

# QoS Based Aggregation in High Speed IEEE802.11 Wireless Networks

Seyed Vahid Azhari<sup>\*†</sup>, Ozgur Gurbuz<sup>†</sup>, Ozgur Ercetin<sup>†</sup>

<sup>\*</sup>School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran

Email:azharivs@iust.ac.ir

<sup>†</sup>Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey

Email:[ogurbuz, oercetin]@sabanciuniv.edu

**Abstract**—We propose a novel frame aggregation algorithm with statistical delay guarantee for high speed IEEE802.11 networks considering link quality fluctuations. We use the concept of effective capacity to formulate frame aggregation with QoS guarantee as an optimization problem. The QoS guarantee is in the form of a target delay bound and violation probability. We apply proper approximations to derive a simple formulation, which is solved using a Proportional-Integral-Derivative (PID) controller. The proposed PID aggregation algorithm independently adapts the amount of time allowance for each link, while it needs to be implemented only at the Access Point (AP), without requiring any change to the 802.11 Medium Access Control (MAC). More importantly, the aggregator does not consider any physical layer or channel information, as it only makes use of queue level metrics, such as average queue length and link utilization, for tuning the amount of time allowance. NS-3 simulations show that our proposed scheme outperforms Earliest Deadline First (EDF) scheduling with maximum aggregation size and pure deadline-based aggregation, both in terms of maximum number of stations and channel efficiency.

**Index Terms**—Effective Capacity, Frame Aggregation, IEEE802.11, WLAN, PID Controller, Link Scheduling.

## I. INTRODUCTION

Due to increased popularity of applications, such as YouTube and live TV, high quality video delivery is becoming more important and critical, not only for service providers, but also for equipment and network vendors. Moreover, it is expected that, by 2019, consumer video traffic will constitute 80% of all consumer Internet traffic [1] due to the rapid development of smart-phones. Hence most video traffic will be eventually delivered over wireless links, in particular over Wi-Fi networks. Wi-Fi, i.e., wireless local area network (WLAN) industry has responded to this demand by introducing the IEEE802.11n and more recently IEEE802.11ac standards, both of which employ Multiple Input Multiple Output (MIMO) techniques to boost the throughput. However, the higher data rates available by MIMO are not well utilized, due to the contention overhead and lengthy headers sent at minimum rate. Furthermore, MIMO links exhibit significant variation that need to be considered for efficient channel use.

The frame aggregation feature has been first introduced in IEEE 802.11n to make better use of the high speed channel. When aggregation mode is enabled, a wireless device sends back-to-back contention-free frames during a transmission opportunity (TXOP) instead of a single frame. As a result, physical (PHY) and medium access control (MAC) layer headers as well as contention overhead are amortized over a large number of frames. There are two types of aggregation specified by the IEEE802.11 standard, namely, Aggregate MAC Service Data Unit (A-MPDU) and Aggregate MAC Protocol Data Unit (A-MPDU). A-MSDU refers to aggregation of multiple link layer payloads above the MAC layer, before being processed by the MAC layer. On the other hand, A-MPDU refers to aggregation of bona-fide MAC layer frames with their individual MAC headers below the MAC layer, before being handed to the PHY for transmission.

A-MSDU is more efficient, as it reduces the MAC overhead to one header per aggregate frame, whereas in A-MPDU, there are as many MAC headers as individual frames. However, a single frame error in A-MSDU will render the entire aggregate frame useless, because there are no independent checksum headers to identify the affected frame. This limits A-MSDU to small size. Moreover, the IEEE802.11 standard does not allow large aggregation sizes in A-MSDU so it is not the best choice of aggregation even under high SNR conditions. In practice, A-MPDU is preferred, since it maintains a separate MAC header for each frame, allowing identification frames, hence errors. Without loss of generality, we only consider aggregation with A-MPDU in this paper.

