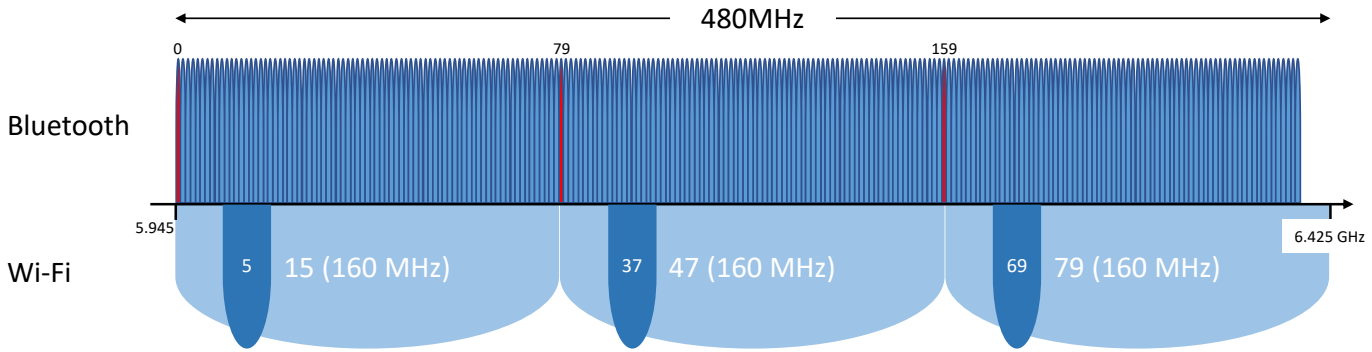
Balancing Wideband & Narrowband Frequency Hopping Channel Access Mechanisms for Operation in 6GHz

Date: 2024-03-13

Authors:

Name	Affiliations	Address	Phone	email
Sebastian Max	Ericsson	Herzogenrath, Germany		sebastian.max@ericsson.com
Leif Wilhelmsson	Ericsson			
Charlie Pettersson	Ericsson			

Assumptions 1: Spectrum

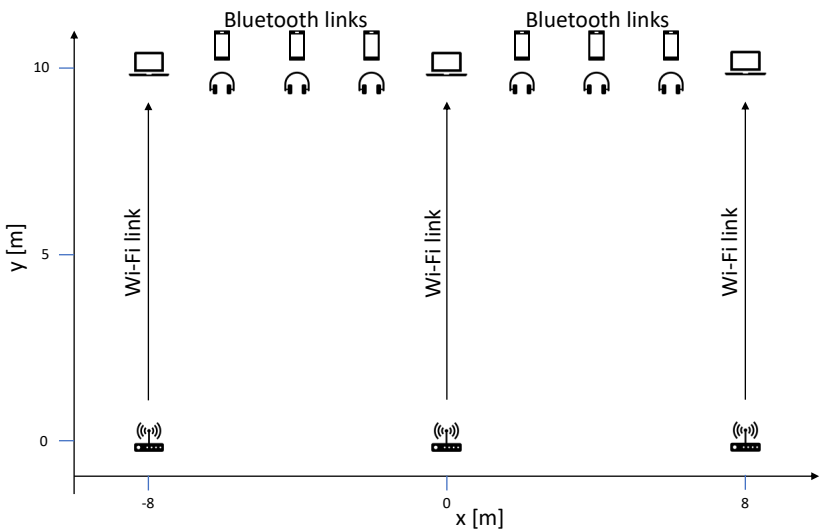


- Wideband system operating according to ETSI EN 303 687
  - Represented by a Wi-Fi system
  - 3 × 160MHz channels (channel number 15, 47, 79) with primary 20MHz channels
- Narrowband Frequency-Hopping system sharing the spectrum
  - Represented by a Bluetooth system
  - 1 MHz channels with 0.5MHz guard band left & right
  - 3 advertisement channels, 233 data channels

Assumptions 2: Scenario

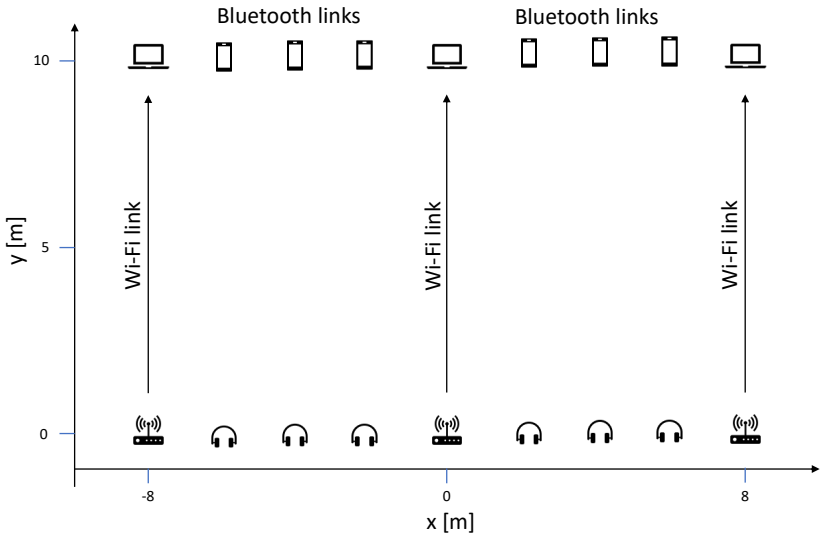
Scenario A

- Three separate Wi-Fi links on three 160MHz channels
  - 23dBm transmit power
  - Energy detection threshold -72dBm/20MHz
- (Up to) six Bluetooth links
  - 10dBm transmit power
  - Energy detection threshold -85dBm/1MHz



Scenario B

- Increase distance of Bluetooth to reduce received signal strength
  - More susceptible to interference
  - Modelling "body blockage" loss between central and peripheral device



