Enabling TDMA for Today’s Wireless LANs

Zhice Yang\textsuperscript{1*}, Jiansong Zhang\textsuperscript{2*}, Kun Tan\textsuperscript{2},
Qian Zhang\textsuperscript{1}, Yongguang Zhang\textsuperscript{2}

\textsuperscript{1}CSE, Hong Kong University of Science and Technology
\textsuperscript{2}Microsoft Research Asia
\*Co-Primary Author
Motivation
Motivation
Motivation
Motivation

• Limitations of Distributed Coordination Function (DCF) in Current Wireless LANs
  • Weak Interference Management
  • Inefficient Channel Access
  • Lacking Guarantee in QoS
Motivation

- Limitations of Distributed Coordination Function (DCF) in Current Wireless LANs
  - Weak Interference Management
  - Inefficient Channel Access
  - Lacking Guarantee in QoS
- Demands for Higher-Efficiency Wireless Network
  - Proliferation of Wireless Devices
  - Emerging Network QoS Sensitive Applications
How to Fill the Gap?

Backhaul Network

AP1

ClientA

AP2

ClientB

ClientC

AP1

AP2

B

B

B

B

C

C

C

C

Time
How to Fill the Gap? -- TDMA

• Arrange the transmission of all the wireless packets in the air to
  • Manage interfering transmissions
  • Reduce contention overhead
  • Provide priorities for QoS transmissions
Research Question

• Arrange the transmission of all the wireless packets in the air to
  • Manage interfering transmissions
  • Reduce contention overhead
  • Provide priorities for QoS transmissions

Is TDMA possible with commodity WLAN devices?
Feasibility for TDMA in WLAN

• Time Synchronization
  • Backhaul network

• Scheduling
  • Central controller

Exist
Architecture Borrowed from SDN

OpenTDMF Controller

OpenTDMF AP1

ClientA

OpenTDMF AP2

ClientB

ClientC

Backhaul Network

Ethernet

Actions

Status

Actions

Status
Architecture Borrowed from SDN

ClientA

OpenTDMF
AP1

ClientB

OpenTDMF
AP2

ClientC

Ethernet

Actions
Status

Controller

AP2<->ClientC in 10 slots

Time

... 10 11 12 ...

C
Architecture Borrowed from SDN

ClientA

OpenTDMF AP1

Controller

OpenTDMF AP2

ClientB

Ethernet

Actions
Status

Actions
Status

ClientC

AP2<-->ClientC in 12 slots

... 10 11 12 ...

Time
Architecture Borrowed from SDN

ClientA

OpenTDMF

AP1

Status

Ethernet

Status

OpenTDMF

AP2

ClientB

ClientC

AP1<->ClientB in 11

AP2<->ClientC in 10,12

OpenTDMF Controller
Architecture Borrowed from SDN
Challenges

• Time Synchronization

Commodity WLAN devices lack means for accurate synchronization

Local Time of AP1: 10 11 12
Local Time of AP2: 10 11 12

~10μs
Challenges

• Time Synchronization

Commodity WLAN devices lack means for accurate synchronization

• Uplink Scheduling Enforcement

Commodity WLAN devices is designed for distributed access and determines channel access independently
Outline

• OpenTDMF Design
  • Time Synchronization
  • Uplink Scheduling Enforcement
• Experiment Results
• Scheduling Examples
• Conclusion
Outline

• OpenTDMF Design
  • Time Synchronization
    • Uplink Scheduling Enforcement
• Experiment Results
• Scheduling Examples
• Conclusion
Backhaul Time Synchronization

• Using IEEE 1588 Precise Time Protocol (PTP) to Synchronize the Wired APs
  • Assumption: The network delay is symmetrical
The Problem

• Large Variation
Analyze the Problem

• The Architecture of the Commodity AP Introduces Variance in the Delay Measurement
Analyze the Problem

• The Architecture of the Commodity AP Introduces Variance in the Delay Measurement
## Our Solution

<table>
<thead>
<tr>
<th></th>
<th>Delayed PTP Packet</th>
<th>Normal PTP Packet</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Before Port0</strong></td>
<td>![Data][1][PTP]</td>
<td>![Data][1][PTP][Data][Data][PTP]</td>
</tr>
<tr>
<td><strong>After Port0</strong></td>
<td>![Data][1][PTP][Data][PTP][Data][Data][PTP]</td>
<td>![Data][1][PTP][Data][Data][PTP][Data][Data][PTP]</td>
</tr>
<tr>
<td>$T_\Delta = T_{data}$</td>
<td></td>
<td>$T_\Delta &gt; T_{data}$</td>
</tr>
</tbody>
</table>
Scheduling Enforcement

• Use busy waiting to ensure accurate software timer event

• Use transmission gate handler in WiFi chip to ensure accurate transmission control
Outline