In order to obtain the best performance, channel condition should be considered, while setting the aggregation size. For instance, if the bit-rate is low, a large TXOP, i.e., a large aggregation size will waste valuable wireless resources. Likewise, transmission of small packets at high data rates will under utilize MIMO links. In this paper, we use the effective capacity theory for setting the appropriate size of a TXOP over a fluctuating channel. According to this theory, if the size of a TXOP is adjusted such that the effective capacity provided to a link is as large as its traffic arrival rate, then a certain delay violation probability can be guaranteed. Direct use of effective capacity can be complicated, as it requires knowledge of the channel service process; however, we apply an approximation

to derive a simpler problem, which considers only queue level metrics. Despite being non-convex, this problem is solved using a Proportional-Integral-Derivative (PID) controller. The main contributions and characteristics of our solution can be listed as follows:

- 1) We propose a frame aggregation algorithm, which provides statistical delay guarantee in high speed WLANs. This is ensured by basing our aggregation algorithm on effective capacity.
- 2) Our aggregation scheme makes no use of physical layer or channel state information, yet it is able to find a suitable aggregation size under channel fluctuations. Our effective capacity based approach is unique in that, it only relies on queue level metrics.
- 3) Our aggregation algorithm works above the Enhanced Distributed Channel Access (EDCA) layer, supported by all commodity Wi-Fi equipment. Furthermore, it is entirely implementable at the transmitter side (e.g., Access Point (AP) for downlink traffic) without requiring any changes to IEEE802.11 MAC.
- 4) Our aggregation scheme outperforms earliest deadline first (EDF) scheduling with maximum aggregation size and pure deadline-based aggregation, both in terms of maximum number of stations and channel efficiency.

The rest of the paper is organized as follows: In the next section, we provide a brief overview of the literature on aggregation and TXOP tuning. Section III provides the background on effective capacity, explaining how it can be employed as a framework for statistical QoS guarantees and how queue level metrics can be related to the effective capacity of a link. Section IV describes our system model and the downlink aggregation problem with statistical QoS. Section IV-C presents our proposed aggregation algorithm, which is based on a PID controller. This is followed by Section V, which presents our ns-3 simulations results, and finally, the paper is concluded in Section VI.

## II. RELATED WORK

When aggregation is considered, the common practice is to set the TXOP, so that all outstanding packets for a link are transmitted [2], [3], [4]. Furthermore, in [5], the next TXOP is allocated to be large enough to consume both outstanding and newly arriving packets, by predicting the traffic generated by a VBR video source with long range dependency. Though this aggregation policy is very simple and it can result in low delay under light load, it can also exhibit poor channel efficiency, as it potentially causes a large number of small size transmissions, which can significantly increase delay or reduce capacity when the channel is operating close to saturation.

The Hybrid Coordination Function (HCF) Controlled Channel Access (HCCA) mode of IEEE802.11n has been used in [4], [6] to perform TXOP allocation. Specifically, [6] allocates TXOPs according to a token bucket, matching the VBR traffic specifications; whereas [4] adjusts TXOP based on queue leftover. An earliest deadline first scheduler is also used by [6] to decide which station's packets should be transmitted.

Our approach, on the other hand, works with EDCA which, unlike HCCA, is deployed by all commercial 802.11 devices.

Most of the previous works do not explicitly consider delay with channel variations. Only recently in [7], an aggregation algorithm that explicitly considers packet deadlines is proposed. In that work, the aggregation size is determined such that the aggregate frame transmission time does not exceed the amount of time remaining until the deadline of the head-of-line (HoL) packet. [8] proposes a control theoretic approach for providing delay guarantee over HCCA. However, some works have considered link quality variations in making aggregation decisions [9], [10], [11]. For example, in [9], TXOP is increased for links with low packet error rate (PER) and it is decreased otherwise. This scheme is purely heuristic with no explicit consideration of delay, and it works best if the channel changes very slowly.

Recently, [10] proposed a channel scheduling approach that takes into account rate fluctuations as a result of a slowly fading channel in an LTE system. Users are allocated channel times proportional to the ratio of their current instantaneous rate to their average rate. The scheme requires knowledge of instantaneous channel rate and delay QoS is not considered at all, unlike our work. In [11] another TXOP allocation scheme considering channel variations is proposed, where a lead-lag compensator borrows TXOPs from bad links and assigns them to those with good channels. A mathematical proof of long term fairness and a bound on the amount of TXOP difference between any two links are provided, however, QoS is not considered.

A recent study in [12] has verified the accuracy of effective capacity applied to IEEE 802.11 networks. Our proposed aggregation algorithm is based on the concept of effective capacity, so that a statistical delay QoS, in the form of a target delay violation probability for a given delay bound is guaranteed. Our algorithm achieves this by only monitoring several queue level performance metrics. Therefore, channel conditions and variations are considered without requiring explicit channel state information. Our scheme not only provides statistical QoS guarantees, but it is much simpler to implement, when compared to the existing works.

