Analytical Modeling of Saturation Throughput in Power Save Mode of an IEEE 802.11 Infrastructure WLAN

Pranav Agrawal*, Anurag Kumar†, Joy Kuri*, Manoj K. Panda‡
*Centre for Electronics Design and Technology
†Electrical Communication Engineering
‡Indian Institute of Science, Bangalore, India - 560 012

Abstract—We consider a single station (STA) in the Power Save Mode (PSM) of an IEEE 802.11 infrastructure WLAN. This STA is assumed to be carrying uplink and downlink traffic via the access point (AP). We assume that the transmission queues of the AP and the STA are saturated, i.e., the AP and the STA always have at least one packet to send. For this scenario, it is observed that uplink and downlink throughputs achieved are different. The reason behind the difference is the long term attempt rates of the STA and the AP due to the PSM protocol. In this paper, we first obtain the the long term attempt rates of the STA and the AP and using these, we obtain the saturation throughputs of the AP and the STA. We provide a validation of analytical results using the NS-2 simulator.

I. INTRODUCTION

In the normal mode of operation, also called the Continuously Active Mode (CAM), an STA always keeps its radio on, so it can receive and transmit at any time. This mode of operation is energy inefficient since STAs draw current even when they are idling. To save power during the period when there is less or no network activity, WiFi cards are provided with controls through which they can be turned off. To leverage this facility, the IEEE 802.11 standard has a feature using which STAs can turn off their radio without losing packets. This mode of operation is called as the Power Save Mode (PSM). In this mode, an STA can be in any one of the two state, active state and sleep state. In the active state the radio of the card is turned on, so it can receive as well as transmit, whereas in the sleep state, the radio of the card is turned off, hence in this state the STA can neither receive nor transmit. Generally PSM is efficient only when there is activity of very short duration during which the STA uploads and downloads bursts of packets. In this paper, we focus on this burst of activity, and assuming that during this activity both the STA and the AP are saturated. We observed that the throughput obtained by the AP is higher than that of the STA. This difference of the throughput is attributed to the different long term attempt rate of the STA and the AP, and attempt rate of the AP being higher.

Contribution: We consider a scenario in which a single STA is associated with the Access Point (AP). The STA is considered to be in PSM and carrying uplink and downlink traffic via the AP. We assume that the AP and the STA are saturated, i.e., immediately after the transmission of a packet, it is replaced by another packet. For this scenario we obtain the following:

- Attempt probability of the AP and the STA
- Throughput achieved by the AP and the STA (We will call it saturation throughput, since queues of the AP and the STA are saturated)

The paper is organized as follows: Section II we discuss related work. In Section III gives the overview of the PSM and the queuing structure at the AP. Section IV gives the overview of the IEEE 802.11 DCF. In Section V we analyze the attempt probabilities of AP and STA. In Section VI we obtain the saturation throughput of the AP and the STA. Section VII shows the comparison of the analytical and simulation results. Finally, Section VIII concludes the paper.

II. RELATED WORK

In a seminal paper, Bianchi [1] proposed an approximate model for the throughput performance of a single-cell IEEE 802.11 network that uses DCF as the medium access mechanism and in which all the nodes are saturated. Kumar et al. [2] extended the model and provided some new insights. In both of these papers, the authors evaluated the attempt probability \( \beta \) (also, the long term attempt rate) in a system slot as a function of the number of contending nodes. While in this paper, we obtain the attempt probability and saturation throughput for a scenario, which requires different analysis than presented in [1] and [2], due to the different behavior of the PSM. In our earlier submission [3], we focussed on the performance of PSM under application level downlink traffic over TCP. While in this paper, we consider both uplink and downlink traffic, which is different from TCP. We analyze attempt probability of the AP and the STA and saturation throughputs, which makes this paper more basic than our earlier submission [3].

This is the first paper to evaluate this scenario, there has been no attempt in the previous studies to obtain saturation throughput and the attempt probability for the saturated STA in PSM and the saturated AP.

Anastasi et al. [4], Lei and Nilsson [5], Baek and Choi [6] and Si et al. [7] evaluated the energy performance of PSM, but none of them attempt to obtain the saturation attempt rates of the STA and the AP. Apart from this, in all the above papers,
authors consider the PSM protocol implementation which is not practical. They consider the following sequence of frame exchanges: First the PSM STA sends the PS-POLL frame through contention, after SIFS AP sends the data packet and after SIFS again the STA sends the MAC ACK. So the AP does not contend to send data. In the presence of traffic from the AP to other STAs, when the AP receives the PS-POLL frame, some packets might already be present in the NIC queue of the AP, and these packets need to be sent first. So the above sequence of frame exchange cannot work under the scenario just described. We consider a different implementation of PSM protocol which is explained in the next section.