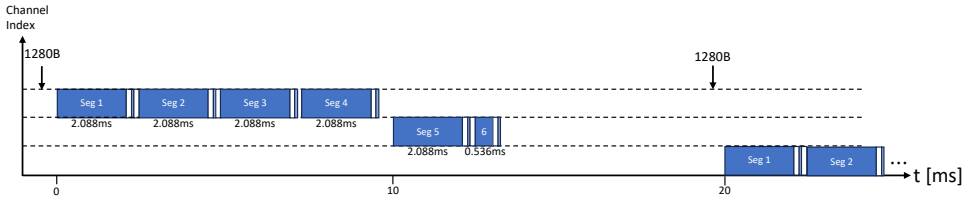
Assumption 3: Traffic & KPIs

Wi-Fi

- FTP 10MB file downloads
  - Back-to-back
  - Channel activity **~100%**
  - Access category BE
  - KPI: File download duration
- FTP 10MB file downloads with pauses
  - Pause duration between downloads [50...150]ms
  - Channel activity **~50%**
  - Access category BE
  - KPI: File download duration

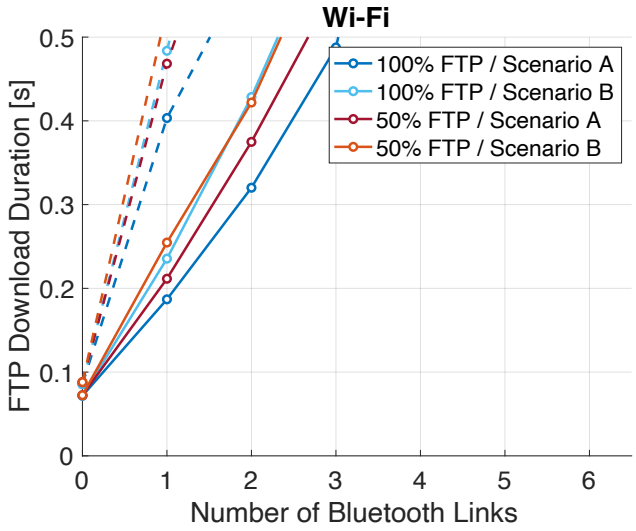
Bluetooth

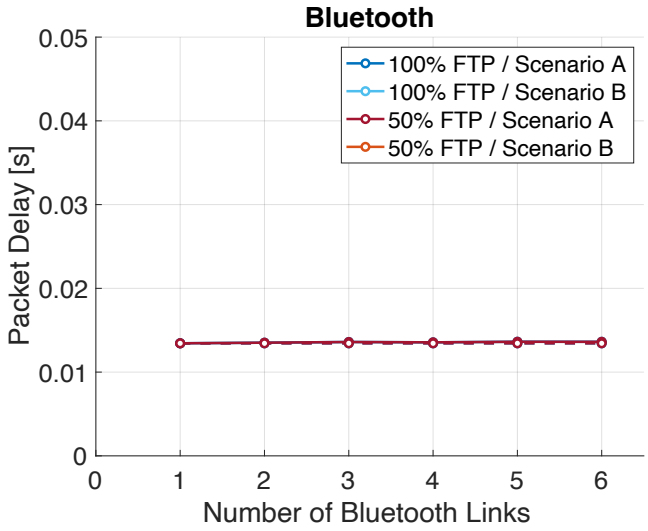
- Constant bitrate traffic (CBR)
  - 1280B every 20ms (≙512kb/s)
  - KPI: Packet delay
- Connection interval 10ms
  - Channel hopping frequency 10ms
- Total channel activity ~57%



Results

(Wi-Fi: FTP / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





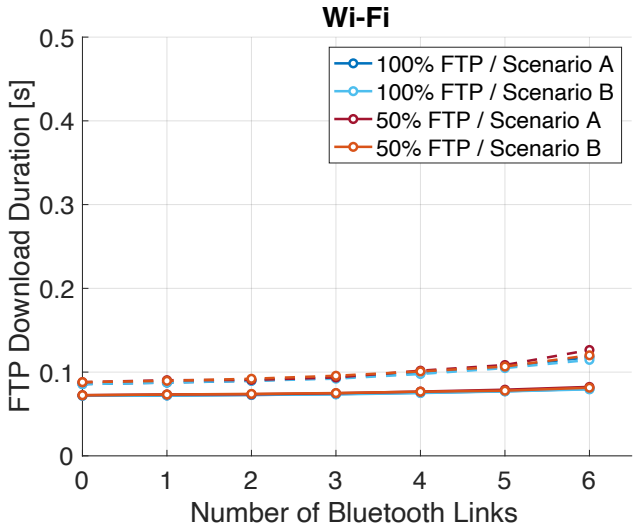
— Wi-Fi FTP download operation is completely disturbed by Bluetooth

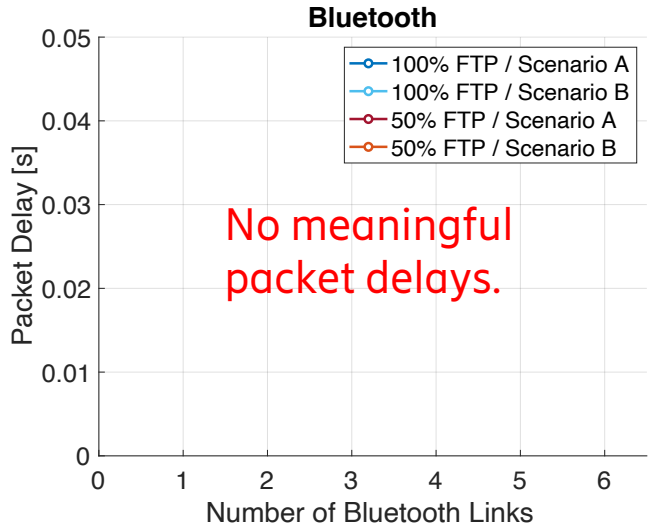
— Scenario A: Bluetooth operates undisturbed
— Scenario B: Bluetooth is completely disturbed

➔ Wi-Fi requires a Bluetooth coexistence mechanism
➔ Introduce LBT before every connection event

Results (introduce LBT for Bluetooth)