## III. THE EFFECTIVE CAPACITY LINK MODEL

Consider a wireless link with stationary random service rate  $r(t)$ , such as that for a stationary Markov fading channel. Wu and Negi have demonstrated in [13] that if the link is supplied by a source rate of  $\mu$ , then using large deviations theory, the distribution of the queuing and service delay can be approximated by applying the Chernoff bound as follows,

$$\Pr\{D(t) \geq D_{max}\} \approx \gamma(\mu)e^{-\theta(\mu)D_{max}}, \quad (1)$$

where  $\gamma(\mu)$  is the link utilization and  $\theta(\mu)$  is called the QoS-index of the link. For a given statistical delay bound  $D_{max}$ , QoS-index determines the steepness of the probability,  $\varepsilon$ , by which the delay bound is violated. It follows that the pair  $\{\gamma(\mu), \theta(\mu)\}$  entirely determines a statistical QoS guarantee,

in the form of a maximum delay, and a delay violation probability,  $\{D_{max}, \varepsilon\}$ , for such a link.

Using a dual argument to the theory of effective bandwidth, it has been shown in [13] that for a desired QoS-index,  $\theta$  and source rate,  $\mu$  the effective capacity of the link is,

$$EC(\theta, \mu) = \Gamma(\theta) = - \lim_{t \rightarrow \infty} \frac{\mu}{\theta t} \log E \left[ e^{-\frac{\theta}{\mu} C(t)} \right], \quad (2)$$

where  $C(t)$  is the cumulative random service process. Clearly, the effective capacity of the link should be as large as its traffic source rate, i.e.,  $\Gamma(\theta) \geq \mu$ , so that a certain statistical delay guarantee determined by the QoS-index is achieved. Conversely, the effective capacity required for a source rate  $\mu$  is achieved with a QoS-index of at most,

$$\theta = \Gamma^{-1}(\mu). \quad (3)$$

Calculating the effective capacity of a link or its inverse  $\Gamma^{-1}(\cdot)$ , for that matter, requires knowledge of the channel service distribution  $C(t)$ . Such information may not be always available and even if it is, obtaining  $\Gamma(\cdot)$  in closed form is generally not possible. Therefore, it is suggested in [13] to use the following approximation for obtaining  $\theta$ , which becomes accurate if (1) is satisfied with equality:

$$\hat{\theta} = \frac{\gamma \mu}{\gamma \mu \bar{S} + \bar{Q}}. \quad (4)$$

Here,  $\bar{S}$  is the average residual service time of the packet currently being transmitted as sampled by an arbitrary packet arrival, and  $\bar{Q}$  is the average queue length. This simple queue level estimate of  $\theta$  is obtained by manipulating Little's law as originally illustrated in [14]. In particular, by writing Little's law for the aforementioned link we get,

$$E[D] = \frac{\bar{Q}}{\mu} + \gamma \bar{S}. \quad (5)$$

Recall that for any positive random variable  $X \geq 0$ ,  $E[X] = \int_0^{+\infty} \Pr\{X \geq x\} dx$ . Applying this to (1) we obtain,

$$E[D] = \frac{\gamma}{\theta}. \quad (6)$$

Equation (4) follows from replacing  $E[D]$  in (5) with (6). The accuracy of the  $\hat{\theta}$  estimate has been recently verified in [12].

Let us now solve (1) for  $\theta$  by requiring the delay violation probability to be less than or equal to a target value  $\varepsilon$ , i.e.,  $\Pr\{D(t) \geq D_{max}\} \leq \varepsilon$ . We obtain the condition,

$$\theta \geq - \frac{\log(\varepsilon/\gamma)}{D_{max}}. \quad (7)$$

Hence, for a given statistical QoS requirement in the form of a delay bound and a target delay violation probability,  $\{D_{max}, \varepsilon\}$ , and for traffic arrival rate of  $\mu$  it follows from (4) and (7) that the following condition should be met,

$$\frac{\gamma \mu}{\gamma \mu \bar{S} + \bar{Q}} + \frac{\log(\varepsilon/\gamma)}{D_{max}} \geq 0. \quad (8)$$

The significance of Equation (8) is that all its parameters including, the probability of a non-empty link queue  $\gamma$ , the average queue length  $\bar{Q}$ , and the average packet residual service time  $\bar{S}$ , can all be easily and accurately estimated at the access point.

