PNOFA: Practical, Near-Optimal Frame Aggregation for Modern 802.11 Networks
Background

- DIFS
- Back off
- DATA
- SIFS
- ACK
Physical-layer bit rate: $\infty$ bps $\rightarrow$ MAC-layer throughput $\approx 50$ Mbps
Background

DATA  ACK  DATA  ACK  DATA  ACK
Background
How many MPDUs should be aggregated?
Throughput vs #MPDUs

Should we aggregate as many as possible?
Frame Aggregation is More Complicated!
Channel Correction Limitation

Channel Estimation

Channel Correction

![Diagram showing MPDU packets and their channel estimation and correction process.](image)
Modeling Optimal A-MPDU Length

\[ T_{\text{put}}(n) = \frac{B N_s(n)}{T_{A-\text{MPDU}}(n)} \]

- Payload size
- # of successful MPDUs
- A-MPDU transmission time
- Total overhead
- MPDU transmission time
Statistically Optimal Algorithm

Throughput (Mbps)

A-MPDU size

1  16  32  48  64
Practical, Near-Optimal Frame Aggregation

• How does PNOFA estimate MDRs?
Practical, Near-Optimal Frame Aggregation

• Operation of PNOFA
Practical, Near-Optimal Frame Aggregation

• How to know when to increase the A-MPDU size?
Evaluation of PNOFA

• Trace-based evaluation
  • Comparison with statistically optimal algorithm
  • Comparison with state-of-the-art algorithms

• Experimental evaluation
  • Implementation on Google Wifi APs
Related Work

MoFA

- Dynamically adjusts A-MPDU size
  - Compares the error rate in the first & seconds half of A-MPDU

STRALE

- Handles channel correction limitation by
  - Adjusting A-MPDU size similar to MoFA
  - Adjusting transmission rate
Trace-Based Evaluation: Scenarios

Access Point (ath9k)  WiFi device
TP-Link WDN4800
Intel AC 3160
TP-Link WDN4200
TP-Link T9UH
Trace-Based Evaluation: Results

![Graph showing throughput over time for different systems](image)

- **Throughput (Mbps)**
- **Time (seconds)**
- Systems: ath9k, STRALE, MoFA, PNOFA, SO
Experimental Evaluation: Implementation

User space
- tcpdump
- PNOFA
- iw

Kernel space
- B-ACK
- Tx Rate
- FA Settings
- ATH10K driver

Hardware
- IPQ4019
- Firmware
Experimental Evaluation: Methodology

PNOFA → Moved at walking speed
Experimental Evaluation: UDP Results

17% average improvement
Conclusions

• PNOFA
  • Near optimal performance
  • Practical (implemented on Google Wifi using tcpdump)
  • Outperforms state-of-the-art algorithms