

High-Priority Timeout for P-EDCA Operation

Date: July 2025

Authors:

Name	Affiliations	Address	Phone	email
Behnam Dezfouli	Nokia	520 Almanor, Sunnyvale, CA		behnam.dezfouli@nokia.com
Davis Robertson				
Mikhail Liubogoshchev				
Klaus Doppler				
Salvatore Talarico				
Kerstin Johnsson				

Introduction

- EDCA is the primary access method for STAs to reserve a TXOP, especially to send aperiodic, event-driven traffic
- EDCA struggles when multiple STAs with Low-Latency (LL) traffic (AC_VO) compete, or when LL STAs contend with AC_BE STAs
- Lowering the tail-time latency of STAs competing for channel access through EDCA has been addressed in several contributions [11-24/1918][11-24/1144][11-24/0864]
- The High-Priority (HiP) EDCA mechanism [11-24/1918][11-24/1144] allows STAs with LL traffic to send Defer Signal (DS) frame after a certain number of retries, and then use RTS/CTS to reserve a TXOP
 - **Shortcoming:** Using the CTS/ACK Timeout duration after sending response-soliciting frames leads to wasted channel time before successful channel reservation
- **In this contribution, we propose a method to reduce the overhead of the P-EDCA procedure by minimizing the time required to detect transmission failure after sending a response-soliciting frame**
- **A STA is allowed to use a High-Priority Timeout (HPTO) duration instead of CTS/ACK Timeout to retry channel access during P-EDCA**

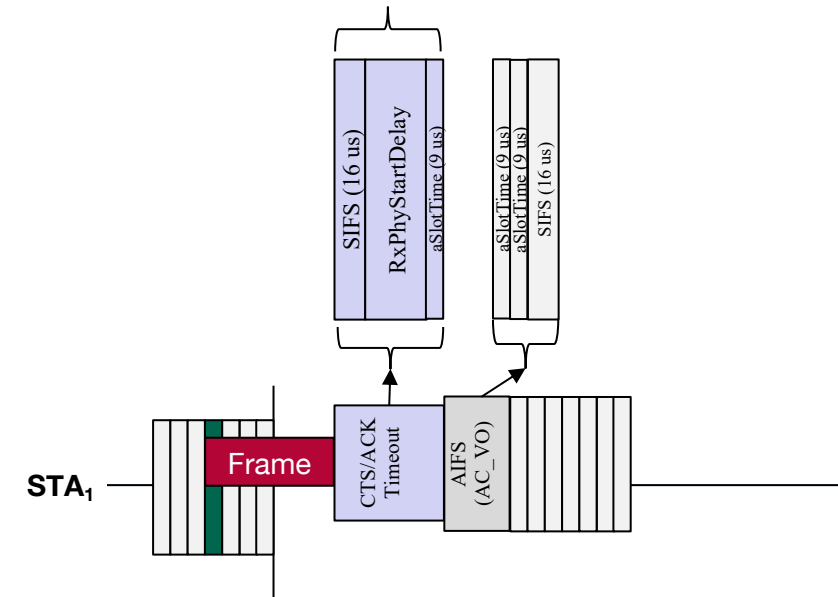
ACK Timeout in 802.11

- When a STA sends a **response-soliciting frame (e.g., RTS)**, it needs to wait for **CTS/ACK Timeout + AIFS[AC]** before competing for channel access again
 - The ACK Timeout duration depends on the PHY layer

For example, for **non-HT** frames:

