

Analytical Models for Energy Consumption in Infrastructure WLAN STAs Carrying TCP Traffic

Pranav Agrawal (Beceem Communications)

Anurag Kumar (ECE, IISc)

Joy Kuri (CEDT, IISc)

Manoj K. Panda (ECE, IISc)

Vishnu Navda (Microsoft Research, India)

Ramachandran Ramjee (Microsoft Research, India)

7 January , 2010

Outline

- 1 Motivation
- 2 Traffic Model
- 3 Four Scenarios
- 4 Long File Download
- 5 Short File Download
- 6 Summary & Future Work

Motivation

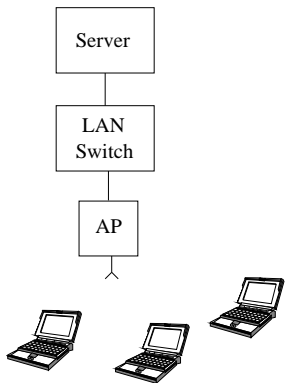
- Network activities account for considerable energy consumption in wireless access devices
- Battery-powered hand held devices are Wi-Fi enabled
- Important to model energy consumption due to TCP traffic

Traffic Model

- N STAs downloading Long files over TCP
- N STAs downloading short files over TCP, accompanied with short duration of inactivity (or *think time*) in between two downloads - Web Browsing.
 - In the case of short files download, files are the part of single *persistent* TCP connection

TCP Modeling Assumptions

- Server is assumed to be local so $RTT = 0$
- No Packet Loss
- TCP window is bounded
- No Retransmission due to Time Out
- Undelayed ACK
- No Duplicate ACKs
- Error-free reception of packets
- No hidden nodes

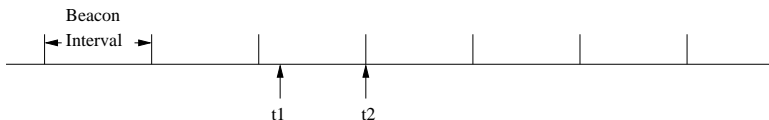


Modes of Operation

- Continuous Active Mode (CAM)
- Power Saving Mode (PSM)

Power Saving Mode (PSM)

- When STA goes to Sleep State it informs AP
- In PSM mode AP stores the packets meant for STAs
- Periodic announcement by AP about the stored packets
- STAs in PSM mode periodically wake up at beacons



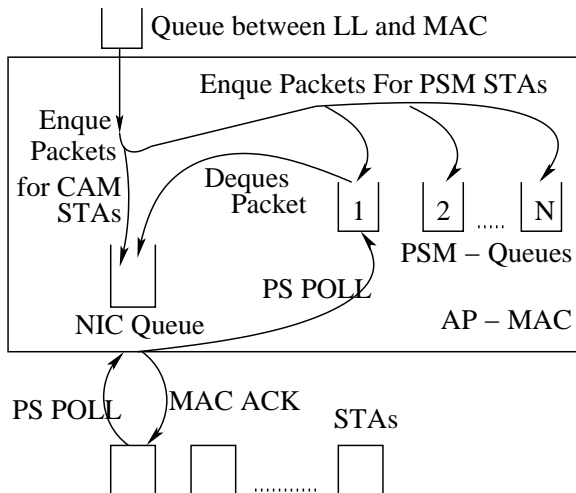
t1 – STA send http request and goes to sleep

t2 – STA wakes up and listen Beacon Frame

PSM

- STA sends a PS POLL to download packet
- AP responds with DATA packet
- “More Bit” indicates more packets stored at the AP for this STA
- STA does not go to Sleep State before downloading all packets from AP

Queue Structure at the AP & PSM Protocol



Previous Work

- In response to the PS-POLL, the AP does not send a MAC ACK, instead it sends the data packet after SIFS
- Difficult to achieve this in real implementations due to the presence of common transmission queue, which contains packet for other CAM STAs as well
- We have modeled the PSM protocol which is closer to actual implementation

Scenarios

	Long File Download	Short File Download
CAM	CAM-FTP (BCG) Model $J_{av}(N)$	Processor Sharing Model $Q_v(N)$
PSM	PSM-FTP Model $J_{av}(N)$	Processor Sharing Model $Q_v(N)$

IEEE 802.11 Model

- Approximate attempt process for detailed IEEE 802.11
 - Track the number of nodes having packets – k
 - β_k -persistent attempt process
 - β_k is obtained by Bianchi's model with k “saturated” nodes
- Given β_k – can obtain probability of slot being idle/success/collision

Long File Downloads

Metrics:

- Throughput
- Average Current
- Efficiency = $\frac{\text{Throughput}}{\text{Average Current}}$