Krashinsky and Balakrishnan [8] and Quiao and Shin [9] focus on the interaction of the TCP slow start, RTT and PSM. Yong et al. [10] propose a way to minimize energy and delay by scheduling and informing the schedule to STAs through beacon frames. Tan et al. [11] propose to take advantage of throttling done by the TCP server in media streaming applications. In all of these papers, the authors focus on the energy saving either by modifying the PSM protocol [10], or by modifying the sleep wake schedule of the radio depending upon the characteristics of the application level traffic [8], [9] and [11].

On receiving a beacon, an STA sees an indication that there is a packet enqueued for it, then it contends for the medium to send the PS-POLL frame. In reply to the PS-POLL frame, the AP immediately sends a MAC ACK. This behavior is in contrast to earlier studies where it is assumed that a data packet is sent immediately. It is not practical to assume that the AP can immediately send a data packet in response to the PS-POLL, since there might be already some packets present in the NIC queue of the AP at the instant when AP receives the PS-POLL frame.

On receiving the PS-POLL from a STA, the AP dequeues the HOL packet from the corresponding PSM queue and enqueues it at the tail of the NIC queue. If the PSM queue of the STA is still non-empty, then the AP sets the More Bit in the dequeued packet to indicate to the STA that there are more packets stored for it at the AP. On receiving this packet, the STA checks the More bit. If it is set then it sends another PS-POLL frame. In this way, the STA does not sleep until its PSM queue at the AP becomes empty. And also, note that each PS-POLL packet permits the STA to download one packet enqueued at the AP.

Since the PS-POLL is a MAC level packet, it is enqueued at the HOL position in the transmission queue of the STA. If the STA is contending for a packet when a PS-POLL frame is generated, then the STA does not sample a new backoff, but uses the residual backoff of the packet under contention. Further, it is not possible that the STA receives a data packet when it is contending for PS-POLL, because it is only after the PS-POLL is sent, a data packet arrives at the STA.

There are some situations which are not specified in the protocol but are implementation dependent. Such situations and the assumed behaviors of an STA and the AP are described here. After sending a PS-POLL, the STA marks its state as waiting for unicast. If before the STA receives the unicast packet, the AP transmits a beacon frame and it indicates that there are packets at the AP for this STA, then this STA will not generate another PS-POLL frame. But this may result in a deadlock when the packet that it is meant for is lost, because then the STA will continue to be awake and will not send another PS-POLL. To prevent this situation, a timer is started when the STA sends the PS-POLL, and if the STA does not receive a packet before timer expiry, it goes to the sleep state. Subsequently in the next beacon interval, if the STA gets an indication, then it will send a PS-POLL to retrieve the packet from the AP. Further, if the beacon frames arrives at the STA when it is contending for PS-POLL, then it ignores the beacon frame, because the STA already knows that there is a packet at the AP for it.

III. PSM - OVERVIEW

In this section, we discuss the implementation of the PSM protocol that we have considered, and the queuing structure at the AP for the packets destined for the PSM clients. When any STA switches to Power Save Mode (PSM), it goes to sleep state (switches off its radio) and also informs the AP about it. Packets arriving at the AP for PSM STAs are stored in separate queues maintained for each PSM STA; we call them PSM Queues, see Fig. 1. There is a NIC queue in which MAC PDUs are enqueued for transmission by the PHY layer. The AP sends beacon frames periodically, through which it informs PSM STAs about the packets enqueued for them. PSM STAs also wake up periodically to listen to the beacon frame. If, on receiving a beacon, the STA is still non-empty, then the AP sets the More Bit in the dequeued packet to indicate to the STA that there are more packets stored for it at the AP. On receiving this packet, the STA checks the More bit. If it is set then it sends another PS-POLL frame. In this way, the STA does not sleep until its PSM queue at the AP becomes empty. And also, note that each PS-POLL packet permits the STA to download one packet enqueued at the AP.

In this section we give a brief overview of IEEE 802.11 DCF. We will refer to any wireless entity as node. If any node wants to transmit a packet then it first samples a backoff, which is a uniformly chosen number in between \((0, b_0 - 1)\), where, \(b_0\) is some number defined by the protocol. We call this range as the collision window and the chosen number as
backoff counter. In IEEE 802.11, time is divided into slots, and any node can attempt at the boundary of the slot only. After sampling the backoff, node senses the channel, if the channel is idle for more than DIFS than the node starts decrementing the backoff counter. The node decrements the counter by 1, if it senses the medium idle for a slot. It continues to decrement the backoff counter until it finds that the channel is busy or the counter reaches 0. On sensing the medium busy, the node freezes its backoff counter and hereafter continuously senses the medium to become free. After sensing the medium free for DIFS duration the node again starts decrementing the remaining counter. If the counter reaches 0, than it attempts to transmit the packet. If there is a collision then it increases the collision window and samples a new backoff from this new window. This procedure continues until either the node is able to successfully transmit the packet or it reaches the maximum number of collisions it can experience before the packet is discarded. We denote $K$ the maximum maximum number of collisions the packet can experience before it is discarded. We denote the initial range of collision window as $(0, b_0 - 1)$, and after $k$ collisions, collision window as $(0, b_k - 1)$.