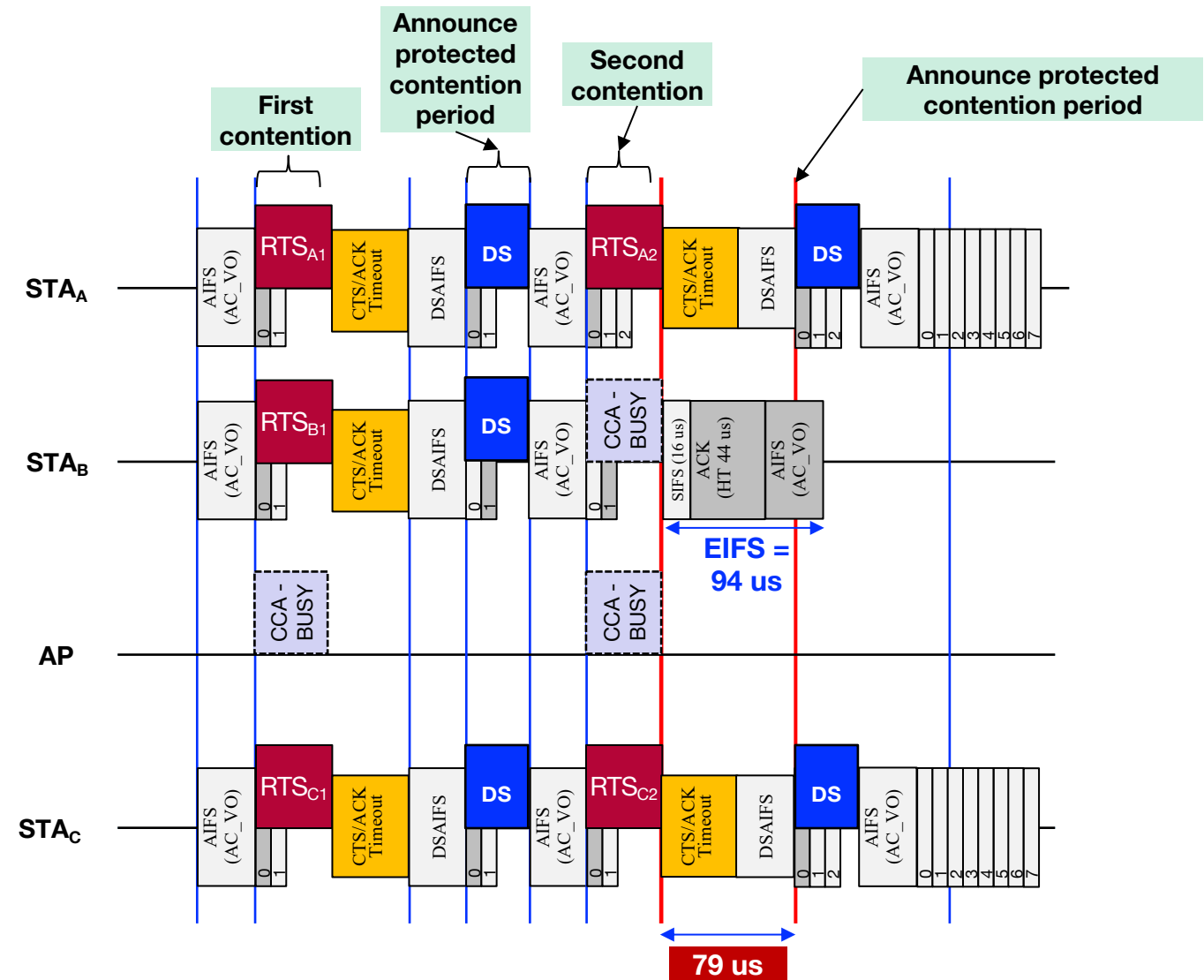
$$16 \text{ (SIFS)} + 20 \text{ (RxPhyStartDelay)} + 9 \text{ (aSlotTime)} = \mathbf{54 \text{ us}}$$

	RxPhyStartDelay (us)
OFDM	20
HT	28 (HT-mixed), 24 (HT-greenfield)
VHT	$36 + 4 \times N_{\text{VHT-LTF}} + 4$
EHT	$32 + 4 \times N_{\text{EHT-SIG}}$ for EHT MU PPDU 32 μ s for EHT TB PPDU



