \delta$.*

PROOF. In the case of one-hop propagation, i.e. in the case of propagation of T_{spare} to the consecutive QSTA:

$$T_i \leq T'_i \leq T_i + T_{i-1}$$

where

$$T'_i = T_i + T_{spare_i},$$

whereas if all previous QSTAs do not use their *TXOP*,

$$0 \leq T_{spare_i} \leq \sum_{j=1}^{i-1} T_j$$

In particular T_{spare} is made available after that QSTA has informed QAP that has no data to send by responding to CF-POLL frame with a CF-NULL frame. This handshake requires a time interval equal to $\tau = t_{SIFS} + t_{NULL} + t_{SIFS}$, thus the general expression is:

$$\begin{aligned} 0 &\leq T_{spare_i} \\ &\leq \sum_{j=1}^{i-1} (T_j - \tau) \\ &\leq \sum_{j=1}^{i-1} (T_j) - (i-1)\tau. \end{aligned}$$

Finally, in order to avoid deadline miss, we can assume to accept T_{spare_i} if and only if it respects its upper bound Θ :

$$0 \leq T_{spare_i} \leq d_i - t_{end_i} + \delta = \Theta$$

where d_i is the absolute deadline of the polled QSTA_{*i*} and δ is an offset. \square

Since the proposed mechanism aims to introduce flexibility in the *TXOP* assignment at each polling by reclaiming the unused allocated resources, it could positively impact on the service provided to QSTA with VBR traffic. In fact VBR traffic makes *TXOP* not suitable since it is computed during the admission control considering mean value parameters. This leads to poor network performance. A metric useful to characterize the *burstiness* traffic with high rate variability is the *burstiness factor* (B), defined as the ratio between mean data rate r during a long time interval and peak data rate r' during the activity interval. Its value ranges from 1, in the case of CBR traffic, to 0, as the peak rate increases.

THEOREM 3. *The maximum traffic burstiness of a QSTA that receives T_{spare} and tolerable by the system is increased and the new burstiness factor B_{max} is equal to:*

$$B_{max} = \frac{t_{end_i} - t_{p_i}}{T_{spare} + t_{end_i} - t_{p_i}} \leq B.$$

where B is the burstiness factor without using UTSS.

PROOF. The use of $T_{spare} \neq 0$ by the next polled QSTA allows more enqueued traffic to be dispatched and the capacity to be increased and absorb peaks in the TS. Thus, with regard to the assigned *TXOP*, using T_{spare} has the same effect of instantaneously increasing the delivery rate. The portion of traffic dispatched using T_{spare} is $\varphi = r \cdot T_{spare}$. Considering that, in general, QSTA_{*i*} cannot use the whole T_i , then

$$\tau \leq t_{end_i} - t_{p_i} = T_{eff_i} \leq T_i$$

and r' is equal to

$$r'_i = \frac{\varphi_i + r(t_{end_i} - t_{p_i})}{t_{end_i} - t_{p_i}} = \frac{r(T_{spare} + t_{end_i} - t_{p_i})}{t_{end_i} - t_{p_i}}.$$

Finally the maximum burstiness B_{max} tolerable by the system, taking advantage by the use of the UTSS mechanism, is:

$$B_{max} = \frac{t_{end_i} - t_{p_i}}{t_{spare} + t_{end_i} - t_{p_i}} \leq B. \quad (2)$$

\square

This result is confirmed by the performance evaluation about end-to-end delay, throughput and discarded packets, see Section 5, where UTSS is evaluated in presence of VBR traffic.

5. PERFORMANCE ANALYSIS

This section evaluates the performance of the proposed UTSS scheduling algorithm versus WCBS and reference schedulers in order to analyze the effect produced by the introduction of the new mechanism. In particular the analysis takes into account the utilization efficiency of the network, the mean access delay, and the discarding rate of queued packets with expired delay bound. The result is preceded by a description of the simulation tools, their settings, the traffic models and the considered scenario.