V. ATTEMPT PROBABILITY

Since the AP is assumed to be saturated, which means it always has a data packet for the STA, so More bit is set in every data packet sent to the STA. On receiving a data packet, the STA has to send a PS-POLL frame. Although the AP is saturated, but all the packets lies in the PSM queue of the STA at the AP, and it is only after it receives a PS-POLL, a packet arrives in the NIC queue of the AP. So while the STA is counting down the backoff counter for the PS-POLL, the NIC queue of the AP is empty, so the STA transmits a PS-POLL without any possibility of collision. We treat the transmission of a data packet by the AP and the subsequent PS-POLL frame by the STA as a single activity period, this activity period also includes the back off counted for the PS-POLL frames, as shown in the Fig 2.

After the STA transmits a PS-POLL, it samples a fresh backoff from the initial collision window $(0, b_0 - 1)$ and again starts contending for the data packet. Thus we see that when the AP successfully transmits a data packet, it results in the STA sampling a fresh backoff for its packet. As a result the STA has to countdown more backoff than the AP to transmit a data packet, due to which the attempt probability of the STA is less than that of the AP. Since the STA uses the residual backoff to transmit a PS-POLL and there is no contention to transmit it, because of this the attempt rate to transmit a PS-POLL is higher than than that of the STA and the AP to transmit a data packet. In this section, we evaluate long term attempt rates of the AP to transmit data packet and the attempt rate of the STA.
to transmit data packet and PS-POLL. In next section, we will evaluate the saturation throughput of the AP and the STA.

We consider the activity duration as the transmission durations of all the frames, collisions, SIFS, DIFS,EIFS. we remove all the activity durations from the figure[2] and leaving behind only the backoff time as shown in Fig. 3. Note that it is the restricted back off time, since we consider the residual backoff counted to transmit the PS-POLL in the activity period as shown in the Fig [2].

Define the following: for \(0 \leq k \leq K; m \geq 0, j \geq 0\)
\[ B_{m,j}^{(k)} := \text{It is the back off sampled by the STA, after the } j^{th} \text{ AP interruption during the contention of its } m^{th} \text{ data packet and the } k^{th} \text{ back off duruation.} \]

Consider Fig. 3 the STA takes a new data packet \((m^{th})\) and samples a backoff \(B_{m,0}^{(0)}\). After decrementing this backoff there is a collision, so the STA samples a new back off \(B_{m,0}^{(1)}\). When the STA is counting down for \(B_{m,0}^{(1)}\), there is a AP success and back off of the STA is interrupted, as the remaining backoff of \(B_{m,0}^{(1)}\) is used by the STA to transmit PS-POLL. The STA now samples a new back off \(B_{m,1}^{(0)}\) and this goes on.

Assume the AP attempts in this restricted back off times at rate \(\beta_a\) per slot and the STA attempts with the rate \(\beta_s\) per slot.

Let the distribution of the \(k^{th}\) back off duration is:
\[ p_j^{(k)} = \frac{1}{b_k}, \text{ for } 0 \leq j \leq b_k - 1, 0 \leq k \leq K \quad (1) \]

Let's denote \(b_j^{(k)}\) the expectation of the \(k^{th}\) back off duration, which is given by:
\[ b_j^{(k)} = \sum_{j=0}^{b_k-1} j \frac{1}{b_k} = \frac{b_k - 1}{2} \quad (2) \]

Let \(k\) be the index of restricted back off times, we see that
\[ \beta_a = \lim_{n \to \infty} \frac{A_a(n)}{n} \quad (3) \]
\[ \beta_s = \lim_{n \to \infty} \frac{A_s(n)}{n} \quad (4) \]

Where, \(A_a(n)\) and \(A_s(n)\) are the attempts count until \(n\), in the restricted back off times. The success instants of the AP (in restricted back off times) are renewal instants.

Denote \(A_{a,m}, A_{s,m}\) the number of attempts made by the AP and the STA respectively to transmit their \(m^{th}\) data packet. Let's denote \(X_{a,m}, X_{s,m}\) the number of back off slots decremented by the AP and the STA respectively to transmit their \(m^{th}\) packet. Note that \(X_{a,m}\) does not include the back off used to transmit PS-POLL frames.