## IV. EFFECTIVE CAPACITY BASED AGGREGATION

### A. System Model

The potential performance improvements of using aggregation are best realized for high throughput non-interactive real-time applications such as live/stored video streaming. Interactive applications, on the other hand, are often constrained by small delays, allowing limited aggregation. Even a high quality video conference at 1Mbps would require packets to be sent in less than 50msec. This is equivalent to 6.25KB of data per transmission, resulting in an aggregation size of roughly four maximum size frames. However, if this were a video streaming application with a delay requirement of only one second, then each transmission could accommodate up to 125KB equivalent to 83 aggregated frames.

Moreover, the benefit of aggregation on the uplink direction is generally limited even for applications such as video streaming. This is due to the small amount of traffic that flows in the uplink direction. Hence, without loss of generality, we consider aggregation in the downlink direction. In the uplink direction, however, the default aggregation policy of commodity 802.11 wireless cards which aggregates all the available frames would suffice, due to the relatively small volume of traffic per transmission.

We consider a single 802.11n Basic Service Set (BSS) with a number of stations (STA) and an AP forwarding traffic to the stations. Frames to each STA are queued at the AP and sent in aggregates, the size of which is determined on a beacon by beacon basis by the AP according to our proposed aggregation scheme. It should be noted that, any other traffic flowing in the uplink/downlink direction which does not fall into the proposed aggregation scheme can co-exist along with the designated traffic and be sent using the default aggregation policy adopted by commodity 802.11 interfaces. As a result it is not considered in our system model.

We consider a slow fading wireless channel modeled using a multi-state Discrete Time Markov Chain (DTMC) [15]. Each MIMO sub-channel is modeled using an independent DTMC with states corresponding to non-overlapping SNR ranges. Each SNR range in turn corresponds to a certain best modulation and coding scheme (MCS) for that MIMO stream, which is supposedly selected by the link adaptation algorithm. The assumed link adaptation algorithm is based on the behavior of IEEE802.11ac cards, which use the same MCS across all MIMO streams, as suggested by the standard [16]; this is also common in 802.11n products. The current slow fading channel model and the link adaptation algorithm are selected to reflect a practical setting. However, it should be noted that our scheme can operate for any other channel model and link adaptation algorithm, since it entirely makes use of the queue level metrics.

### B. Problem Formulation

Consider some downlink  $l$ , ( $l = 1 \dots L$ ) with a given QoS requirement  $\{D_l, \varepsilon_l\}$ . This link is to be assigned a time allowance  $\tau_l$  during each beacon interval of duration,  $BI$ .

The link may consume its time allowance in a single or multiple TXOPs depending on the limitations set by the AP. Our objective is to have the AP schedule a given number  $L$  of down links at minimum resource usage, i.e., total time allowance, while QoS is satisfied using (8) as the constraint. This approach leaves maximum room for newly arriving connections as well as best-effort traffic, effectively maximizing system capacity. Our problem is expressed by the following mathematical program,

$$\min \sum_{l=1}^L \tau_l \quad (9)$$

subject to

$$\frac{\gamma_l(\tau_l)\mu_l}{\gamma_l\mu_l S_l(\tau_l) + Q_l(\tau_l)} + \frac{\log(\varepsilon_l/\gamma_l(\tau_l))}{D_l} \geq 0; \forall l = 1 \dots L \quad (10)$$

$$\sum_{l=1}^L \tau_l \leq BI. \quad (11)$$

Note that, each link is in fact a G/G/1 queue, for which  $\bar{Q}(\tau)$  and  $S(\tau)$  are in general non-convex functions of time allowance  $\tau$ . Even for the simpler case of an M/G/1 queue, for which closed form representations of these quantities are available, the optimization problem still remains non-convex, leading to the practical solution explained in the next section.

### C. PID Control Algorithm

We choose to provide a heuristic solution to the optimization problem in (9) by designing a Proportional-Integral-Derivative (PID) controller. Although there are many other alternatives, a PID control approach has been chosen due to its simplicity, ease of parameter selection and wide usage [17]. Constraint (11) can be relaxed to obtain  $L$  independent optimization problems, which can be independently solved for each link as follows:

$$\min \tau_l \quad (12)$$

$$\frac{\gamma_l(\tau_l)\mu_l}{\gamma_l\mu_l S_l(\tau_l) + Q_l(\tau_l)} + \frac{\log(\varepsilon_l/\gamma_l(\mu_l))}{D_l} \geq 0. \quad (13)$$