(Wi-Fi: FTP / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





- Wi-Fi FTP download durations are close to performance without interference

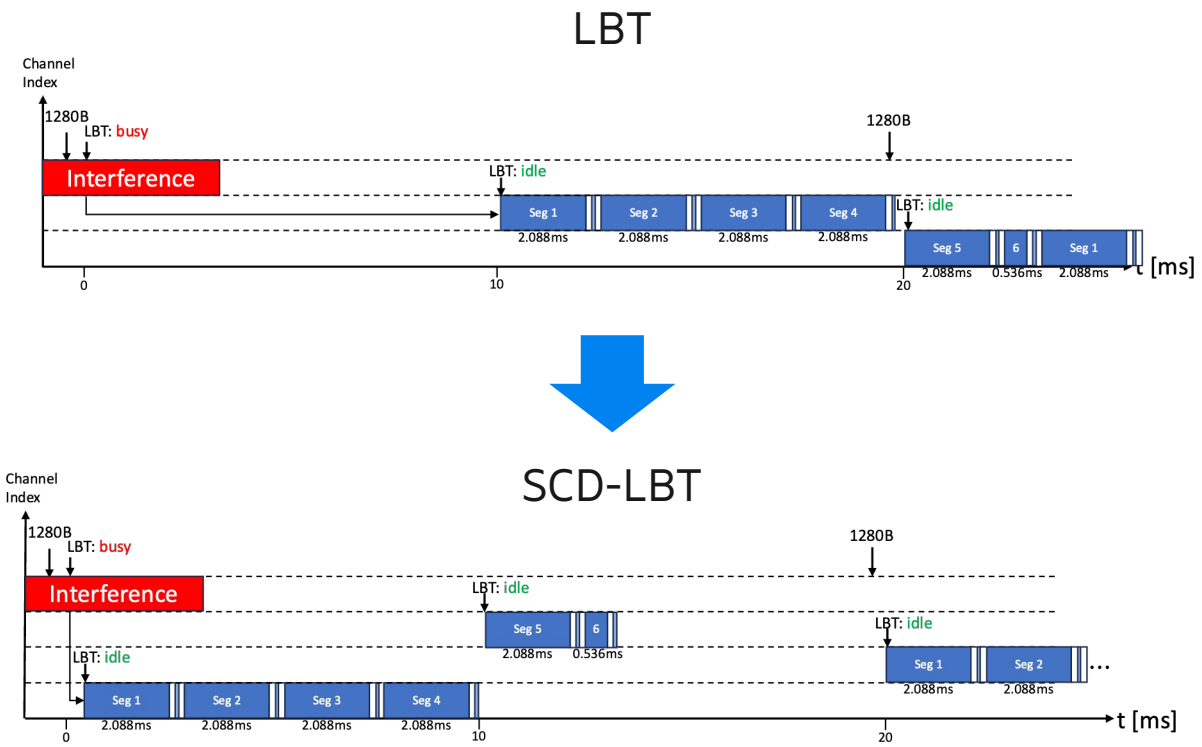
- Bluetooth does not get sufficient reliable access to the channel to deliver the data
- Grow rate of queues is larger than successful delivery rate
- Queues build up during the simulation & delay just keeps growing

➔ Bluetooth needs to be more persistent

➔ Introduce Secondary Channel Deferral (SCD) LBT before every connection event

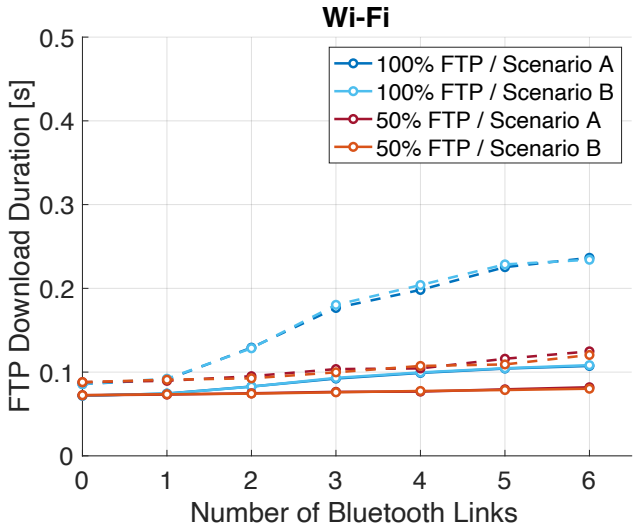
Secondary Channel Deferral (SCD) LBT

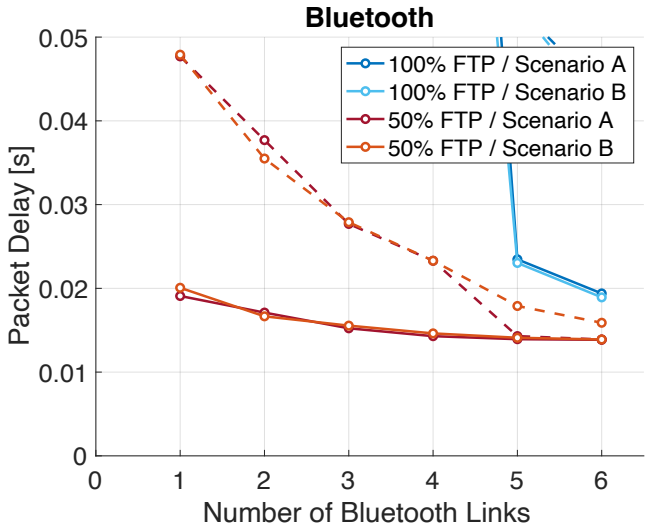
- Basic idea:
  - When initial LBT is busy, try to transmit on another channel as quickly as possible
- Secondary hop performed independent of regular hopping scheme
  - Receiver follows if no packet start is detected
- Regular hopping pattern resumed at next connection event
- Secondary hop frequency separation chosen to match Wi-Fi channel bandwidth (79 channels, ~160MHz)



Results (Enhance LBT by Secondary Channel Deferral)

