Access Delay Distribution Estimation in 802.11 Networks

Avideh Zakhor

Joint work with: E. Haghani and M. Krishnan
Motivation
- Delay sensitive traffic should be delivered within delay bound
- Increasing deployment of 802.11 networks and users affect the delay performance

Access delay:
- Time between a new packet arriving at the head of line of MAC queue and receiving its ACK

PMF of access delay at a node can determine impact of:
- Maximum retransmit limit on delay distribution
  - optimum retransmit strategy
- Sending rate on delay distribution without trial and error via measurements

Objective:
- Propose a distributed framework to estimate the uplink (UL) access delay distribution for each node using only local measurements
Busy/Idle Signal as Local Measurement

- Each node/AP collects binary-valued ‘busy-idle’ (BI) signal
  - 1 when local channel is occupied, 0 otherwise
- AP broadcasts its BI signal periodically \( \Rightarrow \sim 14\text{kb/s}, 3\%\) overhead
- Nodes use their BI signal along with AP’s to estimate \(P_C\)

A station senses the channel before transmission
  - Transmits if channel idle for DIFS

Initiates a counter if channel busy:
  - Counter value uniformly distributed [0,CW-1]
  - Counts down for every idle time slot
  - Pauses counting during busy time slots and resumes after channel is idle for:
    - DIFS if station decoded headers of packets in last busy period
    - EIFS (>DIFS) if could not decode the headers
  - Doubles the CW for unsuccessful transmissions
  - Resets the CW to CWmin if successful
Access Delay at $m^{th}$ Tx Attempt

- $d_H^m$: access delay of a packet at $m^{th}$ Retx attempt
  \[ d_H^m = I_1^m + B_1^m + \cdots + B_{N_m-1}^m + I_{N_m}^m \]
- $I_i^m$: length of $i^{th}$ idle period at $m^{th}$ Retx attempt
- $B_i^m$: length of $i^{th}$ busy period at $m^{th}$ Retx attempt
- $N_m$: number of idle periods required for $m^{th}$ Retx attempt
- $\delta_i$: interframe spacing time after $(i-1)^{th}$ busy period: DIFS or EIFS
- $J_i^m = \max(0, I_i^m - \delta_i^m)$ amount backoff counter decremented at $i^{th}$ idle period
- $W_m$: value of backoff counter at $m^{th}$ Retx attempt

\[ \sum_{i=1}^{N_m} J_i^m = W_m \]

Packet arrives at HoL of MAC after $m-1$ failed Tx attempt
Distribution Estimation Process

- Each node empirically estimates distributions of $B_i$, $I_i$ and $\delta_i$:
  \[ \Pr\{\delta_i = DIFS\} = 1 - \Pr\{\delta_i = EIFS\} \]

- Recall \( \sum_{i=1}^{N_m} J_i^m = W_m \)

- \( N_m \) is the first passage time of a variable \( S_n \triangleq \sum_{i=1}^{n} J_i \) at \( m^{th} \) Retx

\[
N_m = \arg \min_n S_n \geq w_m = P\{S_{n-1} < w_m\} - P\{S_n < w_m\}
\]

- PMF for \( d_H^m \) delay access distribution at \( m^{th} \) retransmission

\[
f_{d_H^m}(d) = \frac{1}{CW_m} \sum_{w_m=0}^{CW_m-1} \sum_{n=1}^{\infty} \Pr\left\{ \sum_{i=1}^{n-1} (\delta_{i+1} + B_i) = d - w_m - DIFS \right\} \times \Pr\{N_m = n\}
\]

\( CW \) contention window size at \( m^{th} \) Retx

\[
\overline{\delta_i} \triangleq J_i - I_i = \min(\delta_i, I_i)
\]
Empirical Estimates of Distributions over 3 sec. interval