It follows that an objective value larger than  $BI$  implies infeasibility of the problem, in which case, each link will have its time allowance  $\tau_l$  rescaled for the solution to become feasible. This, however, results in graceful QoS degradation for all links. Observing that as the constraint increases the delay violation probability becomes much smaller than the target  $\varepsilon_l$ , we can interpret the quantity,

$$\beta_l(t) = \frac{\gamma_l(\tau_l)\mu_l}{\gamma_l\mu_l S_l(\tau_l) + Q_l(\tau_l)} + \frac{\log(\varepsilon_l/\gamma_l(\mu_l))}{D_l} \quad (14)$$

as the measure of link QoS provisioning. The cases for  $\beta < 0$  and  $\beta > 0$  represent under-provisioned and over-provisioned link, respectively. Clearly  $\beta = 0$  is the best choice, as we do not want to waste resources by over-provisioning. In fact, we claim that at optimality, all constraints will be binding. This is justified by observing that the cost of over-provisioning any link is strictly positive. Therefore, due to complementary

slackness, the constraints should be satisfied with equality. It follows that keeping track of the error of  $\beta - 0$  is a suitable candidate for building a PID based controller.

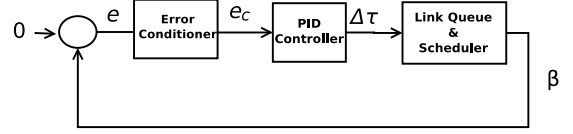


Fig. 1. PID Controller for Scheduling Each Link

Figure 1 shows a diagram of the PID controller for a given link. The AP constantly monitors  $\beta_l$  and at the end of each beacon interval, it calculates the error value,  $e_l = \beta_l - 0$  for all links. It then updates the time allowance of each link for the next beacon interval according to (15) using the PID control law by applying appropriate gains  $k_p, k_i, k_d$  to the error, its cumulative sum and its rate of change, respectively.

$$\tau_l(t+1) = \tau_l(t) - \left( k_p e_l + k_i \sum_{u=t-T}^t e_l(u) + k_d \frac{e_l(t) - e_l(t-1)}{BI} \right) \quad (15)$$

If necessary,  $\tau_l$  is then scaled appropriately to fit within a beacon interval of length  $BI$ . That is, if  $\sum_l \tau_l > BI$  then it will be set to

$$\tau_l \leftarrow \frac{\tau_l}{\sum_l \tau_l}. \quad (16)$$

If such rescaling happens too often, then it means that the system of links are infeasible. Although an admission control mechanism can be easily used to prevent such a case, it is not considered in the scope of this work. Hence, the aggregation algorithm continues to serve links by rescaling their time allowance, so each link experiences some level of QoS deterioration. We believe this is a practical policy for the contention based MAC of WLANs. In fact, commercial APs rarely perform any type of admission control and they allow connections to contend, leaving it up to the end applications to deal with the resulting delay or packet loss. For example, in a video application such as YouTube, the client may request a lower quality video to be streamed.

It is worthwhile to note that, the  $\theta$  bound in (7) is not tight, since it is based on the Chernoff inequality. Hence, there is some amount of intrinsic over-provisioning in our aggregator that will compensate for cases requiring rescaling of  $\tau_l$ . We have observed this behavior in our simulations.

## V. PERFORMANCE EVALUATION

We compare the performance of our proposed PID-based aggregation algorithm with two other algorithms, namely, EDF with maximum aggregation similar to [4], [6] and the deadline based aggregation in [7]; denoted as ‘‘EDF’’ and ‘‘Deadline’’, respectively. In the EDF scheme, the AP always serves the station for which its HoL packet has the earliest deadline and aggregates as many packets as available. In the Deadline scheme, the AP will only aggregate those packets, which will violate their deadline until the next beacon interval. The rationale behind the Deadline scheme is to postpone packets

TABLE I  
SIMULATION PARAMETERS

Simulation Time	100 s
Average Channel Rate	70 Mbps
Channel Coherence Time	3 msec
Traffic Source Rate	6 Mbps
Total AP Queue Size	60K Packets
Packet Size	1500 Bytes
Default $\varepsilon$	1%
Default Delay Bound	5 s

as much as possible, so that a large aggregation size can be achieved for better channel efficiency.