Using the renewal reward theorem [2], following can be written for \(\beta_a\) and \(\beta_s\):

\[ \beta_a = \lim_{n \to \infty} \frac{A_a(n)}{n} = \frac{E[A_{a,1}]}{E[X_{a,1}]} \quad (5) \]

\[ \beta_s = \lim_{n \to \infty} \frac{A_s(n)}{n} = \frac{E[A_{s,1}]}{E[X_{s,1}]} \quad (6) \]

A. Attempt Probability of the AP

Since the STA is attempting with probability \(\beta_s\) in every slot. So, given the AP attempts in any slot the probability that there will be a collision is \(\beta_a\). Recalling, \(K\) is the maximum number of collisions a packet can experience before it is discarded. The expected number of attempts between any two success of the AP is given by:

\[ E[A_{a,1}] = 1 + \beta_a + \beta_a^2 + \ldots + \beta_a^K \quad (7) \]

Detailed derivation of the above expressions can be found in [2].

B. Attempt Probability of the STA

Let's denote \(b_j^{(k)}\) as the mean duration of the \(k^{th}\) back off duration to transmit any packet, taking into account of the AP interruption and the back off restart. So the \(E[X_{s,1}] = b_j^{(0)}\).

In each slot the AP attempts with probability \(\beta_a\), and given the STA attempts in any slot, the probability that there is a collision is \(\beta_s\).

We start from the point, where the STA has meet \(k^{th}\) collision while transmitting a packet. After this point, the STA waits for DIFS duration and samples a new backoff from the interval \((0, b_k - 1)\). As shown in the fig. 4 we have numbered the slots after DIFS duration as 0, 1, 2, 3,...

Let's assume that the STA samples the back off \(x\), which means the STA will attempt to transmit in slot number \(x\). There are three possible cases:

- AP success – In this case the AP attempts in any slot \(i \in (0, x - 1)\), probability of this event is \((1 - \beta_a)^i \beta_a\) and the number of slots counted by the STA is \(i\). Since the STA restarts the back off, so from here onwards the expected back off duration is \(b_j^{(0)}\).
- STA success – In this case the AP does not attempt in any slot \(i \in (0, x)\), probability of this event is \((1 - \beta_a)^x \beta_a\) and the number of slots counted by the STA is \(x\).
- Collision – In this case the AP attempts in slot \(x\), probability of this event is \((1 - \beta_a)^x \beta_a\). The number of slots counted by the STA is \(x\), and for \(k < K\) the

\[ \frac{\text{DIFS}}{0\ 1\ 2\ 3\ 4\ \ldots\ x-1\ x} \]

Fig. 4
expected back off duration from here onwards is \( \tilde{b}^{(k+1)} \) and for \( k = K \) it is 0.

Expression for \( \tilde{b}^{(k)} \) can be written as follows:

\[
\tilde{b}^{(k)} = \frac{b_k-1}{b_k} \left[ \sum_{x=0}^{x-1} (1 - \beta_a)^x \beta_a(i + \tilde{b}^{(0)}) \right] + \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^x \beta_a(x + I_{(k<K)} \tilde{b}^{(k+1)}) + \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^{x+1} (x)
\]  

(8)

Let us denote the following:

\[
X_k = 1 - \frac{1 - (1 - \beta_a)^{b_k}}{b_k} \\
Y_k = 1 - (1 - \beta_a)^{b_k-1} \\
Z_k = \sum_{x=0}^{b_k-1} \left[ \frac{1}{b_k} \sum_{i=0}^{x-1} (1 - \beta_a)^i \beta_a(i) \right] + \sum_{x=0}^{b_k-1} \frac{1}{b_k} [(1 - \beta_a)^x \beta_a(x) + (1 - \beta_a)^{x+1} (x)]
\]

(9)

Using the above notations, \( \tilde{b}^{(k)} \) can be written as follows,

\[
\tilde{b}^{(k)} = X_k \tilde{b}^{(0)} + Y_k \tilde{b}^{(k+1)} + Z_k \quad \text{for } k < K \\
= X_k \tilde{b}^{(0)} + Z_k \quad \text{for } k = K
\]

(10)

Using the above recursive equation, following expression can be written for \( \tilde{b}^{(0)} \):

\[
\tilde{b}^{(0)} = \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l \tilde{b}^{(0)} + \sum_{k=0}^{K} Z_k \prod_{l=0}^{k-1} Y_l = \frac{\sum_{k=0}^{K} Z_k \prod_{l=0}^{k-1} Y_l}{1 - \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l}
\]

(11)

Expected number of attempts between two success of the STA is given by the following equation:

\[
E[A_{s,1}] = 1 + \beta_a + \beta_a^2 + \ldots + \beta_a^K
\]

(12)

Detailed derivation of the above expression can be found in [2].