(Wi-Fi: FTP / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





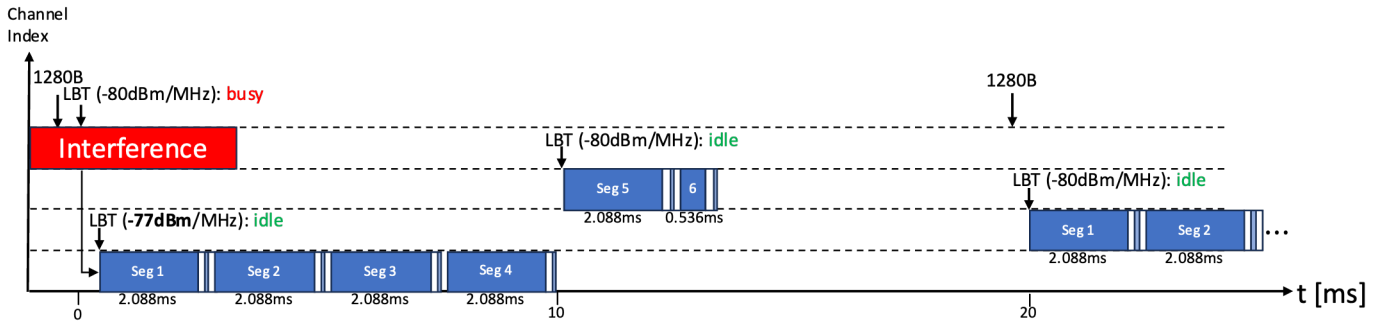
- 50%-load: Close to undisturbed
- 100%-load: Slightly disturbed

- Bluetooth operation is significantly improved for the Wi-Fi 50%-load
  - Still significant chance for high delay
- Wi-Fi's 100%-load still blocks Bluetooth
- Bluetooth links support each other

➔ Promising approach

➔ Mitigate impact to delay of Bluetooth in the presence of high Wi-Fi load

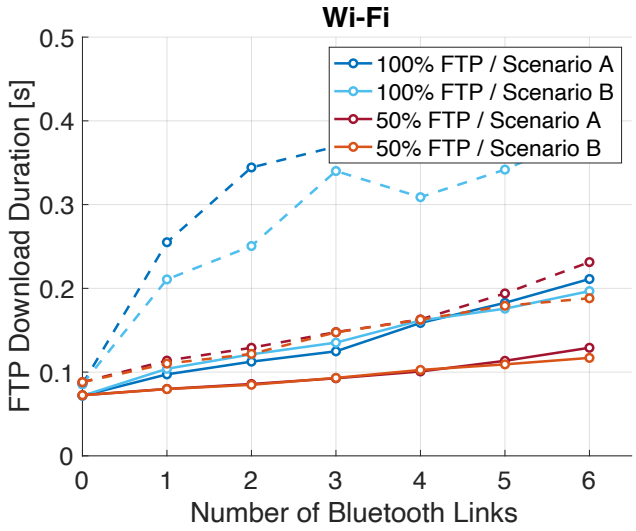
High-Load Mitigation for Bluetooth: SCD-LBT with EDT Ramp-Up

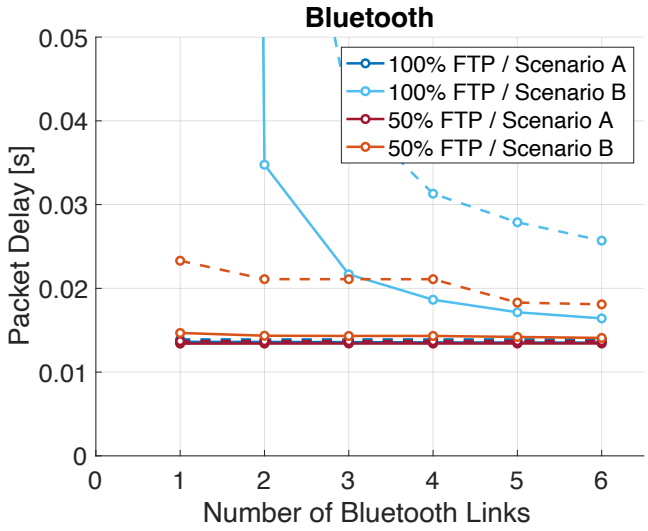


- Initial LBT done with -80dBm/MHz Energy Detection Threshold (EDT)
  - Idle: Transmit
  - Busy: Start Secondary Channel Deferral procedure
    - Deferral channel is ~160MHz separated from original channel
    - **Increase EDT by 3dB**
- Allow up to 5 deferrals, ending at EDT = -65dBm/MHz
  - Bluetooth slowly raises EDT, becomes less sensitive & increases chance of observing an idle channel

Results (Enhance Bluetooth SCD-LBT with EDT Ramp-Up)

(Wi-Fi: FTP / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





— Bluetooth's power backoff increases delays in 100%-load case, but only slightly in 50%-load case

— Significant decrease of 95-percentile unless Bluetooth is heavily interfered

✓ Mitigate impact to delay of Bluetooth in the presence of high Wi-Fi load: Stepwise increase of Energy Detection Threshold in case of blocked channel

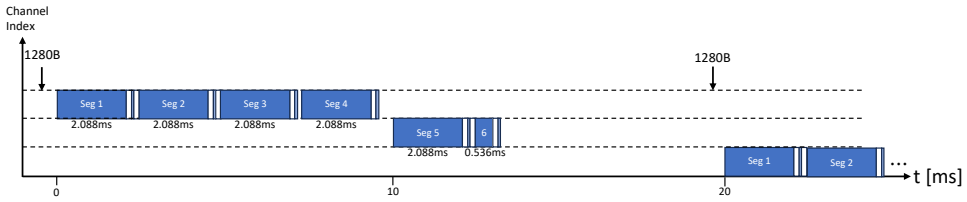
Assumption 3b: Traffic & KPIs

Wi-Fi

- Cloud Gaming
  - 30Mb/s downlink video & uplink control
  - Channel activity **~15%**
  - Access category VI & VO
  - KPI: gaming delay