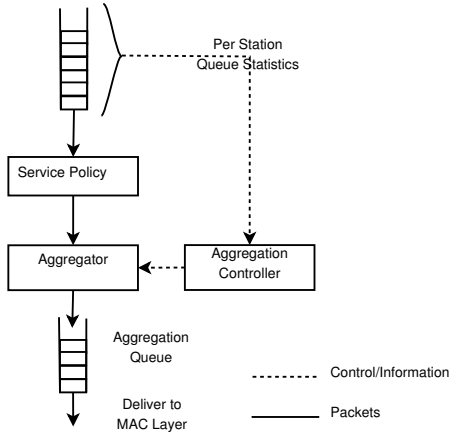


Fig. 2. Implementation of the aggregation algorithm in ns-3

In our ns-3 simulation environment<sup>1</sup>, we consider a single 802.11n BSS containing one AP and multiple stations all receiving traffic from the AP in the downlink direction. Our aggregation scheme is implemented as shown in Figure 2. It should be noted that our scheme is entirely implemented on the AP side and it does not modify any of the 802.11 MAC functions.

We consider channel utilization, average end-to-end delay and delay violation probability. We also consider the amount of time allowance allocated to each station when evaluating our proposed PID aggregation algorithm. Table I lists our main simulation parameters, which are used by default, unless noted otherwise. Moreover, we have generated enough simulation runs so that the 95% confidence interval of our results is within 10% of the reported mean values.

Figure 3 compares our proposed PID aggregation algorithm (denoted by PID Control) with EDF and Deadline schemes, when the number of stations varies from one to ten and for a target delay violation probability of 1%. Figure 3a plots the channel utilization and shows that our proposed PID aggregation achieves the best performance. In particular, when there are four stations, PID is able to satisfy the required delay violation probability, while keeping channel utilization

close to 50%, whereas, EDF and Deadline utilize more channel time at, 80% and 75%, respectively. PID reaches a maximum utilization of 98% for 8 stations, while EDF and Deadline saturate at almost 100% and 95%, respectively for 7 stations. The lower utilization of Deadline is due to its large queue build up at the AP, which results in packet loss.

Figure 3b supports these results showing that PID Control serves more stations compared to EDF and Deadline, while satisfying the statistical delay guarantee  $\{5sec, 1\%\}$ . The target  $\varepsilon$  of 1% is highlighted, showing that Deadline supporting 5 STAs has the lowest capacity, while EDF can support one more station. However, PID can support up to 8 STAs, providing roughly 30% higher capacity over EDF. This is due to better channel utilization, as previously illustrated in Figure 3a.

Furthermore, Figure 3c shows the average delay provided by each aggregation algorithm. Deadline based aggregation maintains an average delay just below the delay bound of 5 seconds for up to 6 stations. However, the average delay quickly becomes large, deteriorating the delay violation probability after that. EDF is on the other side of the spectrum, providing very small delay as long as the number of stations are small. This suggests that EDF is over-provisioning, which explains its slightly lower channel efficiency as compared to Deadline (see Figure 3a). PID, however, maintains an average delay between that of Deadline and EDF, but it is still far below the target delay bound, which is acceptable.

We next evaluate how our PID aggregation algorithm performs for different QoS requirements,  $\{D, \varepsilon\}$ . Figure 4 considers the effect of the delay bound on channel utilization and time allowance for each station. As the delay bound is relaxed, our PID aggregator makes more efficient use of the channel by allocating smaller time allowance to each station. For example, when there is only one station the time allowance of each station reduces from 15.38 msec to 11.7 msec, when the delay bound is varied between 1 and 20 seconds. This is a reduction of almost 25%, which can be a significant improvement if many stations are considered. For instance, when there are six stations, channel utilization can be lowered by 11%, when delay bound is changed from 1 to 20 seconds.

These results, in particular, show how PID aggregator is able to delicately relate delay bound to the amount of resource allocated to a link. However, the savings diminish as the number of stations increases. For instance, when there are 8 stations, time allowance of each station improves by less than a milli-second for the same range of delay bound. This is because the intrinsic over-provisioning provided by effective capacity diminishes as the number of stations becomes large<sup>2</sup>.

Furthermore, if more delay violation can be tolerated then PID can take advantage and reduce time allowance accordingly. Table II shows that for a fixed delay bound of 5s, the time allowance per STA reduces by almost 1.5 msec when  $\varepsilon$  is allowed to change from 0.1% to 20%. These results, however, suggest that  $\varepsilon$  has negligible effect of the

<sup>2</sup>Over-provisioning is intrinsic to effective capacity because the delay violation probability expressed in (1) is obtained using the Chernoff bound which is not very tight.