PMF of $I_m$

PMF of $B_m$

PMF of $N_m$

$W_m=100$

$W_m=1000$
Distribution of Total Access Delay

- Total access delay: channel access time plus
  \[ T = \text{packet air TX time} + \text{ACK timeout} \]
  \[ d_H = B_0 + (d_H^1 + T) + \cdots + (d_H^M + T) \]
  \( M \) total # of Retx \( 1 \leq M \leq \gamma \)
  \( \gamma \) max Retx limit
- The PMF of the total access time given \( \gamma \)

\[
\Pr(d_H = \tau) = \sum_{m=1}^{\gamma} \Pr\{d_H = \tau \mid M = m\} \times \Pr\{M = m\}
\]

\[
\Pr\{d_H = \tau \mid M = m\} = \Pr\{B_0 + d_H^1 + \cdots + d_H^m = \tau - (m - 1) \cdot T\}
\]
Total MAC layer delay

- Total delay at MAC layer: queuing delay + access delay
  \( \theta_p = \) arrival time of packet \( p \) at MAC queue
  \( A_p = \) arrival time of ACK for packet \( p \)
  \( d_{Hp} = \) access delay for packet \( p \)

\[
A_p = \max(A_{p-1}, \theta_p) + d_{H_p}
\]

- Use distributions of \( \theta_p \) and \( d_{Hp} \) to estimate steady-state distribution of \( A_p \)

- \( E_p = \) Total MAC Delay for packet \( p \):
  \[
  E_p = A_p - T_{ACK} - \theta_p
  \]

- Probability of successful arrival of packet \( p \) within delay bound:
  \( \Gamma_p = P(E_p < \text{delay limit}) \)

- Effective throughput = steady state value of \( \Gamma_p \)

\[
\eta = \lim_{p \to \infty} \frac{1}{p} \sum_{i=1}^{p} \Gamma_i
\]
Simulation Results: Access Delay

- A 50 node and 7 AP simulation set up
- Saturated network traffic
- A node collects BI information for 3 sec
- Access delay for 200 sec simulation

![Graphs showing access delay for different values of $\gamma$]

$\gamma = 1$

$\gamma = 8$
Simulation Results: Effective Throughput

- A 50 node and 7 AP simulation set up
- Saturated network traffic
- A node collects BI information for 3 sec
- 200ms delay limit
- Compare effective throughput for varying $\gamma$ and input rate for each node
Simulation Results: Retransmit Limit

- A 50 node and 7 AP simulation set up
- Three sample simulations in each:
  - One node sends traffic at 200Kbps, others send saturated traffic
  - The node collects BI information for 3 sec
- Compare effective throughput for varying $\gamma$
- Estimates within 5% margin of simulation results
A 50 node and 7 AP simulation set up
A sample sends traffic at various CBR others send saturated traffic
Compare effective throughput for varying \( \gamma \)
Using optimum retransmit limit, node can reach 90% effective throughput at various rates
Future Work: AP selection in 802.11n

- Currently AP selection based on signal strength
- Need to optimize:
  - Send opportunities – affected by traffic saturation at AP
  - Probability of success – affected by SINR and set of hidden nodes
- Can evaluate these quantities via BI signal.
- Before a node joins an AP it should listen to the B/I signal sent by two or more APs to determine the “best” AP to join.
AP Selection: Example 1

- If choosing AP based on signal strength, all nodes will use same AP
- Network throughput can be improved if some nodes choose to connect to AP 2 instead
AP Selection: Example 2 - Beamforming

- Without beamforming, node 2 must contend with node 1
  - With beamforming, it can send simultaneously
- Upshot: beamforming allows optimum AP selection
  - Catch: need to know network topology

![Diagram of AP Selection Example 2: Beamforming](image-url)
Beamforming and BI signals

- 802.11n supports beamforming
  - Explicit beamforming: nodes share channel information to help determine beamforming direction or "steering"
- Knowledge of direction of received signals → directional BI signal
- Quantize direction into sectors to create directional BI signal
- During promiscuous listening, can implicitly estimate direction of transmitter
- Angles only known modulo 180 degrees
- APs periodically broadcast BI signal to nodes
Illustration of Directional BI signal
Example of topology reconstruction (1)

- AP and N1 know relative positions
- AP and N2 know relative positions
- N1 can sense N2 when N2 is sending to AP.
Example of topology reconstruction (2)

- Individually, N1 and AP have ambiguity as to location of N2

![Diagram showing the positions of N1, N2, and AP with possible locations marked as 2 and 3.]