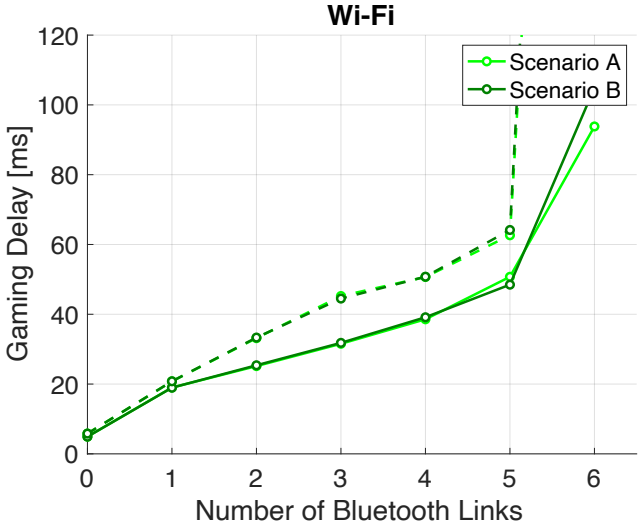
Bluetooth

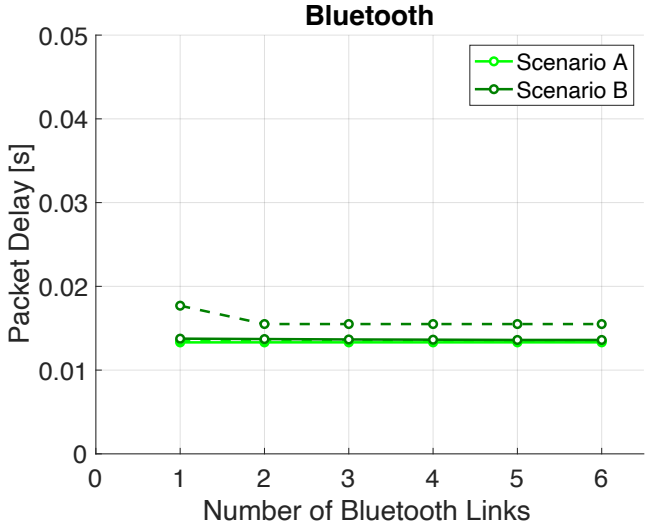
- Constant bitrate traffic (CBR)
  - 1280B every 20ms (≙512kb/s)
  - KPI: Packet delay
- Connection interval 10ms
  - Channel hopping frequency 10ms
- Total channel activity ~57%



Results (Bluetooth with LBT + SCD + ED Ramp-Up)

(Wi-Fi: Cloud Gaming / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





— Cloud gaming traffic delay: Significant increase by Bluetooth interference

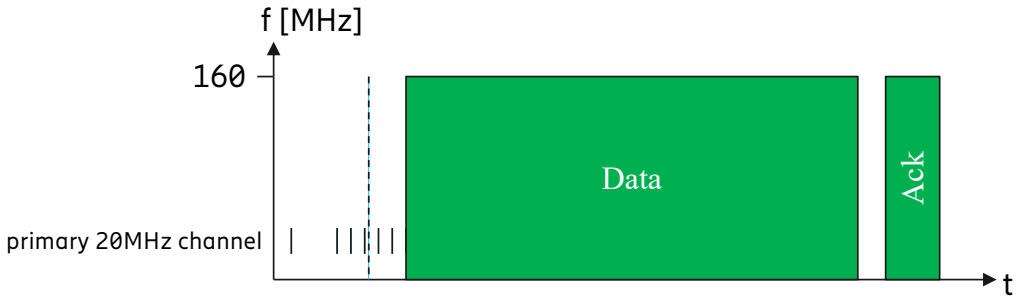
— Bluetooth delay close to performance without interference due to low Wi-Fi channel occupancy

— SCD / ED Ramp-Up barely active

➔ Decrease Wi-Fi delay in the presence of narrowband interference

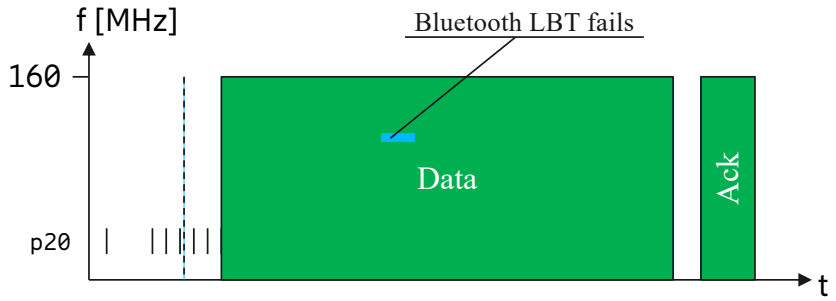
Wi-Fi Channel Access Procedure & Delay

- Wi-Fi uses “Option 2” according to EN 303 687 for channel bonding:
  - Initially, channel access only listens on the “primary” 20MHz channel
  - Count down backoff while the channel remains idle
  - 25µs before the transmission energy detection is performed on the full channel
  - Data transmission starts if primary 20MHz channel & full bonded channel remains idle



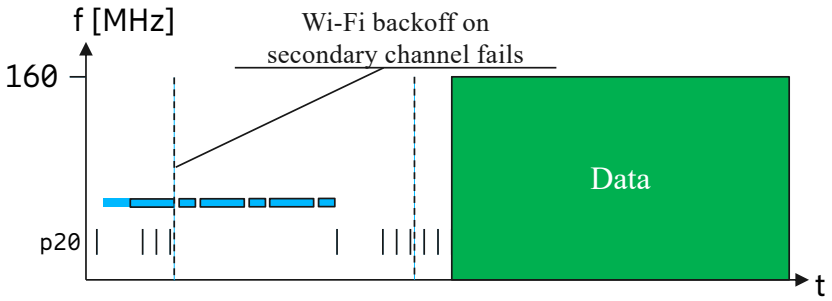
Order of Events on the Channel

Wi-Fi before Bluetooth



- Clear Channel Assessment (CCA) result is idle on all 20 MHz subchannels
- Wi-Fi transmission starts
- Bluetooth hops on channel, LBT detects transmission and defers
  - Mitigation by secondary channel deferral
- No delay increase