<sup>1</sup><https://www.nsnam.org/>

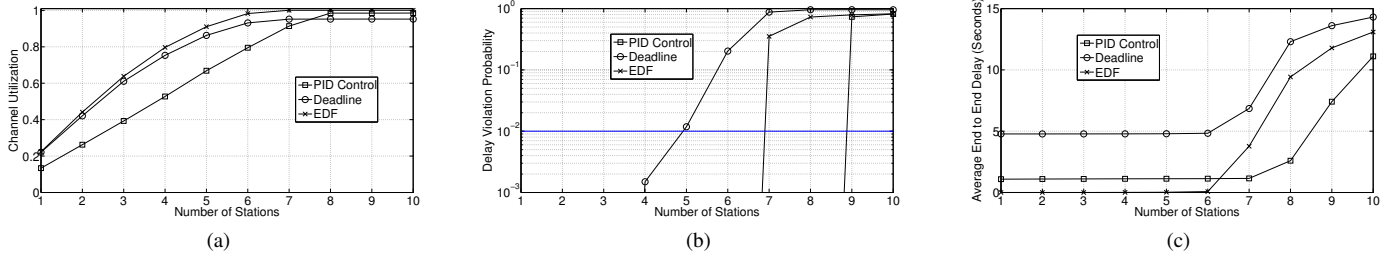


Fig. 3. Comparing Performance of PID, EDF and Deadline Based Aggregation Algorithms: (a) Channel Utilization, (b) Delay Violation Probability and (c) Average Delay.

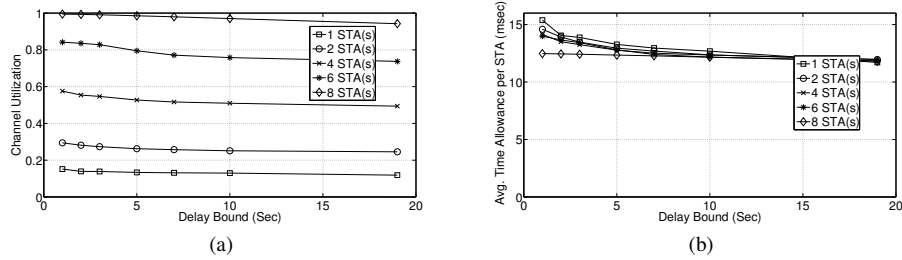


Fig. 4. The Effect of Delay Bound: (a) Channel Utilization and (b) Average Time Allowance Allocated to Each Station During a Beacon Interval.

TABLE II  
THE EFFECT OF TARGET DELAY VIOLATION PROBABILITY ON TIME ALLOWANCE

$\varepsilon_l$	0.001	0.01	0.02	0.05	0.10	0.20
Average $\tau_l$ (msec)	13.77	13.27	13.11	12.88	12.68	12.33

TABLE III  
THE EFFECT OF HETEROGENEOUS SOURCE RATES ON TIME ALLOWANCE

Source Rate ( $\lambda_l$ )	6Mbps	12Mbps	18Mbps
Average $\tau_l$ (msec)	12.60	24.20	35.60

link resource requirement as compared to the delay bound. In other words, it is more resource costly to achieve a target delay bound than it is to achieve it with precision. Likewise, it is simpler to provide very tight delay guarantee than it is to improve the delay bound.

In addition, we consider how PID aggregator differentiates between resource allocated to stations based on their source traffic demand. The results of Table III show how time allowance of three stations associated to an AP is different based on their traffic source rate. Note that, for all three stations a target delay violation probability of 1% was met.

## VI. CONCLUSIONS AND FUTURE WORK

We propose a novel frame aggregation algorithm for providing statistical delay guarantee over high data rate WLANs. Our aggregation algorithm requires no knowledge of the channel, as it uses average queue level metrics combined with the concept of effective capacity to achieve the statistical delay guarantee. Our approach is based on a PID controller implemented at the AP to determine the time allowance allocated for each link. Simulations show that the proposed PID based aggregation algorithm performs better than EDF service policy with maximum aggregation and deadline based aggregation.

In particular, our scheme improves channel utilization and maximum number of stations that can be supported for a given delay bound and delay violation probability. Furthermore, we illustrate how delay bound and its acceptable violation probability affect the time allowance required for each station. Our results suggest that making a delay bound tighter is less resource consuming than making the bound smaller.