Long File Downloads in CAM – Model

- Number of STAs = N , Max. size of TCP Window = W_{max} .
- $X_{ack}(t)$ is the number of TCP ACKs with STAs and $X_{data}(t)$ is the number of data packets with the AP at t
- Since TCP Window is bounded and $RTT = 0$ so at any instant t , $X_{ack}(t) + X_{data}(t) = NW_{Max}$
- $X_{ack}(t)$ or $X_{data}(t)$ can only change at end of a successful transmission
- In between two success instants the number of STAs contending for packets does not change
- The STA and the AP attempts in any back off slot with probability β_{k+1} if there are k STAs and the AP are contending

Long File Downloads in CAM – Model

- $X_{ack}(t)$ changes its state at ending of success instant, it can be argued that $X_{ack}(t)$ embedded at endings of success instants is DTMC
- BCG model, all TCP ACKs are distributed uniformly among all the STAs
- Probability that there is success of the AP when there are k TCP ACKs with the STAs is $\frac{1}{1+\min(k,N)}$

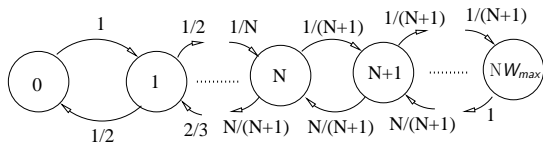


Figure: DTMC process X_m

Throughput

Using Renewal Reward Theorem

$$\Theta_N = \frac{\sum_{k=0}^N \frac{1}{k+1} \pi_k + \sum_{k=N+1}^{NW_{max}-1} \frac{1}{N+1} \pi_k}{\sum_{k=0}^{NW_{max}} \pi_k E_k[T]} \quad (1)$$

where, π_k is the stationary probability of k STAs contending in a cycle and $E_k[T]$ is the expected time until the end of the next success, when the number of STA at the beginning is k .

Average Current

- Transmitting State - When STA is transmitting
- Receive and Decode state - When STA is listening to the ongoing transmission and also decoding it
- Listen State (possible due to NAV) - When STA is listening to the ongoing transmission but not decoding it
- Idle State - When the channel is idle
- Sleep State - Low power state in which STA is not sensing the medium

Calculation of Average Current

$$J_{av} = \Phi_{Tx} J_{Tx} + \Phi_{RxD} J_{RxD} + \Phi_{Ls} J_{Ls} + \Phi_{Id} J_{Id} + \Phi_{SI} J_{SI} \quad (2)$$

Long File Download in PSM - Model

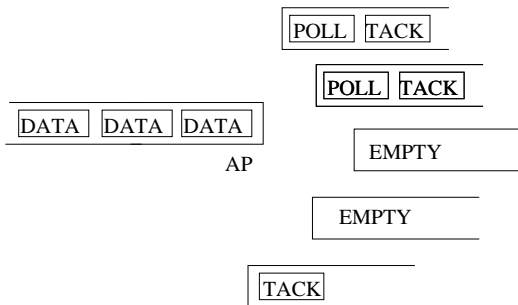
Large number of STAs (> 5)

- Every two packets (1 TCP ACK + 1 PS-POLL) sent by each of the STAs, N packets need to be transmitted by the AP
- IEEE 802.11 is packet fair
- Small number of STAs contend at any time
- Most of the TCP window reside at the AP
- Packets backlogged at the AP so “More” bit is always set
- PS-POLL enqueued at the HOL position and the TCP ACK at the end of the queue

Long File Download in PSM - Model ($N > 5$)

- 1 DATA packet goes to an empty STA and the total number of contending STAs increases by one.
- 2 Some STAs contend to send PS-POLLs and some TCP-ACKs.
- 3 When a STA successfully transmits a PS-POLL, the number of STAs contending to send TCP-ACKs increases by one.
- 4 When a STA successfully transmits a TCP ACK, the number of STAs contending decreases by one.

Long File Download in PSM - Model ($N > 5$)



- X = No. of STAs with PS POLL
- Y = No. of STAs with TCP ACK
- AP always Saturated

Long File Download in PSM - Model ($N > 5$)

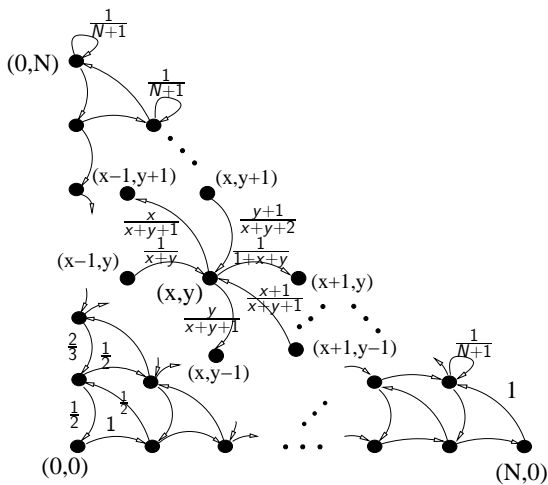


Figure: 2-Dimensional DTMC process (X_m, Y_m)