□ Sample Scenario

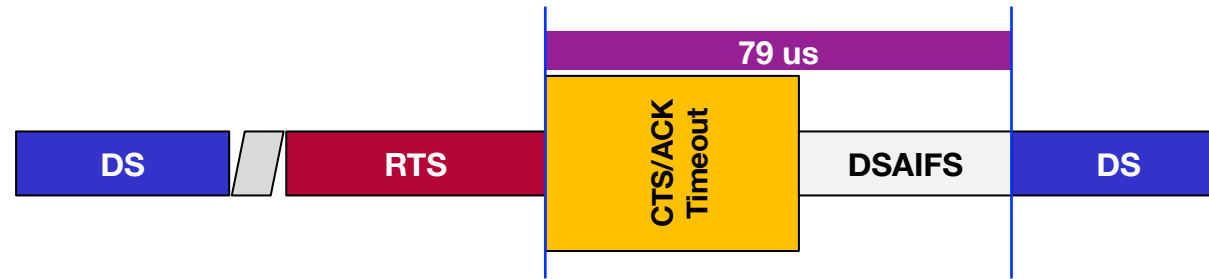
- **First contention round:** STA_A, STA_B, and STA_C compete; all three STAs use CTS/ACK Timeout after sending their frames
- STA_A, STA_B, and STA_C send DS frames to announce **protected contention period**
- **Second contention round (protected):** STA_A and STA_B send RTS frames, **and they both fail**
- They are allowed to send another DS frames
- **Both STAs wait for CTS/ACK Timeout before sending the next DS frame**



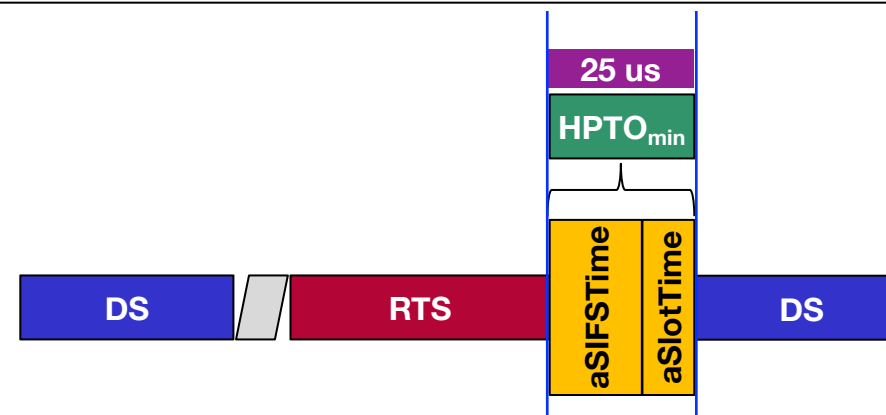
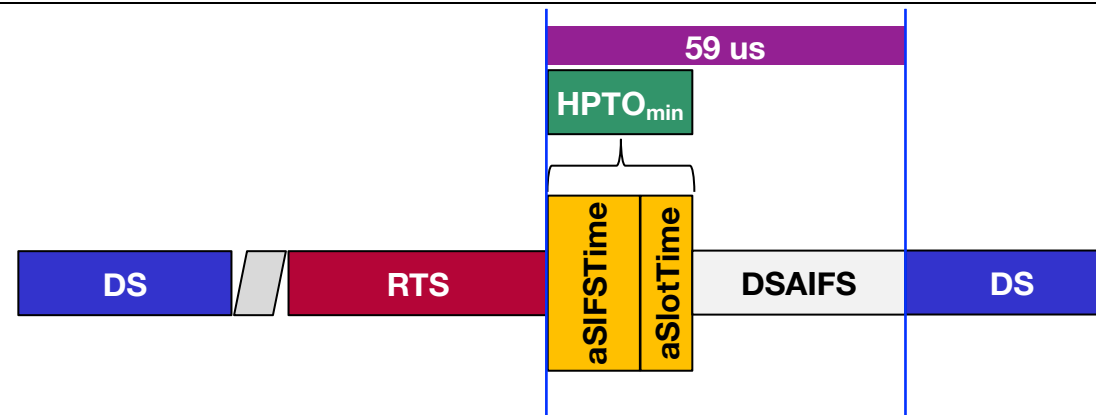
A Shorter Retry Timeout: High-Priority Timeout (HPTO)