Possible location of N2 to N1
Possible location of N2 to AP
Example of topology reconstruction (2)

- N1 can compare its BI signal to that of AP
  - AP’s BI signal at $\theta_0$ is similar to N1’s BI at angle $\theta_3$ → with high probability N2 is at location 2.
Contention Window Adaptation

- CSMA – nodes sense channel and send when free
  - Problem – all nodes would start simultaneously
  - Solution: random backoff using “contention window” (CW)
- Standard
  - Random backoff \( W \sim U(0, CW-1) \)
  - CW adapts over time
    - Packet fail: \( CW \leftarrow \min(\alpha \cdot CW, CW_{\text{max}}) \)
    - Packet success: \( CW \leftarrow CW_{\text{min}} \)
    - 802.11: \( \alpha = 2, CW_{\text{min}} = 32, CW_{\text{max}} = 1024 \)
Problem

- Implicit assumption that all losses are due to collision
- Example:
  single node with poor channel conditions $\rightarrow$ high loss $\rightarrow$ large CW $\rightarrow$ poor channel utilization
  - CW=512 time slots @ 20 $\mu$s = 10ms
  - 2KB packet @ 11 Mbps < 2ms
  - Could have had 5 more attempts!
Example: Symmetric topology

- Nodes all hidden to each other & equidistant from AP
- Different curves of same color are different noise levels i.e. Pe
- Optimal $W$ increases with # hidden nodes
- Pe does not affect optimal $W$
How can this be improved?

- With estimate of collision probability, $P_C$, from [1], can optimize parameters based on collision probability rather than only recent loss history.

- Tunable parameters:
  - $\alpha$
  - $CW_{\text{min}}$
  - $CW_{\text{max}}$
  - Distribution of random backoff

- Plan: choose new parameters every 5 seconds based on observations over 5 seconds, primarily BI signal.
Analysis

- Maximize sum of log throughput – enforces fairness

\[
U = \sum_j \log TP_j
\]

\[
TP_i = R_i S_i (1 - P_{SC}) (1 - P_{DC}) (1 - P_e)
\]

\[
S_i = \frac{1}{L_i + \overline{W}_i/I_i}
\]

\[
P_{DC} = 1 - \prod_{k \in \mathcal{N}_i} \left( 1 - \frac{1}{\overline{W}_k} \right)
\]

\[
P_{SC} = 1 - \prod_{j \in \mathcal{H}_i} \left( 1 - \frac{L_i + L_j - 1}{L_j + \overline{W}_j} \right)
\]

\[U = \text{utility, } TP_i = \text{throughput } R_i = \# \text{ bits per packet,}
\]

\[S_i = \text{fraction of slots in which node } i \text{ starts sending,}
\]

\[I_i = \text{proportion of time channel is idle, } L_i = \text{packet length in slots,}
\]

\[\overline{W}_i = \text{average contention window size, } P_e = \text{Pr\{channel error\},}
\]

\[P_{DC} = \text{Pr\{direct collision\}, } P_{SC} = \text{Pr\{staggered collision\},}
\]

\[\mathcal{N}_i = \text{set of neighboring nodes, } \mathcal{H}_i = \text{set of hidden nodes}\]
Future work

- Investigate the use of B/I signal for AP selection
- Develop directional B/I signal for beamforming 802.11n situations ➔ optimal AP selection
- Optimizing contention window parameters using B/I signal.
- Investigate packet aggregation for 802.11n using B/I signal.