C. Attempt probability to transmit PS-POLL frame

Let denote \( X_{ps,m} \) the total backoff decremented by the STA for transmitting PS-POLLS, when it was trying to send its \( m^{th} \) packet. Let denote \( A_{ps,m} \) the total number of PS-POLLS sent by the STA during the period when it was trying to send its \( m^{th} \) packet. We are interested in finding out the attempt rate to send PS-POLLS, which can be written as follows:

\[
\beta_{ps} = \lim_{r \to \infty} \frac{1}{r} \sum_{m=0}^{r} A_{ps,m} = \frac{E[X_{ps,m}]}{E[A_{ps,1}]} 
\]

(13)

1) Calculation of \( E[X_{ps,1}] \): Denote \( \tilde{b}^{(k)}_{ps} \) the mean backoff duration spent in transmitting PS-POLLS, after the STA has suffered \( k \) collisions of a data packet. So \( E[X_{ps,1}] \) equal to \( \tilde{b}^{(0)}_{ps} \).

By considering the cases, as done in the previous section following gives \( \tilde{b}^{(k)}_{ps} \):

\[
\tilde{b}^{(k)}_{ps} = \sum_{x=0}^{b_k-1} p_x^{(k)} \left[ \sum_{i=0}^{x-1} (1 - \beta_a)^i \beta_a((x - i) + \tilde{b}^{(0)}_{ps}) \right] + \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^x \beta_a(0 + I_{(k<K)} \tilde{b}^{(k+1)}_{ps}) + \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^{x+1} (0)
\]

(14)

In the similar way, following gives the expression of \( \tilde{b}^{(0)}_{ps} \):

\[
\tilde{b}^{(0)}_{ps} = \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l \tilde{b}^{(0)}_{ps} + \sum_{k=0}^{K} Z_k^{(1)} \prod_{l=0}^{k-1} Y_l
\]

(15)

\[
Z_k^{(1)} = \frac{b_k-1}{b_k} \sum_{x=0}^{b_k-1} x \left[ \frac{1}{b_k} \sum_{i=0}^{x-1} (1 - \beta_a)^i \beta_a(x - i) \right]
\]

(16)

2) Calculation of \( E[A_{ps}] \): Denote \( \tilde{a}^{(k)}_{ps} \) the number of successful transmission of the AP in between two successful transmission of the STA, after the STA has suffered \( k \) collisions of a data packet. By considering the cases, as done in the previous section following equation gives the expected number of data transmissions by the AP: Expression for \( \tilde{a}^{(k)}_{ps} \) can be written as follows:

\[
\tilde{a}^{(k)}_{ps} = \sum_{x=0}^{b_k-1} p_x^{(k)} \left[ \sum_{i=0}^{x-1} (1 - \beta_a)^i \beta_a(1 + \tilde{a}^{(0)}_{ps}) \right] + \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^x \beta_a(0 + I_{(k<K)} \tilde{a}^{(k+1)}_{ps}) + \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^{x+1} (0)
\]

(17)
Following expression gives the value of $\tilde{a}_{ps}^{(0)}$:

$$
\tilde{a}_{ps}^{(0)} = \sum_{k=0}^{K} X_k \prod_{i=0}^{k-1} Y_i \tilde{a}_{ps}^{(0)} + \sum_{k=0}^{K} Z_k^{(2)} \prod_{i=0}^{k-1} Y_i
$$

$$
= \frac{\sum_{k=0}^{K} Z_k^{(2)} \prod_{i=0}^{k-1} Y_i}{1 - \sum_{k=0}^{K} X_k \prod_{i=0}^{k-1} Y_i}
$$

(18)

where, $Z_k^{(2)}$ is given by the following equation:

$$
Z_k^{(2)} = \sum_{x=0}^{b_k} \sum_{y=0}^{x-1} (1 - \beta_a)^y \beta_a (1 - \beta_a)^{b_k - y}
$$

(19)

VI. Saturation Throughput

In this section, we analyze the saturation throughput obtained by the AP and the STA. Denote $N_{ap}(t)$ and $N_{sta}(t)$ the number of successful transmissions by the AP and the STA, in the time interval $(0, t)$. Following equations give the throughput achieved by the STA and the AP:

$$
\Theta_{AP} = \lim_{t \to \infty} \frac{N_{AP}(t)}{t}
$$

$$
\Theta_{STA} = \lim_{t \to \infty} \frac{N_{STA}(t)}{t}
$$

(20)