- In this contribution, we allow STAs involved in P-EDCA to **use a High-Priority Timeout (HPTO) duration instead of CTS/ACK Timeout** to reduce the channel time used by P-EDCA
- **$HPTO_{min} = aSIFTtime + aSlotTime$**
 - $HPTO_{min}$ provides
 - Enough time for the **receiver** of the response-soliciting frame to receive, process, and start sending a reply to the sender, and
 - Enough time for the **sender** of the response-soliciting frame to perform carrier sensing (CCA) and switch to TX mode if the channel is sensed as idle
- **When a STA sends a response-soliciting frame (e.g., RTS), it can detect transmission failure if the channel is sensed as idle during the aSlotTime (after SIFS)**
- Note: A longer HPTO, such as $aSIFTtime + 2 \times aSlotTime$ may be used if sensing the channel for a longer duration is necessary

High-Priority Timeout (HPTO)

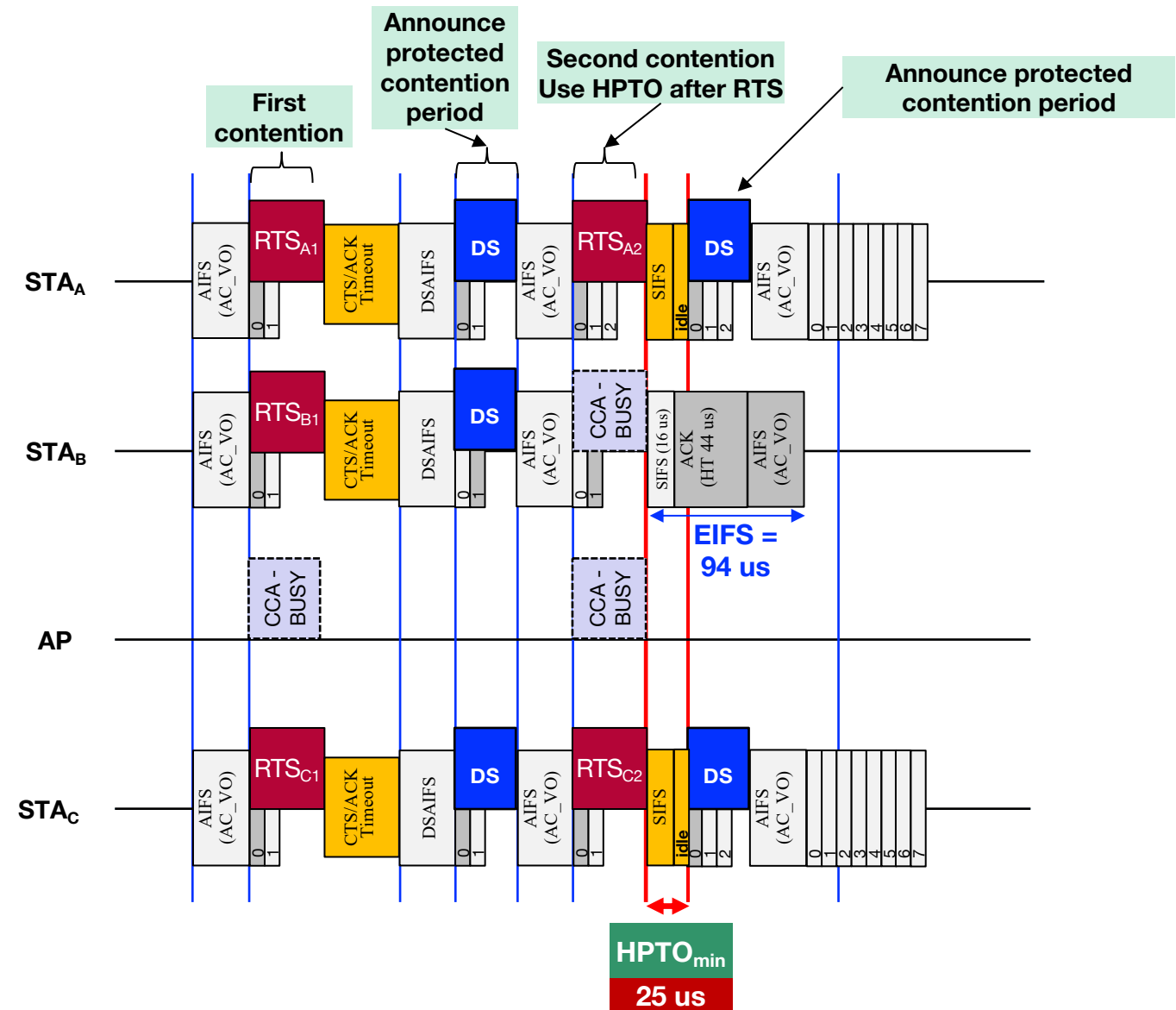


- $HPTO_{min}$ is the minimum amount of time a STA needs to wait before determining a transmission failure
- After sending an RTS, a STA may use the $HPTO_{min}$ duration to determine transmission failure
- If the channel is sensed as idle during the aTimeSlot duration, the transmission is considered as failed