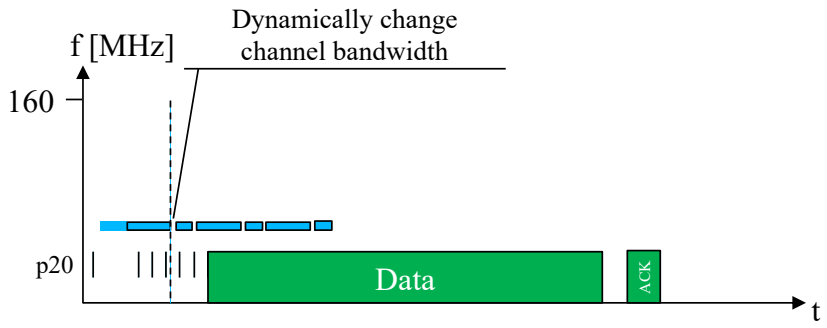
Bluetooth before Wi-Fi



- Bluetooth hops on channel, LBT detects an idle channel
- Bluetooth transmission starts
- Wi-Fi CCA detects Bluetooth transmission and defers
  - Bluetooth is on primary channel: anytime during backoff
  - Bluetooth is on secondary channel: during the last 25 µs of the backoff
- Wi-Fi continues to observe until the channel is idle and re-starts backoff
- Increase of channel access delay

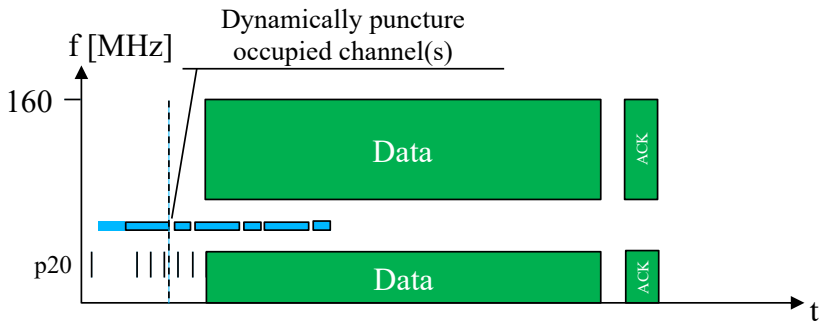
Wi-Fi Bandwidth Adaptation

Dynamic Backoff Bandwidth



- Dynamic Backoff allows adapting the channel bandwidth during countdown
- Primary 20MHz must be idle
- Only available bandwidths are 80MHz, 40MHz, and 20MHz

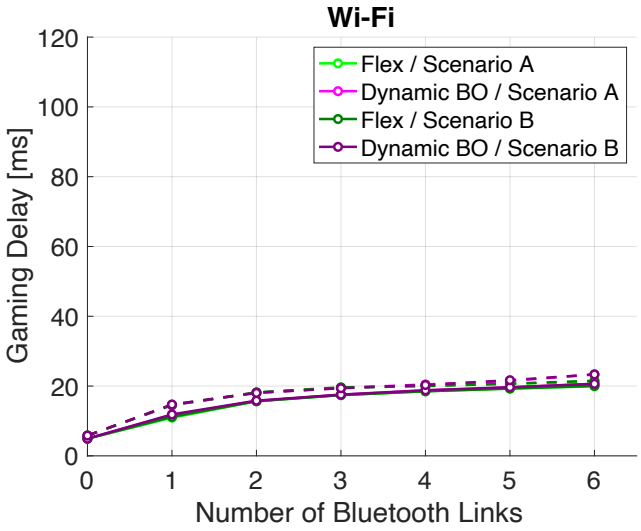
Fully Dynamic Puncturing

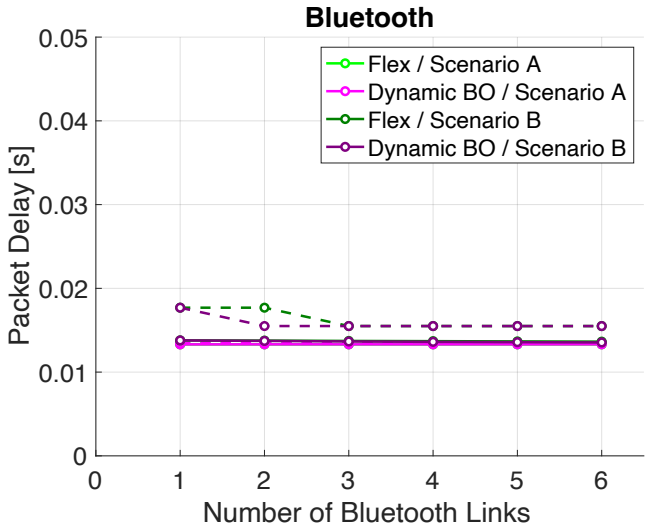


- Puncturing allows 20 MHz-sized "holes" in the spectrum
- Primary 20 MHz must be idle
- Fully dynamic in
  - Time: Puncturing is instantaneously, backoff simply continues
  - Frequency: Puncture arbitrary 20 MHz-channel holes
- This corresponds to "Option 1" in EN 303 687

Results (Enhance Wi-Fi by Bandwidth Adaptation)

(Wi-Fi: Cloud Gaming / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





- Flexible Puncturing gives the best delay
- Significant simpler dynamic backoff bandwidth is not much worse