5.1 Simulation settings and traffic model

Performance is evaluated through simulation using *ns-2* network simulator [16]. It is assumed that QSTAs communicate directly without hidden node problem. Thus RTS/CTS mechanism, MAC level fragmentation and multirate support are disabled. The Physical layer is specified in the IEEE 802.11g standard which uses OFDM (*Orthogonal Frequency-Division Multiplexing*) as mandatory modulation scheme; its parameters are listed in Table 1.

Parameters	Value	Parameters	Value
SIFS (μs)	10	PLCP header (b)	24
DIFS (μs)	28	Preamble (b)	72
PIFS (μs)	19	Data Rate ($Mbit/s$)	54
Slot Time (μs)	9	Basic Rate ($Mbit/s$)	1

Table 1: MAC/PHY simulation parameters.

The presented results have been obtained simulating independent replication of 700 s with a warm-up time of 100 s until the 95% confidence interval is reached for each measure.

The network scenario chosen for simulations is composed by seven QSTA and one QAP. Each QSTA transmits *one* uplink TS which is received by the QAP. Each TS has a *Traffic Specification* (TSPEC) different from the others TS. In particular one station sends VoIP traffic encoded with G.729A codec, one station transmits a video conference and five stations transmit video streaming applications. One station send data traffic with SDU of 1500 bytes through legacy *Distributed Coordination Function* (DCF). The data station operates in asymptotic condition, i.e. it is always backlogged in order to saturate the channel. The parameters of the VoIP traffic are shown in Table 2.

Parameter	Value	Parameter	Value
Frame size (B)	10	Payload size (B)	20
Frame per packet	2	IP/UDP/RTP	
Period (s)	0.02	Header size (B)	40
Data rate (kb/s)	24	SDU size (B)	60

Table 2: G.729A VoIP traffic stream parameters.

Video streaming traffic has been generated using pre-encoded high quality MPEG4 trace files of 60 minutes each from the Internet archive of traces [15]. Such traces are: *Jurassic Park* (VS1), *Silence of the lambs* (VS2), *Mr. Bean* (VS3), *Die hard III* (VS4), *Robin Hood* (VS5). The video conference (VC) session has been represented by the pre-encoded *LectureHQ-Reisslein* trace file.

Parameter	VC	VS1	VS2
Mean frame size (B)	3800	3800	2900
Maximum frame size (B)	11386	11386	22239
Period (s)	0.040	0.040	0.040
Mean data rate (kb/s)	770	770	580
Maximum data rate(kb/s)	3300	3300	4400
Parameter	VS3	VS4	VS5
Mean frame size (B)	2900	3500	4600
Maximum frame size (B)	15251	16960	16550
Period (s)	0.040	0.040	0.040
Mean data rate (kb/s)	580	700	910
Maximum data rate(kb/s)	310	3400	3300

Table 3: Video streaming and video conference parameters.

5.2 Efficiency analysis

Here we evaluate the efficiency of UTSS, intended as a measure of how well it utilizes the network resources, and we compare the result with the one of the other considered schedulers.

The first efficiency parameter is the *null rate* defined as the number of CF-Null packets sent by a QSTA in response to a CF-Poll, when it has no traffic to transmit. In Fig. 6 UTSS and WCBS have similar values of null rate for each traffic stream: this is because

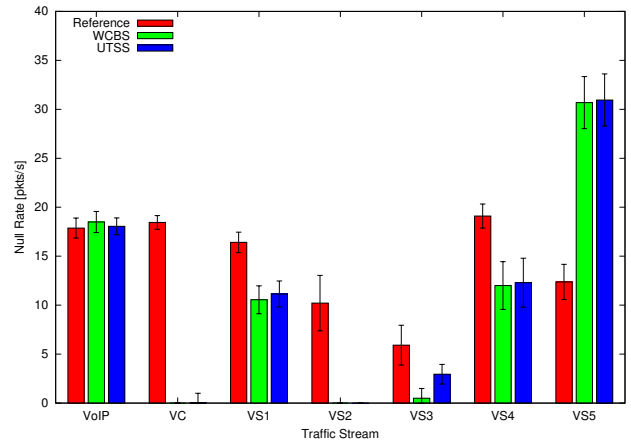


Figure 6: Null Rate.