Long File Download - Aggregate Throughput

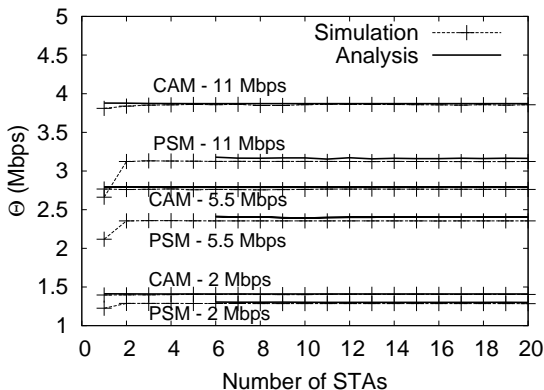


Figure: Aggregate Throughput - (CAM & PSM)

Long File Download - Average Current

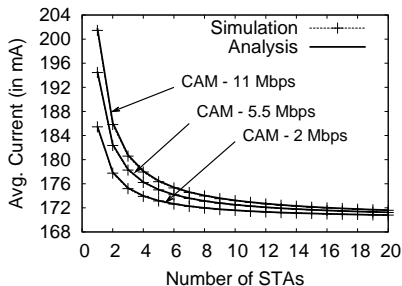


Figure: CAM

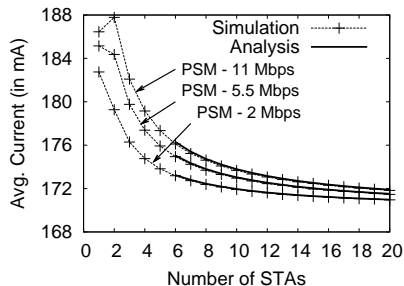


Figure: PSM

Long File Download - Efficiency

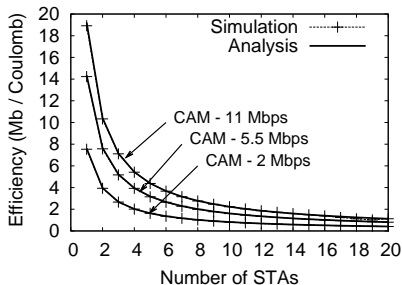


Figure: CAM

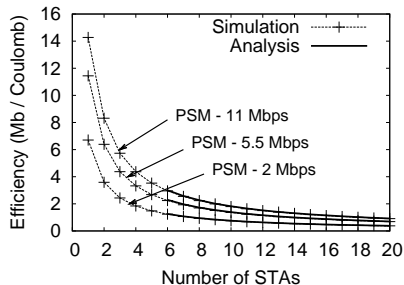


Figure: PSM

Short File Download - Traffic Model

- Constant number of STAs N
- STA sends request for the file, downloads it and then remains in think state for some time and then again sends a request.
- All file downloads are part of single TCP connection.
- Processor Sharing Model: since no preference is given to any STA each STA gets service at equal rate.
- File size is exponentially distributed with mean L .
- Service time is exponentially distributed with mean $\frac{L}{\Theta_N}$, where Θ_N is the obtained from the long file downloads.
- Think time is exponentially distributed with mean $\frac{1}{\mu}$.

Short File Download - Metrics

- Average charge ($E[Q_f]$) per file – total *charge* drawn by all the STAs in a given interval divided by the total number of files downloaded in the same interval.
- Average sojourn time ($E[S]$) – total *time* taken by all the files downloaded in a given interval divided by the total number of files downloaded in the same interval.

Short File Download in CAM - Model

Let $X(t)$ denote the number of STAs downloading at any instant t .

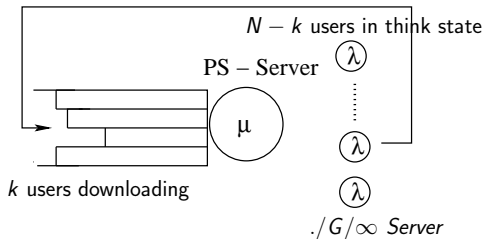


Figure: Closed queueing network model for short file downloads.