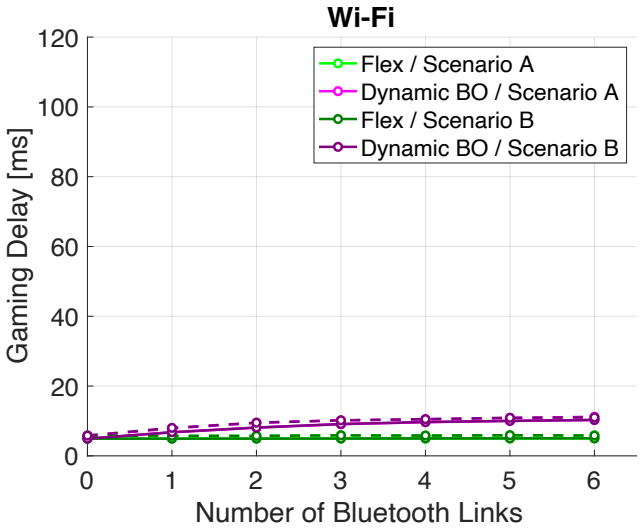
- No significant change for Bluetooth links

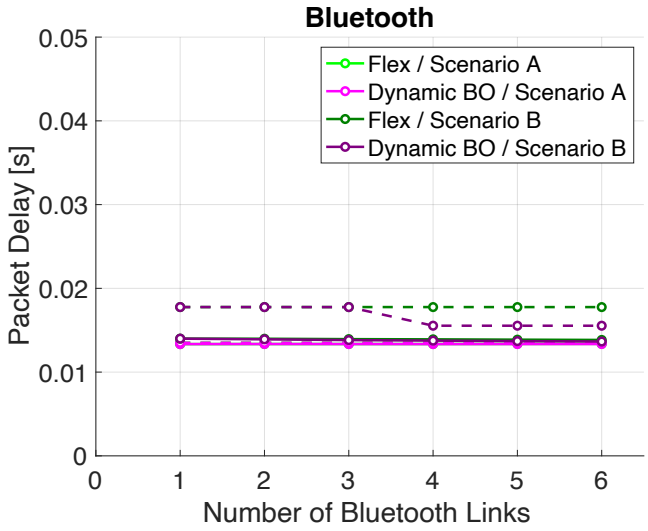
➔ Wi-Fi is able to adapt to narrowband interference

➔ Can we do better?

Results (Bluetooth avoids Wi-Fi's primary-20 channels)

(Wi-Fi: Cloud Gaming / Bluetooth: CBR; solid = mean / dashed = 95-percentile)





- Flexible Puncturing reaches optimum gaming delay
- Always idle primary 20MHz channel combined with Dynamic Backoff still provides significant delay improvements

- No significant change for Bluetooth links

✓ Decrease Wi-Fi delay in the presence of narrowband interference: Dynamic channel bandwidth + guaranteed idle primary-20 channels

Conclusions

- Dynamic sharing of 480MHz frequency spectrum between wideband and narrowband frequency-hopping systems is feasible
- Coexistence & performance of both systems can be achieved by
  - (Short) LBT in narrowband

  plus
  - Mitigation of channel access delay in narrowband by
    - Quickly trying a different channel and
    - Ramp-up of the energy detection threshold in case of retries

  plus
  - Mitigation of channel access delay in wideband by
    - Dynamic bandwidth adaptation or
    - Puncturing

  (plus
  - Leaving the primary 20 MHz idle for wideband.)

- Further Thoughts:
  - Narrowband may detect spectrum occupied and avoid channels
    - Separation in frequency should give best results if feasible
    - Learning might be supported by external information, agreements, LBT results
    - Has its own challenges, known from Adaptive Frequency Hopping in 2.4GHz
  - Define a "simpler" mode of operation, governed by restrictions
    - Inspired by Short Control Signalling, but adapted for narrowband frequency hopping

