DeepBeam: Deep Waveform Learning for Coordination-Free Beam Management in mmWave Networks

Date: 2021-05-10

Authors:

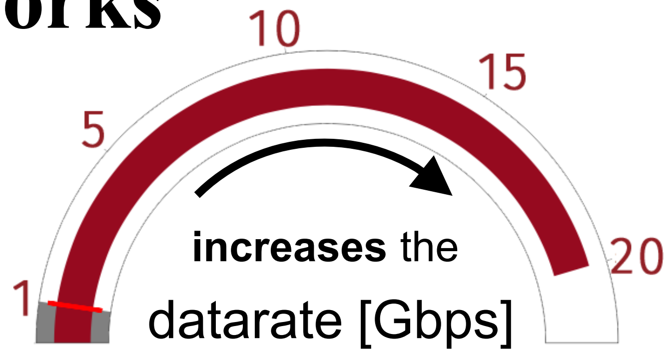
Name	Affiliations	Address	Phone	email
Francesco Restuccia	Northeastern University, WIOT Institute	360 Huntington Ave, Boston, MA	617-373-3655	frestuc@northeastern.edu
Michele Polese				m.polese@northeastern.edu
Tommaso Melodia				melodia@northeastern.edu

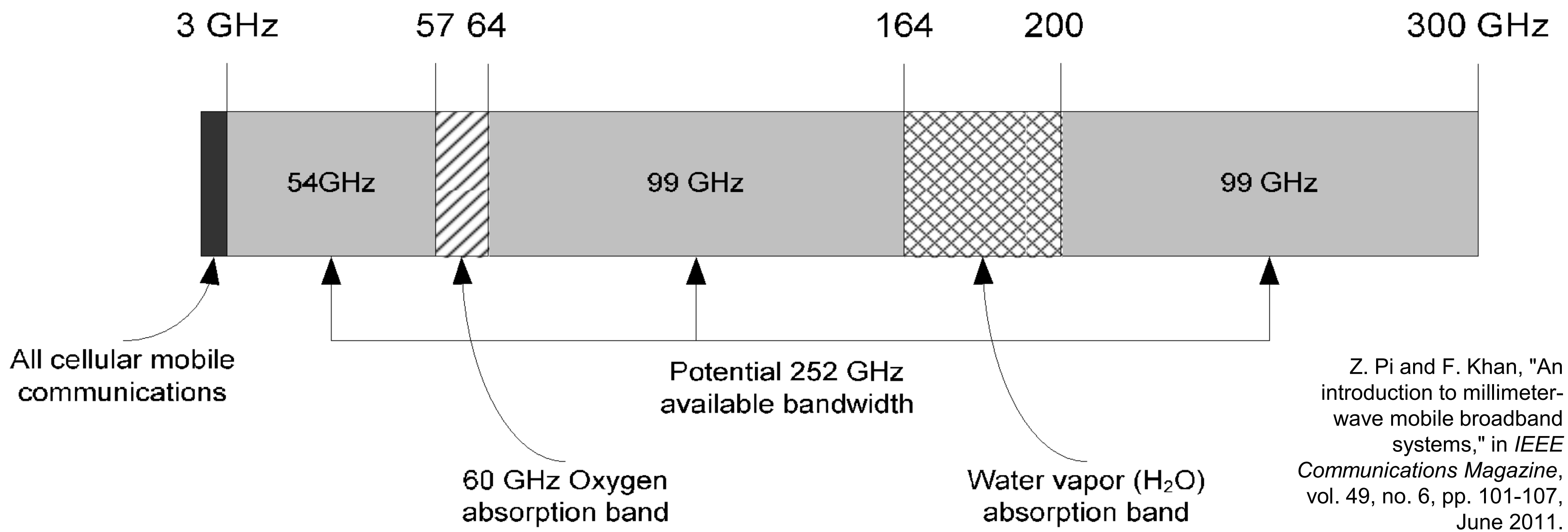
Outline

- Motivation
- Contributions
- DeepBeam framework
- DeepBeam use cases
- Experimental evaluation
- Conclusions