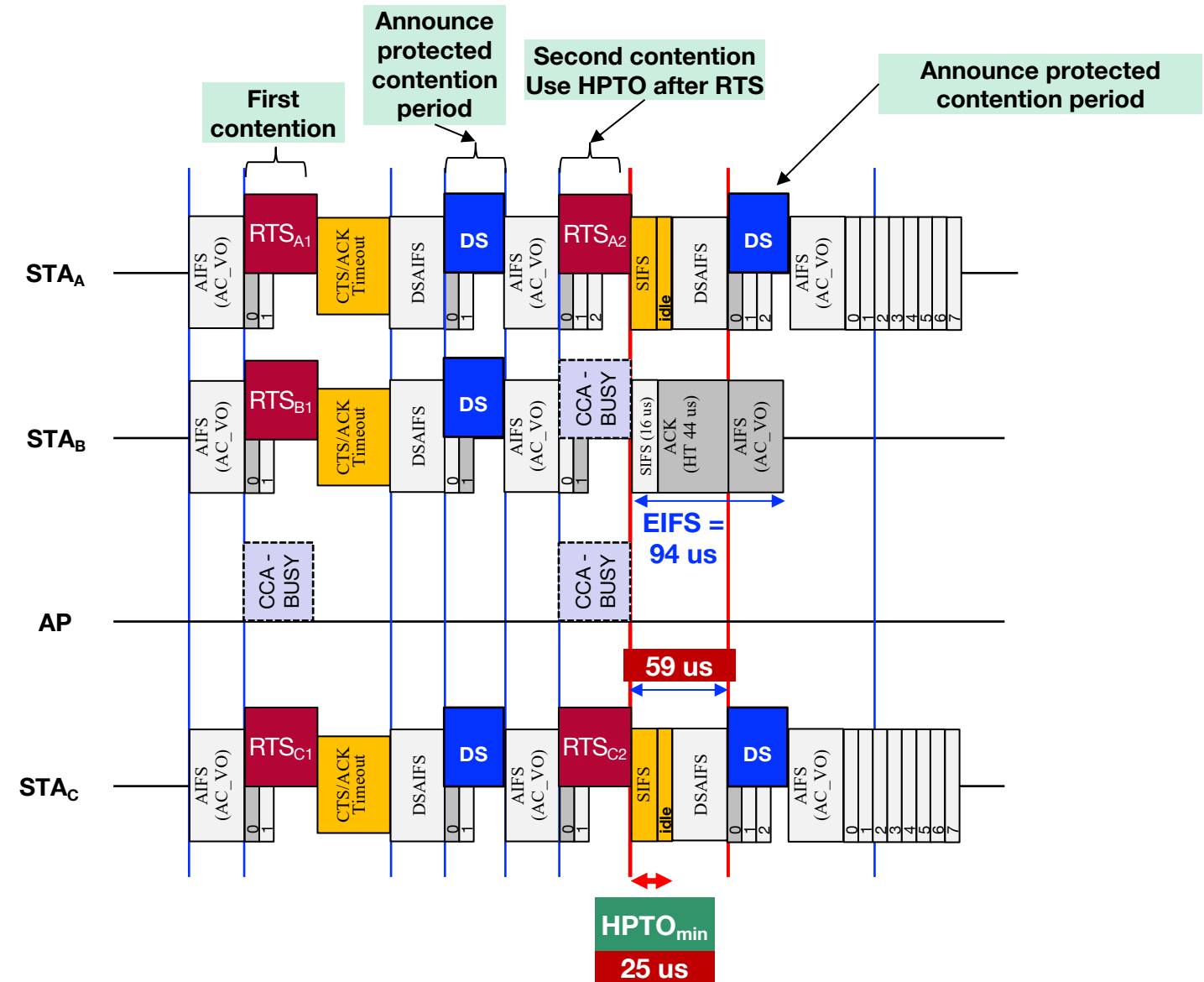
❑ Using HPTO for DS retransmissions

- $HPTO = HPTO_{min} + aSlotTime$
- **First contention round:** STA_A, STA_B, and STA_C compete; all three STAs use CTS/ACK Timeout after sending their frames
- STA_A, STA_B, and STA_C send DS frames to announce **protected contention period**
- **Second contention round (protected):** STA_A and STA_B send RTS frames, and they both fail
- They are allowed to send another DS frames
- **Instead of using CTS/ACK Timeout after their RTS frames, both STAs wait for HPTO before sending the next DS frame**



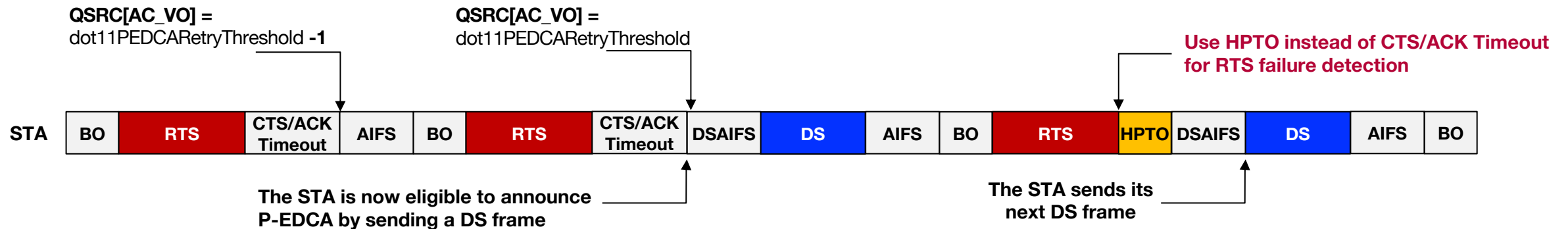
❑ Using HPTO for DS retransmissions

- In the figure, it is assumed that the earliest time a P-EDCA eligible STA can resend a DS frame is **DSAIFS** after the failure detection
- This results in a 59 us interval between the end of a RTS and the start of a DS
 - Compare with 79 us when using CTS/ACK Timeout (slide 4)



❑ Using HPTO for DS retransmissions

- Assume a STA starts P-EDCA by sending DS frame when $QSRC[AC_VO] = \text{dot11PEDCARetryThreshold}$
- During the P-EDCA's protected contention periods, after sending an RTS frame, the STA uses HPTO to detect RTS transmission failure
- If the channel is sensed as idle during the aSlotTime of HPTO, the STA starts the DSAIFS duration immediately



Performance Evaluation

VO STA:

AIFSN = 2

CWmin = 4, CWmax = 8 (AC_VO)

RTS/CTS: on

P-EDCA, HPTO variants:

P-EDCA: off, HPTO: off

P-EDCA: on (enforce RTS/CTS: on, enforce preamble detection: on/off), HPTO: on/off

Variants with max_hip_attempts and retx_threshold
max_ds_attempts = {3}
retx_threshold = {1}

TXOP limit: 2.08 ms (multiple pkts in the same TXOP can be transmitted with RIFS spacing)

BE STAs:

AIFSN = 3

CWmin = 16, CWmax = 1024

RTS/CTS: on

P-EDCA, HPTO: off (using AC_BE EDCA)

TXOP limit: 5 ms (multiple pkts in the same TXOP can be transmitted with RIFS spacing)

AP:

AIFSN = 1

CWmin = 16, CWmax = 64

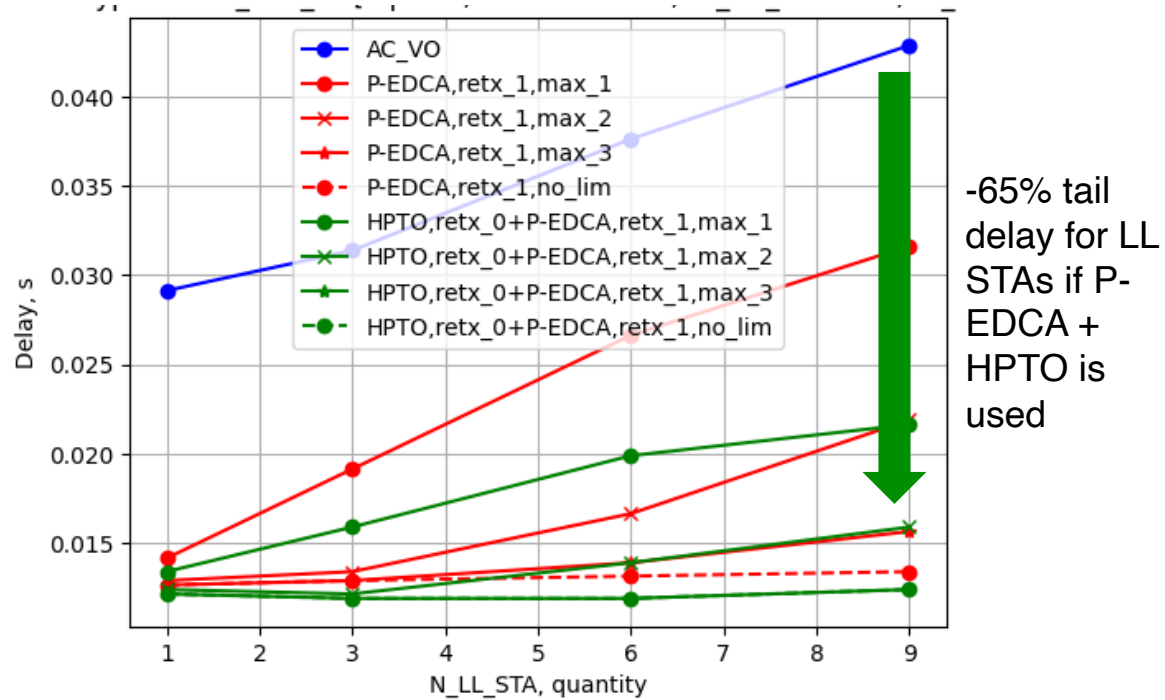
RTS/CTS: on

P-EDCA, HPTO: off

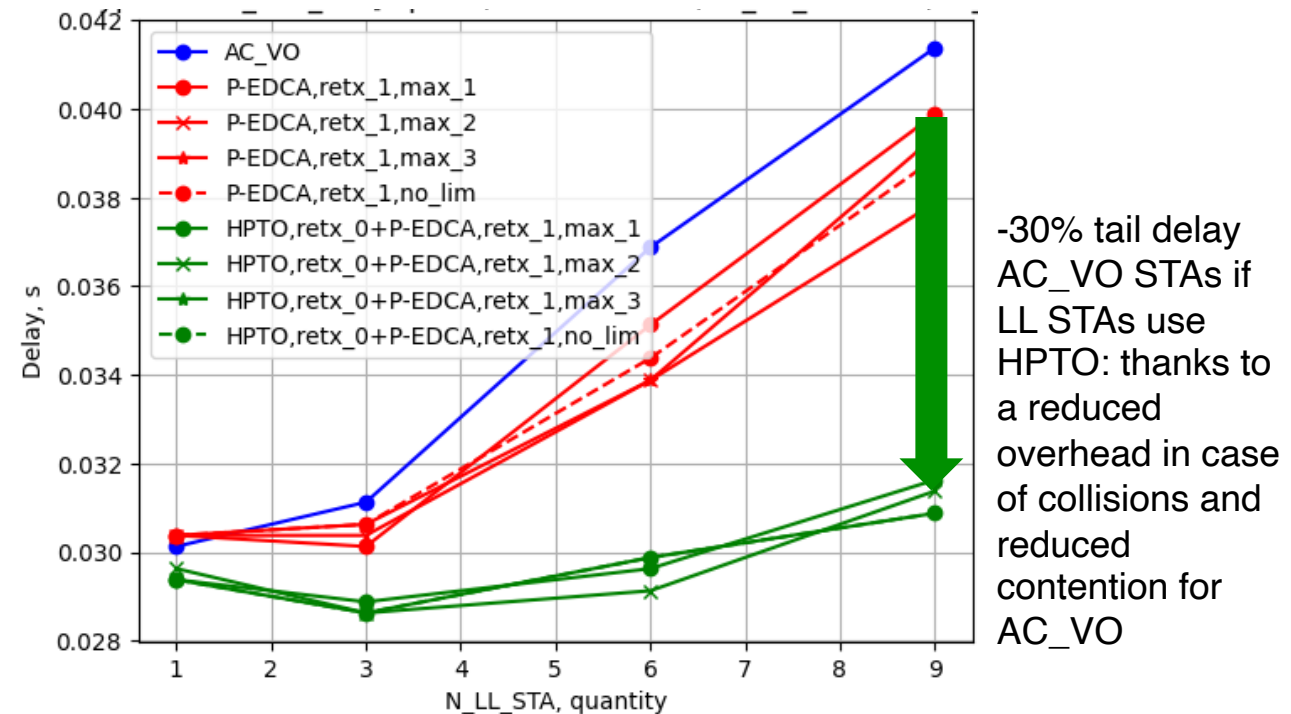
TXOP limit: 5 ms (multiple pkts in the same TXOP can be transmitted with RIFS spacing)

Performance benefits of HPTO. Results

LL STAs, 99th percentile



AC_VO STAs, 99th percentile



Summary

- Normally, when a STA sends a response-soliciting frame (e.g., RTS, data), it waits for an CTS/ACK Timeout duration to determine if the transmission has failed
- **In this contribution, we proposed that STAs involved in P-EDCA may bypass CTS/ACK Timeout and instead use a High-Priority Timeout (HPTO) duration**
 - After sending a response-soliciting frame, a STA waits for $HPTO_{min} = aSIFSTime + aSlotTime$ to determine transmission failure