Annex: Cloud Gaming Traffic Model & KPI

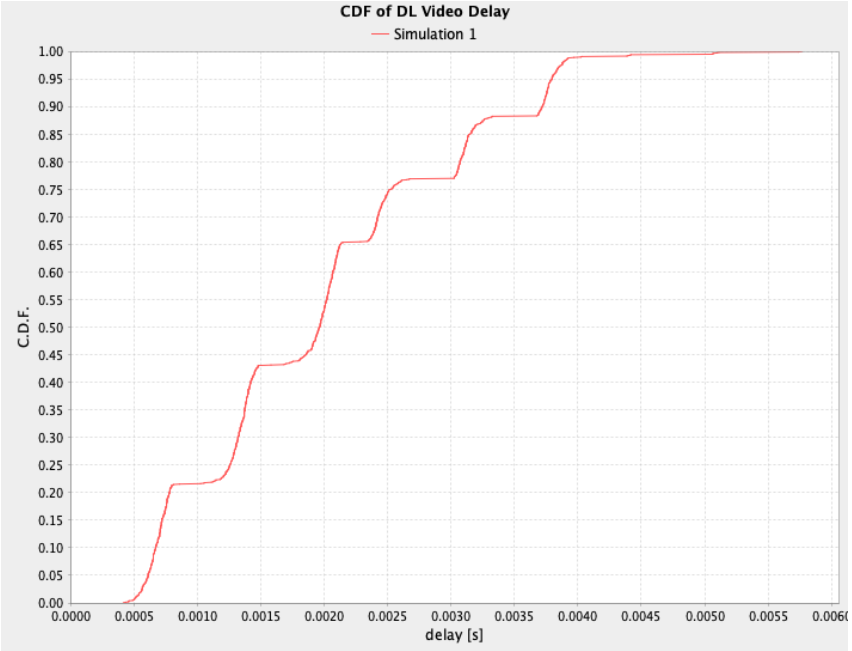
Cloud Gaming Traffic Model & KPI

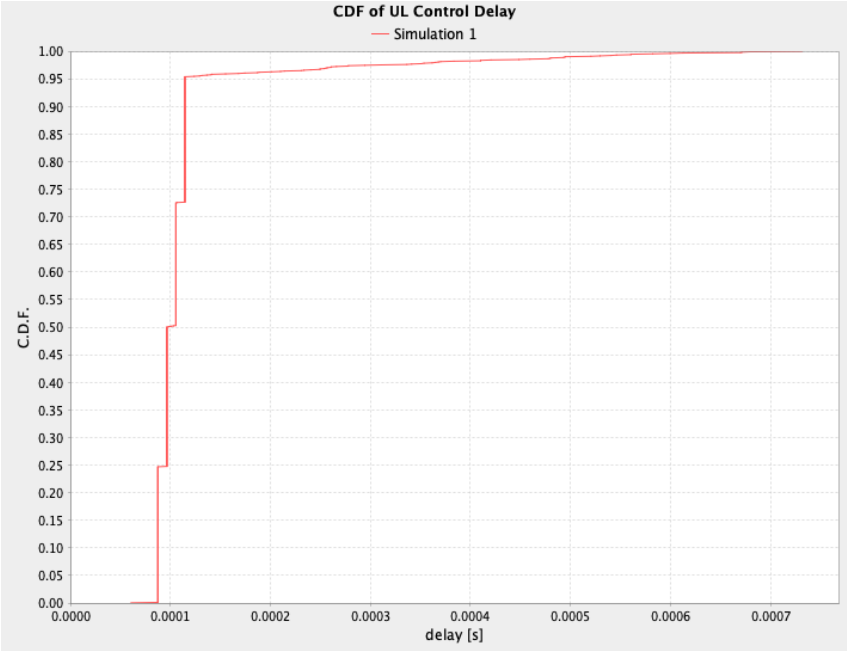
- Downlink: XR Video Model over RTP
  - Mean load 30 Mb/s with 10 Mb/s standard deviation
  - 60 frames/s, every 120th frame is an I-frame (2×size)
  - Access category video
  - Measure frame delivery delay
- Uplink: Control Messages over UDP
  - Mean size 468 B with 90 B standard deviation
  - Mean inter-frame interval 15 ms with 6 ms standard deviation
  - Access category voice
  - Measure frame delivery delay

KPI:

(Mean & 95-percentile) Gaming Delay

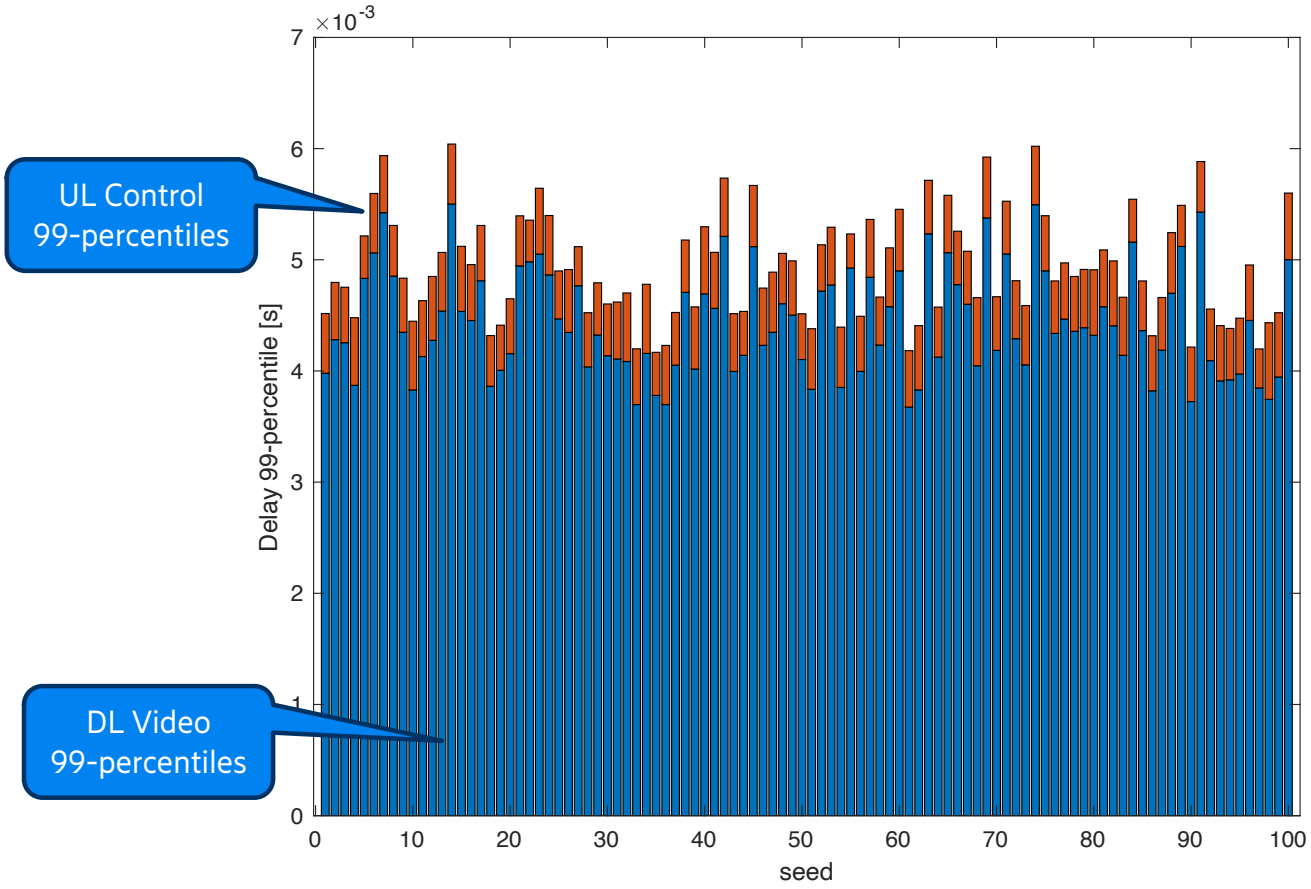
Single Simulation





— Single simulation results in CDFs for the downlink video packet delay & uplink control packet delay

99-percentiles of 100 simulations



- Run 100 simulations
  - 100 CDFs
  - 100 UL Control 99-percentiles
  - 100 DL Video 99-percentiles
- Define "gaming delay": sum of UL & DL 99-percentiles
  - 100 gaming delays
- KPI: Mean / 95-percentile of gaming delay