they use the same polling interval. Often the null rate of these schedulers is better than the reference one since they poll the stations taking into account the different SI of each TS: such feature lets these schedulers to perform a zero null rate with some TS (e.g. VC and VS2 in Fig. 6). In the case of UTSS, even if the polling instant can be advanced, the polling interval variation is negligible and it does not affect the average null rate.

Fig. 7 shows the polling interval used by the schedulers with each TS. The reference scheduler uses a unique value of the polling interval for all the TSs, as reported in Sec. 2, which is less than the minimum MSI of the all admitted TS. Instead, WCBS and UTSS poll each TS_i using its SI_i .

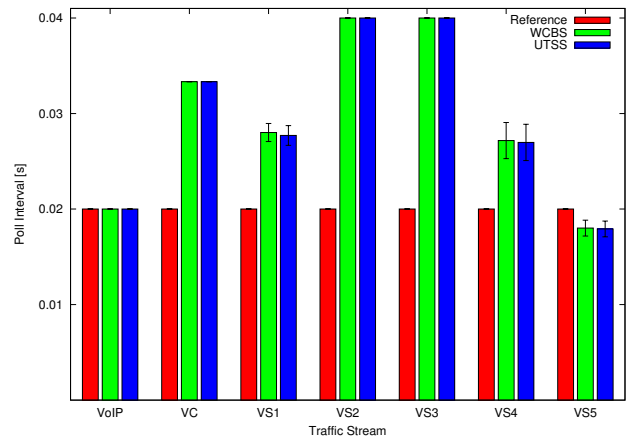


Figure 7: Poll Interval.

Finally Fig. 8 shows that UTSS does not significantly affect the throughput, thus network utilization is roughly the same.

5.3 Delay analysis

The access delay is defined as the time elapsed from when the frame reached the MAC layer until when the frame is successfully acknowledged. Fig. 9 shows that mean value of the access delay of UTSS is improved with respect to WCBS, in particular for TS with higher VBR, as VS2 and VS3. This confirms the obtained analytical results about deadline miss (see Theorem 2). Compared to the reference scheduler UTSS performs worse when serving TSs

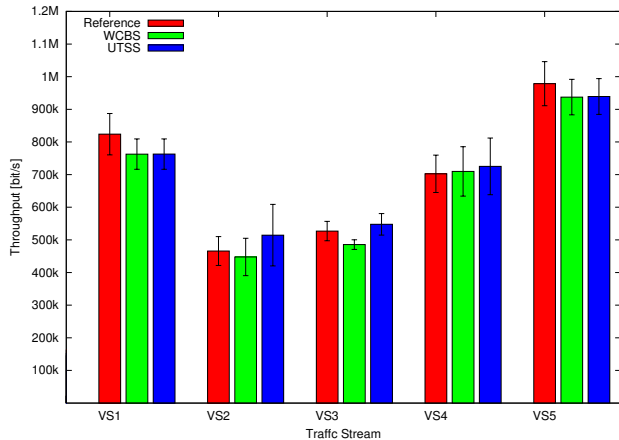


Figure 8: Throughput of VBR stations.

with less variable bit rate. This result is confirmed by the behavior of other EDF-based schedulers [5].

Looking at the *Cumulative Distribution Function (CDF)* of the access delay, Fig. 10 shows that after a time interval of 0.05 s UTSS allows 70% of transmitted packets whereas WCBS is limited to only 30%.