Short File Download in CAM - Model

$X(t)$ is a CTMC

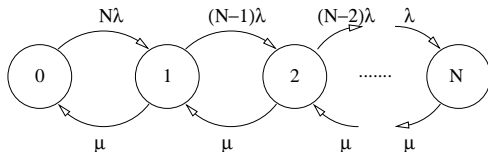


Figure: Transition rate diagram of $X(t)$

Short File Downloads in CAM - Expected charge drawn per file

$$E[Q_f] = \frac{\sum_{k=0}^N \pi_k [kJ_{k,a} + (N-k)J_{k,p}]}{\sum_{k=0}^N \pi_k (N-k)\lambda} \quad (3)$$

where, π_k is the stationary probability of k STAs downloading files and $(N-k)$ STAs in think state. Let $J_{k,a}$ be the average current drawn by k STAs when they are downloading long files, $J_{k,p}$ be the average current drawn by a STA listening to the traffic of k STAs downloading long files.

Short File Downloads in CAM - Expected Sojourn Time

Using Little's Theorem, following expression can be written for expected time to download a file:

$$E[S] = \frac{\sum_{k=0}^N k\pi_k}{\sum_{k=0}^N \pi_k(N-k)\lambda} \quad (4)$$

Short File Downloads in PSM - Model

- When the user requests a file, the STA wakes up and sends a HTTP request packet and then again goes back to sleep.
- The server is local to the LAN, so packets from the server arrive immediately at the AP
- This information is sent to the STA in the next beacon frame, from here onwards STA remains awake till file download completes
- Let X_m denote the number of STAs in the download state at m^{th} the beacon instants, it is a DTMC
- Number of users that complete their think times in a beacon interval do not depend on the number of users who complete their transfers in the same beacon interval

Short File Downloads in PSM - Model

- Let k STAs downloading files at the start of a beacon interval.
- So $N - k$ STAs are in think state, probability that i users finish their think times within the beacon interval is $Bi(N - k, i) = \binom{N-k}{i} (1 - e^{-\lambda b})^i (e^{-\lambda b})^{N-k-i}$.
- Let $q(k, j, b)$ be the probability that j users complete their downloads out of k active users within the time interval of b .

The transition probability (p_{k,k_1}) of process X_m :

$$p_{k,k_1} = \sum_{j=\max(0,k-k_1)}^{\min(i,N-k_1)} q(k,j,b) Bi(N-k, k_1 - k + j) \quad 0 < k \leq N, 0 \leq k_1 \leq N$$

$$\binom{N}{k_1} (1 - e^{-\lambda b})^{k_1} (e^{-\lambda b})^{N-k_1} \quad k = 0, 0 \leq k_1 \leq N$$

Short File Downloads in PSM - Expected Sojourn Time

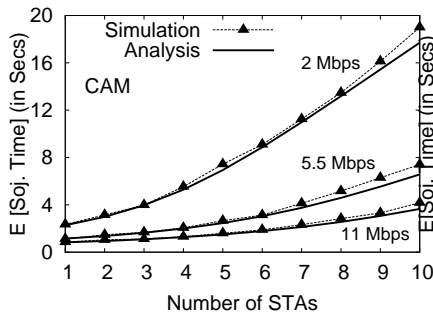


Figure: CAM

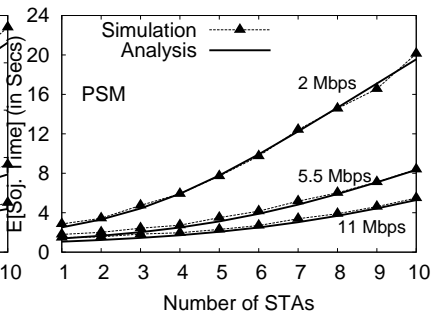


Figure: PSM

Short File Download - No. of File Downloads per 100 C

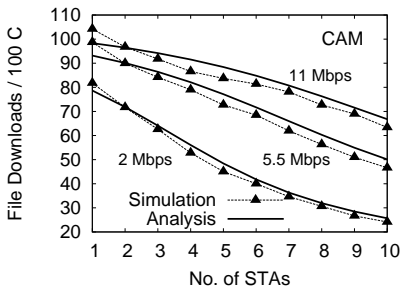


Figure: CAM

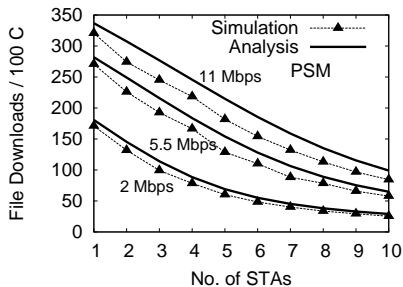


Figure: PSM

Conclusion and Future Work

- Analyzing scenarios involving UDP traffic.
- Performance degrades with increasing number of STAs

Thank You