As future work, we will investigate fine tuning parameters for our PID controller; however, even without fine tuning, the observed performance improvement is significant. We intend to apply the PID aggregation scheme to the latest IEEE 802.11ac standard, which allows higher data rates and larger aggregate frame size.

## REFERENCES

- [1] CISCO, "Cisco visual networking index: Forecast and methodology," *2014-2019 White Paper*, May 2015.
- [2] A. Ksentini, A. Gu eroui, and M. Naimi, "Adaptive transmission opportunity with admission control for ieee 802.11 e networks," in *Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*. ACM, 2005, pp. 234–241.
- [3] H. Liu and Y. Zhao, "Adaptive edca algorithm using video prediction for multimedia ieee 802.11 e wlan," in *Wireless and Mobile Communications, 2006. ICWMC'06. International Conference on*. IEEE, 2006, pp. 10–10.
- [4] H.-W. Ferng and A. Leonovich, "Periods scheduling under the hcca mode of ieee 802.11e," *Wireless Communications, IEEE Transactions on*, vol. 13, no. 12, pp. 7037–7049, Dec 2014.
- [5] W.-K. Kuo, "Traffic scheduling for multimedia transmission over ieee 802.11 e wireless lan," *IET communications*, vol. 2, no. 1, pp. 92–97, 2008.
- [6] A. Grilo, M. Macedo, and M. Nunes, "A scheduling algorithm for qos support in ieee802.11 networks," *Wireless Communications, IEEE*, vol. 10, no. 3, pp. 36–43, 2003.
- [7] E. Charfi, C. Gueguen, L. Chaari, B. Cousin, and L. Kamoun, "Dynamic frame aggregation scheduler for multimedia applications in ieee 802.11n networks," *Transactions on Emerging Telecommunications Technologies*, pp. n/a–n/a, 2015. [Online]. Available: <http://dx.doi.org/10.1002/ett.2942>
- [8] A. Annese, G. Boggia, P. Camarda, L. Grieco, and S. Mascolo, "Providing delay guarantees in ieee 802.11 e networks," in *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, vol. 4. IEEE, 2004, pp. 2234–2238.

- [9] K. Ju, D. Lee, and K. Chung, "Dynamic txop allocation to support qos based on channel conditions in wireless networks," in *Computing Technology and Information Management (ICCM), 2012 8th International Conference on*, vol. 2. IEEE, 2012, pp. 721–724.
- [10] R. Bhatia, T. Lakshman, A. Netravali, and K. Sabnani, "Improving mobile video streaming with link aware scheduling and client caches," in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014, pp. 100–108.
- [11] A. Arora, S.-G. Yoon, Y.-J. Choi, and S. Bahk, "Adaptive txop allocation based on channel conditions and traffic requirements in ieee 802.11e networks," *Vehicular Technology, IEEE Transactions on*, vol. 59, no. 3, pp. 1087–1099, March 2010.
- [12] A. Davy, B. Meskill, and J. Domingo-Pascual, "An empirical study of effective capacity throughputs in 802.11 wireless networks," in *Global Communications Conference (GLOBECOM), 2012 IEEE*. IEEE, 2012, pp. 1770–1775.
- [13] D. Wu and R. Negi, "Effective capacity: a wireless link model for support of quality of service," *Wireless Communications, IEEE Transactions on*, vol. 2, no. 4, pp. 630–643, July 2003.
- [14] A. Eckberg, "Approximations for bursty (and smoothed) arrival queueing delays based on generalized peakedness," in *Proc. 11th Int. Teletraffic Congress*, 1985, pp. 331–335.
- [15] H. S. Wang and N. Moayeri, "Finite-state markov channel-a useful model for radio communication channels," *Vehicular Technology, IEEE Transactions on*, vol. 44, no. 1, pp. 163–171, 1995.
- [16] "Ieee standard for information technology– telecommunications and information exchange between systemslocal and metropolitan area networks– specific requirements–part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications–amendment 4: Enhancements for very high throughput for operation in bands below 6 ghz." *IEEE Std 802.11ac-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012)*, pp. 1–425, Dec 2013.
- [17] K. J. strm and T. Hgglund, *PID Controllers - Theory, Design, and Tuning (2nd Edition)*. ISA, 1995. [Online]. Available: <http://app.knovel.com/hotlink/toc/id:kpPIDCTDT1/pid-controllers-theory/pid-controllers-theory>