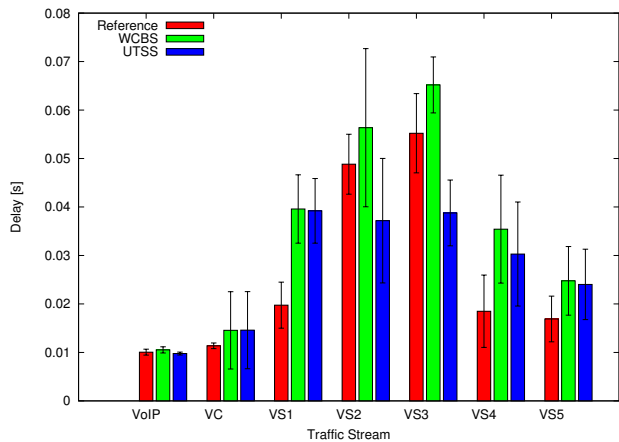


Figure 9: Mean Access Delay.

This is the most important result that highlights how UTSS is suitable for improving the real-time behavior of WCBS in the case of VBR traffic.

5.4 Packets drop analysis

Here we evaluate the amount of dropped packets from the queues due to expiration of *Delay bound* traffic parameter. Such parameter is chosen taking into account the length of the play-out buffer of a typical consumer device. For this reason this analysis aims to highlight the differences between the schedulers and not the absolute value performed by each single scheduler.

In Fig. 11 the number of dropped packets is improved by UTSS when the considered TS is highly variable. This is due to the fact that UTSS, while it recovers the unused time from the previous transmission, it reduces the waiting time of the scheduled TS. In particular, the VS3 TS experiences an improvement up to 60% with respect to WCBS and reference scheduler. This result confirms the analytical consideration about the burstiness (see Theo-

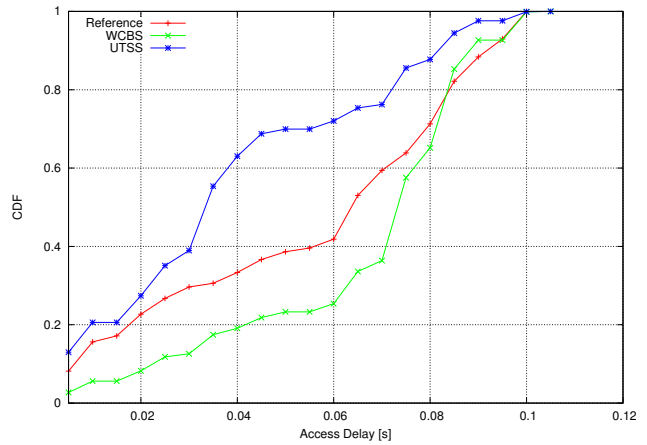


Figure 10: Access Delay CDF of VS3 Traffic Stream.

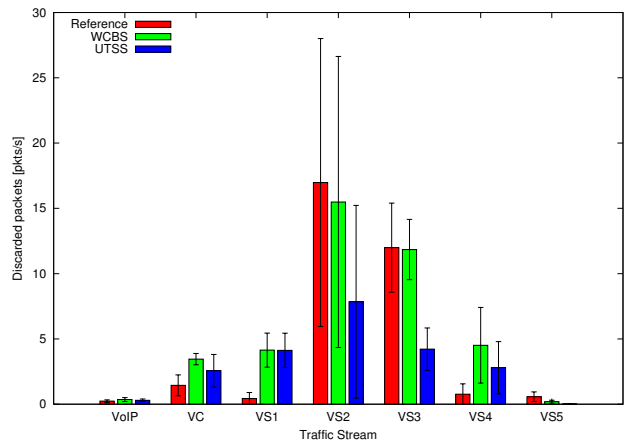


Figure 11: Discarded packets.

rem 3), which demonstrates that UTSS is suitable to absorb data rate peaks.

6. CONCLUSIONS

In this paper we have presented *Unused Time Shifting Scheduler*, a novel HCCA scheduler for IEEE 802.11e networks that is suitable for improving performance of a HCCA real-time scheduler, like WCBS, by reclaiming the unused portions of transmission opportunity intervals assigned for traffic streams transmission.