Consider the following instants: 1) End of successful transmissions of the data packet either by the AP or the STA, 2) End of idle slots, 3) End of collisions. Let us call the $l$th such instant as $M_l$. These points are renewal points, for the process $N_{ap}(t)$ and $N_{sta}(t)$. Define $T_l = M_l - M_{l-1}$. For any $(M_{l-1}, M_l)$, define $H_{AP,l}$, it is 1 if there is a successful transmission of data packet by the AP else 0. Define $H_{STA,l}$ it is 1 if if there is a successful transmission by the STA in $(M_{l-1}, M_l)$ else 0. By applying Renewal reward theorem [12], following equations gives the saturation throughput of the STA and the AP:

$$
\Theta_{AP} = \frac{E[H_{AP}]}{E[T]} = \frac{P_{sAP}}{P_{sAP}(T_{sAP}) + P_{sSTA}T_{sSTA} + P_{idle} \delta + P_c T_c}
$$

$$
\Theta_{STA} = \frac{E[H_{STA}]}{E[T]} = \frac{P_{sSTA}}{P_{sAP}(T_{sAP}) + P_{sSTA}T_{sSTA} + P_{idle} \delta + P_c T_c}
$$

(21)

Since, a transmission of the AP is accompanied by the transmission of a PS-POLL. As discussed earlier, the STA does not have to contend to transmit a PS-POLL, but the STA still has to wait till the residual backoff of the data packet becomes 0. To account for this residual backoff, we evaluated the $\beta_{ps}$, the expected time taken to transmit PS-POLL is given by:

$$
E[T_{PSPL}] = (1 - \beta_{pspl})(\delta + E[T_{PSPL}]) + \beta_{pspl} T_{PSPL}
$$

$$
= T_{PSPL} + \frac{(1 - \beta_{pspl})}{\beta_{pspl}} \delta
$$

(22)

The above equations uses the following notations, and these uses the 802.11 parameters shown in the table [1].

- $P_{sAP}$ It is the total probability that AP wins the contention $= \beta_a (1 - \beta_a)$
- $P_{sSTA}$ It is the total probability that STA wins the contention $= \beta_s (1 - \beta_s)$
- $P_c$ It is the total probability that there is a collision $= \beta_s \beta_a$
- $P_{idle}$ It is the total probability that the channel is idle $= (1 - \beta_s)(1 - \beta_a)$
- $T_{sAP}$ It is the time spent when the AP wins contention $= SIFS + DIFS + T_{ACK} + T_{AP,d} + E[T_{PSPL}]$
- $T_{sSTA}$ It is the time spent when the STA wins contention $= SIFS + DIFS + T_{ACK} + T_{STA,d}$
- $T_c$ It is the time spent when there is a collision $= \max(T_{AP,d}, T_{STA,d}) + EIFS$

$$
T_{sPSPL} = T_{PSPL} + SIFS + DIFS + T_{ACK}
$$

$$
T_{ACK} = T_P + T_{PHY} + \frac{L_{ACK}}{C_c}
$$

$$
T_{PSPL} = T_P + T_{PHY} + \frac{L_{PSPL}}{C_c}
$$

$$
T_{AP,d} = T_P + T_{PHY} + \frac{L_{MAC} + L_{AP}}{C_d}
$$

$$
T_{STA,d} = T_P + T_{PHY} + \frac{L_{MAC} + L_{STA}}{C_d}
$$

Fig. 7: Throughput – AP and STA (11 Mbps)

VII. Simulation Results

Simulation results are obtained using NS-2.33 [13] and the various parameters used are taken from the 802.11b standard (given in Table 1). Data packet size transmitted by the AP and the STA is taken as 512 bytes. RTS threshold taken as 600 bytes, which means that the AP and the STA sends the data packet by using basic scheme. Maximum number of attempts ($K$) that the STA or AP can make to transmit a packet is...
taken as $7. b_k = 2^{5+k}$ for $0 \leq k \leq 5$ and for $5 < k \leq 7 b_k = 2^{10}$. To verify our analysis, we plotted the attempt probability of the STA and AP to transmit the data packet for varying initial range of collision window $(0, b_0 - 1)$, in the figures it is written as CWmin (minimum collision window). Figure 5a and 5b shows the attempt probability of the STA and AP to transmit data packet. we can see that the long term attempt rate of the STA is more than that of the STA, as a result of which the throughput of the AP is higher than that of the STA as shown in the Figure 7. Figure 6a shows the long term attempt rate of the STA to transmit PS-POLLs, on comparing it with figures 5a and 5b it is clear that the attempt rate of the STA to transmit PS-POLL is higher than that of the the AP and the STA. Figure 6b shows the collision probability which is the product of $\beta_a$ and $\beta_s$ as a function of the minimum collision window ($CWmin = b_0$). We can see that there is a close match between the analytical values and the simulation results which shows the correctness of our analysis.