- The combination of P-EDCA with HPTO allows improving latencies for both LL STAs, which use P-EDCA and HPTO, and the legacy AC_VO STAs, which benefit from a low-overhead collision resolution by the LL STAs.

Straw Poll

SP1. Do you agree that when "QSRC[AC_VO] \geq dot11PEDCARetryThreshold - 1" a P-EDCA capable STA may use the High-Priority Timeout (HPTO), defined as SIFS + aSlotTime, to detect the failure of an RTS frame, as long as PSRC[AC_VO] < dot11PEDCAConsecutiveAttempt?

YES/NO/ABSTAIN

SP2. Do you agree that when "QSRC[AC_VO] \geq dot11PEDCARetryThreshold - 1" a P-EDCA capable STA may use the High-Priority Timeout (HPTO), defined as SIFS + aSlotTime, to detect the failure of an RTS frame transmission, increment QSRC[AC_VO], and subsequently transmit the DS frame at:

- Option A: The end of HPTO,
- Option B: The end of HPTO + DSAIFS[AC_VO],

as long as PSRC[AC_VO] < dot11PEDCAConsecutiveAttempt

Note: $DSAIFS[AC_VO] = aSIFSTime + (AIFSN + DSr) \times aSlotTime$ [from the latest PDT]

YES/NO/ABSTAIN