• OpenTDMF Design
  • Time Synchronization
  • Uplink Scheduling Enforcement

• Experiment Results
• Scheduling Examples
• Conclusion
Uplink Control

• Polling for Uplink Transmission
  • Poll packet is a normal packet with a poll flag in the control field of the MAC header
  • Clients response the poll packet with the uplink data
  • Uplink can be treated as downlink
First Transmission Problem

• AP needs to know about the packet queue information in clients to schedule polling
  • Clients piggyback queue information in every uplink packet
• AP doesn’t know when the client want to transmit the first uplink packet
Group Polling

• Group polling for the first transmission
  • Group poll packet is a poll packet with group address

<table>
<thead>
<tr>
<th></th>
<th>Client1</th>
<th>Client2</th>
<th>Client3</th>
<th>Client4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random SIFS</td>
<td>+0 slots</td>
<td>+3 slots</td>
<td>+1 slots</td>
<td>+4 slots</td>
</tr>
</tbody>
</table>
Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

<table>
<thead>
<tr>
<th>Random SIFS</th>
<th>Client1</th>
<th>Client2</th>
<th>Client3</th>
<th>Client4</th>
</tr>
</thead>
<tbody>
<tr>
<td>+0 slots</td>
<td>+3 slots</td>
<td>+1 slots</td>
<td>+4 slots</td>
<td></td>
</tr>
</tbody>
</table>
Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

<table>
<thead>
<tr>
<th></th>
<th>Client1</th>
<th>Client2</th>
<th>Client3</th>
<th>Client4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random SIFS</td>
<td>+0 slots</td>
<td>+3 slots</td>
<td>+1 slots</td>
<td>+4 slots</td>
</tr>
</tbody>
</table>

Diagram:
- Group Poll
- Client1 Info
- Client2 Info
- Client3 Info
- Client4 Info
Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

<table>
<thead>
<tr>
<th></th>
<th>Client1</th>
<th>Client2</th>
<th>Client3</th>
<th>Client4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random SIFS</td>
<td>+0 slots</td>
<td>+3 slots</td>
<td>+1 slots</td>
<td>+4 slots</td>
</tr>
</tbody>
</table>

![Diagram showing group polling and SIFS intervals for different clients]

Time
Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

<table>
<thead>
<tr>
<th></th>
<th>Client1</th>
<th>Client2</th>
<th>Client3</th>
<th>Client4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random SIFS</td>
<td>+0 slots</td>
<td>+3 slots</td>
<td>+1 slots</td>
<td>+4 slots</td>
</tr>
</tbody>
</table>

```plaintext
Group Poll
[Group Poll]
[Client1 Info]
[Client3 Info]
[Client2 Info]
[Client4 Info]
```

Time
Outline

• OpenTDMF Design
  • Time Synchronization
  • Uplink Scheduling Enforcement

• Experiment Results

• Scheduling Examples

• Conclusion
Implementation

• Implemented with TP Link 4900 with Atheros AR9381 and AR9580 WiFi Chip
• Modified ath9k driver
• Modified linuxptp program
Timing Error

Backhaul Network

- AP1
- AP2
- AP3

- ClientA
- ClientB
- ClientC

Graph showing Timing Error (μs) versus Ethernet Background Traffic (Mbps) with Mean, Median, and Max lines.
Uplink Efficiency

Spectrum Efficiency

Number of Clients

Legacy, 2ms agg.
Polling, 2ms agg.

30%
Outline

• OpenTDMF Design
  • Time Synchronization
  • Uplink Scheduling Enforcement
• Experiment Results
• Scheduling Examples
• Conclusion
# Topology and Policy

![Network Diagram]

### Knowledge of the Controller

AP1\(\leftrightarrow\)B conflicts with AP2\(\leftrightarrow\)C & AP1\(\leftrightarrow\)B requires high priority

<table>
<thead>
<tr>
<th>Control Policy for AP1</th>
<th></th>
<th>Control Policy for AP2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Flow ID</strong></td>
<td><strong>Time Slots</strong></td>
<td><strong>Priority</strong></td>
</tr>
<tr>
<td>AP1(\leftrightarrow)B</td>
<td>1,2 mod 3</td>
<td>High</td>
</tr>
<tr>
<td>AP1(\leftrightarrow)A</td>
<td>ALL</td>
<td>Normal</td>
</tr>
<tr>
<td>AP2(\leftrightarrow)C</td>
<td>0 mod 3</td>
<td>Normal</td>
</tr>
</tbody>
</table>
Outline

• OpenTDMF Design
  • Time Synchronization
  • Uplink Scheduling Enforcement
• Experiment Results
• Scheduling Examples
• Conclusion
Conclusion

• Thoughtful study of accurate synchronization in commodity AP

• Enable polling based uplink transmission in commodity WiFi chips

• Build the OpenTDMF system and validate the feasibility of TDMA in commodity WLANs.
Thank you!