VIII. CONCLUSION

In this paper, we have presented the analytical modeling of the saturation throughput, when a single STA is associated to the AP, and the STA is in PSM. Due to the different behavior of the STA in PSM from STA in CAM, different analysis is required. We have analyzed the long term attempt rates of the STA and the AP and using them we obtained the saturation throughputs. We have shown that our modeling is correct as the analytical results matches well with the simulation results. As far as the previous literature is concerned, it is the first work which has modeled the scenario presented in the paper. And also, we have considered a more realistic implementation of the PSM, which is different from the PSM protocol considered
in previous papers. Our future work will focus on modeling the scenarios, when there are more than one STA in PSM. We will also analyze the mixed scenario in which there are some STAs in CAM and some in PSM.

REFERENCES

[1] G. Bianchi, “Performance analysis of the ieee 802.11 distributed coordination function,” Selected Areas in Communications, IEEE Journal on, vol. 18, no. 3, pp. 535–547, Mar 2000.
[2] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, “New insights from a fixed point analysis of single cell ieee 802.11 wlan,” in INFOCOM 2005, 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, vol. 3, March 2005, pp. 1550–1561 vol. 3.
[3] Analytical models for energy consumption in infrastructure wlan stas carrying tcp traffic (submitted in comnets 2010). [Online]. Available: http://arxiv.org/abs/0909.3717v4
[4] G. Anastasi, M. Conti, E. Gregori, and A. Passarella, “Saving energy in wi-fi hotspots through 802.11 psr: An analytical model,” in 2nd Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt’04), March, 2004, pp. 227–236.
[5] H. Lei and A. A. Nilsson, “An mgf/q queue with bulk service model for power management in wireless lans,” in PE-WASUN ’05: Proceedings of the 2nd ACM international workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks. New York, NY, USA: ACM, 2005, pp. 92–98.
[6] S. Baek and B. D. Choi, “Performance analysis of power save mode in ieee 802.11 infrastructure wireless local area network,” Journal of Industrial and Management Optimization (JIMO), vol. 5, no. 3, pp. 481–492, August, 2009.
[7] P. Si, H. Ji, F. Yu, and G. Yue, “Ieee 802.11 dcf psr model and a novel downlink access scheme,” in Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE, 31 2008-April 3 2008, pp. 1397–1401.
[8] R. Krushinsky and H. Balakrishnan, “Minimizing energy for wireless web access with bounded slowdown,” Wirel. Netw., vol. 11, no. 1-2, pp. 135–148, 2005.
[9] D. Qiao and K. Shin, “Smart power-saving mode for ieee 802.11 wireless lans,” in INFOCOM 2005, 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, vol. 3, March 2005, pp. 1573–1583 vol. 3.
[10] Y. He, R. Yuan, X. Ma, J. Li, and C. Wang, “Scheduled psr for minimizing energy in wireless lans,” in Network Protocols, 2007. ICNP 2007. IEEE International Conference on, Oct. 2007, pp. 154–163.
[11] E. Tan, L. Guo, S. Chen, and X. Zhang, “Pm-throttling: Minimizing energy consumption for bulk data communications in wi-lans,” in Network Protocols, 2007. ICNP 2007. IEEE International Conference on, Oct. 2007, pp. 123–132.
[12] R. W. Wolff, Stochastic Modeling and the Theory of Queues (Paperback), Prentice Hall, 1989.
[13] ‘The network simulator ns2.” [Online]. Available: http://www.isi.edu/nsnam/ns/.

APPENDIX

1) Calculation of $E[X_{ps,1}]$: Denote $\tilde{b}_{ps}^{(k)}$ the mean backoff duration spent in transmitting PS-POLLs, after the STA has suffered $k$ collisions of a data packet. Then $\tilde{b}_{ps}^{(0)}$ is equal to $E[X_{ps,1}]$.

Lets assume that the STA samples the back off $x$, in that case there are three possible cases:

- AP success – In this case the AP should attempt in any slot $i \in (0, x-1)$. Probability that the AP attempts in $i^{th}$ slot is $(1-\beta_a)^i \beta_a$, and the number of slots counted by the STA to transmit PS-POLL is $x-i$. Since the STA will restart the back off so from here onwards the expected back off duration is $\tilde{b}_{ps}^{(x)}$.
- STA success – In this case the AP should not attempt in any slot $i \in (0, x)$. Probability of this event is $(1-\beta_a)^x + 1$ and the number of slots counted by the STA to transmit PS-POLL is 0.
- Collision – In this case the AP should attempt in slot $x$. Probability of this event is $(1-\beta_a)^x \beta_a$ and the number of slots counted by the STA is 0, and for $k < K$ the expected back off duration from here onwards is $\tilde{b}_{ps}^{(x+1)}$.