This mechanism is effective especially when high variable bit rate traffic stream are involved, such as in the case of video streaming transmissions.

The analytical evaluation shows that it efficiently uses the wireless medium in presence of different types of traffic, and increases the burstiness tolerable by the network.

These results has been validated through simulations showing that UTSS improves the performance of the existing schedulers, i.e. WCBS and reference scheduler, in terms of null rate, throughput and access delay.

7. REFERENCES

- [1] P. Ansel, Q. Ni, and T. Turletti. FHCF: A simple and efficient scheduling scheme for IEEE 802.11e wireless lan. *Mobile Networks and Applications*, 11(3):391–403, June 2006.

- [2] T. P. Baker. Stack-based scheduling for real-time processes. *Real-Time Syst.*, 3(1):67–99, 1991.
- [3] G. Boggia, P. Camarda, L. G. Grieco, and S. Mascolo. Feedback-based control for providing Real-Time services with the IEEE 802.11e MAC. *IEEE/ACM Trans. Net.*, 2(15):323–333, Apr. 2007.
- [4] G. Cecchetti and A. L. Ruscelli. Performance evaluation of Real-Time schedulers for HCCA function in IEEE 802.11e wireless networks. In *Proc. ACM Q2SWinet*, Vancouver, Canada, Oct. 2008.
- [5] G. Cecchetti and A. L. Ruscelli. Real-Time support for HCCA function in IEEE 802.11e networks: a performance evaluation. *Security and Communication Networks*, 2010. Special Issue on Security for QoS Assured Wireless Networks.
- [6] G. Cecchetti, A. L. Ruscelli, and F. Checoni. W-CBS: a scheduling algorithm for supporting QoS in IEEE 802.11e. In *Proc. QSHINE*, pages 1–7, Vancouver, British Columbia, Canada, 2007. ACM.
- [7] C. Cicconetti, L. Lenzi, E. Mingozzi, and G. Stea. Design and performance analysis of the Real-Time HCCA scheduler for IEEE 802.11e WLANs. *Computer Networks*, 51(9):2311–2325, 2007.
- [8] J. Cowling and S. Selvakennedy. A detailed investigation of the IEEE 802.11e HCF reference scheduler for VBR traffic. In *Proc. 13th ICC*, pages 453–459, Chicago, US, Oct. 2004.
- [9] Y. Fan and C. Huang. Real-Time traffic scheduling algorithm in WLAN. In *Proc. 4GMF*, San Diego, CA, 2005.
- [10] H. Fattah and C. Leung. An overview of scheduling algorithms in wireless multimedia networks. *IEEE Wireless*, 9(5):76–83, Oct. 2002.
- [11] C. Garcia, D. Prett, and M. Morari. Model predictive control: theory and practice: a survey. *Automatica*, 25(3):335–348, 1989.
- [12] A. Grilo, M. Macedo, and M. Nunes. A service discipline for support of IP QoS in IEEE 802.11 networks. In *Proc. PWC*, Laapenranta, Finland, Aug. 2001.
- [13] A. Grilo, M. Macedo, and M. Nunes. A scheduling algorithm for QoS support in IEEE 802.11e networks. *IEEE Wireless Comm.*, 10(3):36–43, June 2003.
- [14] A. Grilo and M. Nunes. Performance evaluation of IEEE 802.11e. In *13th IEEE PIMRC*, volume 1, pages 511–517, Lisboa, Portugal, Sept. 2002.
- [15] <http://traces.eas.asu.edu/>, 2005.
- [16] <http://www.isi.edu/nsnam/ns/>. *Network Simulator 2*, 1995.
- [17] IEEE802.11. Wireless LAN medium access control (MAC) and physical layer (PHY) specification, 2007.
- [18] IEEE802.11b. Wireless LAN medium access control (MAC) and physical layer (PHY) specification: Higher-speed physical layer extension in the 2.4 GHz band, 1999.