So the expression for $\tilde{b}_{ps}^{(k)}$ can be written as follows:

$$\tilde{b}_{ps}^{(k)} = \sum_{x=0}^{b_{ps}-1} p_{ps}^{(k)} \left[ \sum_{i=0}^{x-1} (1-\beta_a)^i \beta_a ((x-i)+\tilde{b}_{ps}^{(0)}) + \sum_{x=0}^{b_{ps}-1} p_{ps}^{(k)} (1-\beta_a)^x \beta_a (0+I_{(k\leq K)} \tilde{b}_{ps}^{(k+1)}) + \sum_{x=0}^{b_{ps}-1} p_{ps}^{(k)} (1-\beta_a)^x (0) \right]$$

$$\tilde{b}_{ps}^{(k)} = X_{k} \tilde{b}_{ps}^{(0)} + Y_{k} \tilde{b}_{ps}^{(k+1)} + Z_{K}^{(1)} \quad \text{for } k < K$$

$$= X_{K} \tilde{b}_{ps}^{(0)} + Z_{K}^{(1)} \quad \text{for } k = K$$

where,

$$Z_{K}^{(1)} = \sum_{x=0}^{b_{ps}-1} \left[ \frac{1}{b_{ps}} \sum_{i=0}^{x-1} (1-\beta_a)^i \beta_a (x-i) \right]$$

Using the above recursive equation, following equation can
be written for $\tilde{a}_{ps}^{(0)}$:

$$\tilde{b}_{ps}^{(0)} = \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l \tilde{b}_{ps}^{(0)} + \sum_{k=0}^{K} Z_k^{(1)} \prod_{l=0}^{k-1} Y_l$$

$$= \frac{\sum_{k=0}^{K} Z_k^{(1)} \prod_{l=0}^{k-1} Y_l}{1 - \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l}$$  \hspace{1cm} (26)$$

2) Calculation of $E[A_{ps}]$: Denote $\tilde{a}_{ps}^{(k)}$ the number of successful transmission of the AP in between two successful transmission of the STA, after the STA has suffered $k$ collisions of a data packet. Then $E[A_{ps}]$ is equal to $\tilde{b}_{ps}^{(0)}$.

Let's assume that the STA samples the back off $\tilde{a}_{ps}^{(0)}$.

- **AP success** – In this case the AP should attempt in any slot $i \in (0, x - 1)$. Probability that the AP attempts in $i^{th}$ slot is $(1 - \beta_a)^i \beta_a$ and the number of AP success is 1. Since the STA will restart the back off so from here onwards the expected number of AP success is $\tilde{a}_{ps}^{(k)}$.

- **STA success** – In this case the AP should not attempt in any slot $i \in (0, x)$. Probability of this event is $(1 - \beta_a)^x$ and the number of AP success is 0.

- **Collision** – In this case the AP should attempt in slot $x$. Probability of this event is $(1 - \beta_a)^x \beta_a$ and $k < K$ the expected back off duration from here onwards is $\tilde{a}_{ps}^{(k+1)}$ and for $k = K$ is 0.

So the expression for $\tilde{a}_{ps}^{(k)}$ can be written as follows:

$$\tilde{a}_{ps}^{(k)} = \sum_{x=0}^{b_k-1} p_x^{(k)} \left[ \sum_{i=0}^{x-1} (1 - \beta_a)^i \beta_a \left( (x - i) + \tilde{a}_{ps}^{(0)} \right) \right]$$

$$+ \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^x \beta_a \left( 0 + I_{\{k<K\}} \tilde{a}_{ps}^{(k+1)} \right)$$

$$+ \sum_{x=0}^{b_k-1} p_x^{(k)} (1 - \beta_a)^{x+1} (0)$$

$$\tilde{a}_{ps}^{(k)} = X_k \tilde{a}_{ps}^{(0)} + Y_k \tilde{a}_{ps}^{(k+1)} + Z_k^{(2)} \hspace{1cm} \text{for} \hspace{1cm} k < K$$

$$= X_K \tilde{a}_{ps}^{(0)} + Z_K^{(2)} \hspace{1cm} \text{for} \hspace{1cm} k = K \hspace{1cm} (27)$$

where, $Z_K^{(2)}$ is given by the following equation:

$$Z_K^{(2)} = \sum_{x=0}^{b_k-1} \left[ \frac{1}{b_k} \sum_{i=0}^{x-1} (1 - \beta_a)^i \beta_a (1) \right] $$

$$\hspace{1cm} (28)$$

Follwing expression gives the value of $\tilde{a}_{ps}^{(0)}$:

$$\tilde{a}_{ps}^{(0)} = \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l \tilde{a}_{ps}^{(0)} + \sum_{k=0}^{K} Z_k^{(2)} \prod_{l=0}^{k-1} Y_l$$

$$= \frac{\sum_{k=0}^{K} Z_k^{(2)} \prod_{l=0}^{k-1} Y_l}{1 - \sum_{k=0}^{K} X_k \prod_{l=0}^{k-1} Y_l} \hspace{1cm} (29)$$