- [19] IEEE802.11e. 802.11e-2005 IEEE standard for information technology telecommunications and information exchange between systems local and metropolitan area networks specific requirements part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Amendment 8: Medium access control (MAC) quality of service enhancements, 2005.
- [20] I. Inanc, F. Keceli, and E. Ayanoglu. An adaptive multimedia QoS scheduler for IEEE 802.11e wireless LANs. In *Proc. IEEE ICC*, Istanbul, Turkey, June 2006.
- [21] J. R. Jackson. Scheduling a production line to minimize maximum tardiness. Research Report 43, University of California, Los Angeles, CA, 1955. Management Science Research Project.
- [22] C. Kuan and K. Dimiyati. Utilization model for HCCA/EDCA mixed mode in IEEE 802.11e. *ETRI Journal*, 29(6):829–831, Dec. 2007.
- [23] W. K. Lai, C. Shien, and C. Jiang. Adaptation of HCCA/EDCA ratio in IEEE 802.11 for improved system performance. *Int. J. on Innovative Computing, Inf. and Contr.*, 5(11(B)):4177–4188, Nov. 2009.
- [24] P. Larcheri and R. Lo Cigno. Scheduling in 802.11e: Open-loop or closed-loop? In *Proc. IFIP WONS*, Les Ménuires, France, Jan. 2006.
- [25] J. Lee, I. Shin, G. Park, W. Song, J. Kim, and J. Hong. An efficient bandwidth reclaim scheme for the integrated transmission of real-time and non real-time messages on the WLAN. In *Proc. ICCS*, volume 4490, pages 925–932, Beijing, China, May 2007.
- [26] C. L. Liu and J. W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *J. ACM*, 20:46–61, Jan. 1973.
- [27] R. Lo Cigno, L. Palopoli, and A. Colombo. Analysis of different scheduling strategies in 802.11e networks with multi-class traffic. In *Proc. IEEE LCN*, Dublin, Ireland, Oct. 2007.
- [28] S. Lu, V. Bharghavan, and R. Srikant. Fair scheduling in wireless packet networks. *IEEE/ACM Trans. Net.*, 7(4):473–489, Aug. 1999.
- [29] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor. IEEE 802.11e wireless LAN for quality of service. In *Proc. European Wireless*, volume 1, pages 32–39, Florence, Italy, Feb. 2002. invited paper.
- [30] L. Palopoli, R. Lo Cigno, and A. Colombo. Control and optimization of HCCA 802.11e access scheduling. In *Proc. IEEE CDC*, New Orleans, LA, Dec. 2007.
- [31] A. L. Ruscelli, G. Cecchetti, A. Alifano, and G. Lipari. Enhancement of QoS support of HCCA schedulers using EDCA function in IEEE 802.11e networks. *Ad Hoc Networks*, In Press, 2010. Special Issue on Recent advances in analysis and deployment of IEEE 802.11e and IEEE 802.11p protocol families.
- [32] V. A. Siris and C. Courcoubetis. Resource control for the EDCA and HCCA mechanisms in IEEE 802.11e networks. In *WiOpt*, pages 22–27, Boston, Massachusetts, 2006.
- [33] D. Skyrianoglou, N. Passas, and A. K. Salkintzis. ARROW: An efficient traffic scheduling algorithm for IEEE 802.11e HCCA. *IEEE Transaction on Wireless Communications*, 5(12):3558–3567, Dec. 2006.
- [34] S.-L. Tsao. Extending earliest-due-date scheduling algorithms for wireless networks with location-dependent errors. In *52nd IEEE VTC-Fall*, volume 1, pages 223–228, Boston, MA, Sept. 2000.
- [35] S. Wang and A. Helmy. Performance limits and analysis of contention-based IEEE 802.11 MAC. In *Proc. IEEE LCN*, Tampa, FL, Nov. 2006.