mmWaves in mobile networks

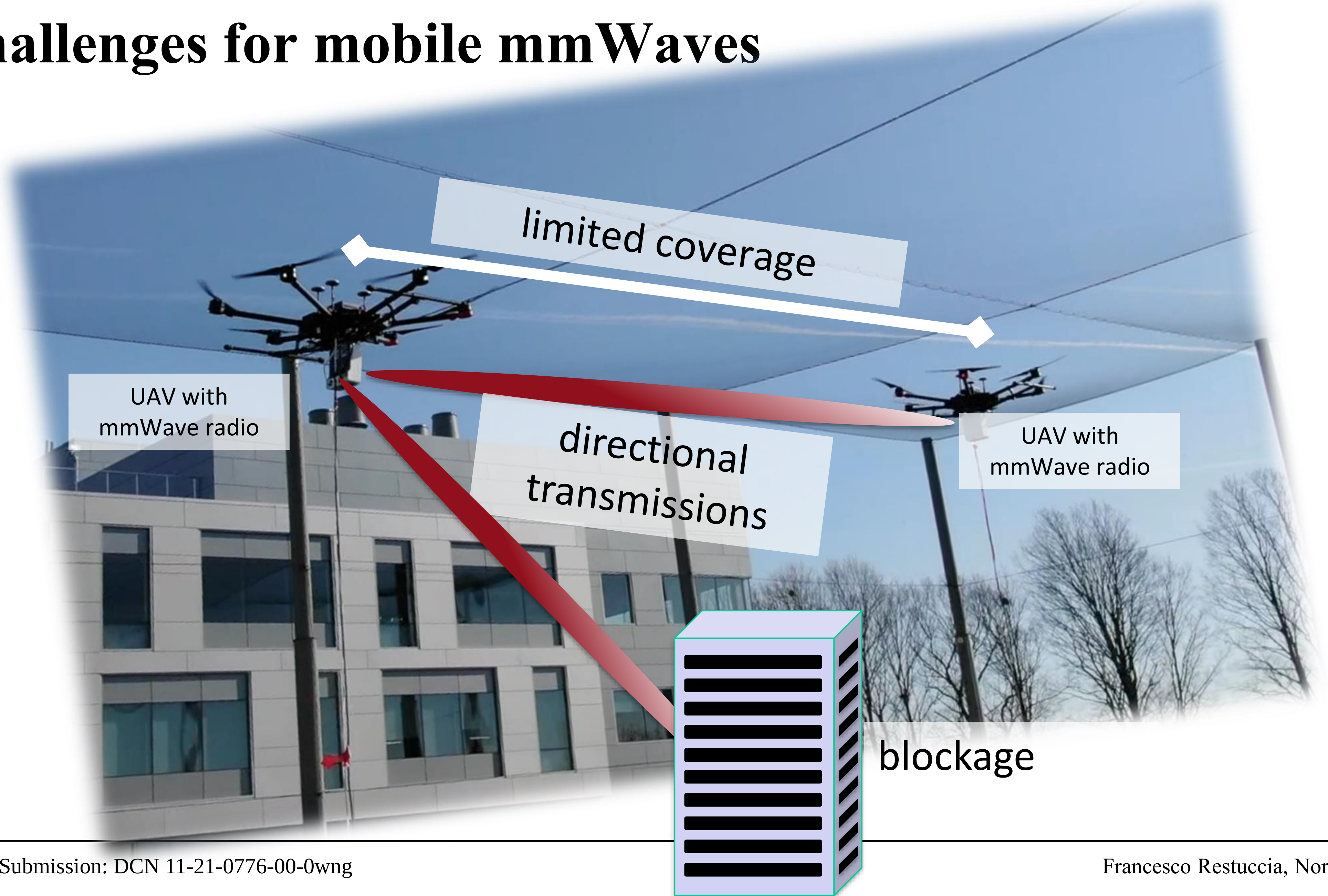
IEEE 802.11ad
supports **frequencies up to**
70.2 GHz with 2.16GHz channels



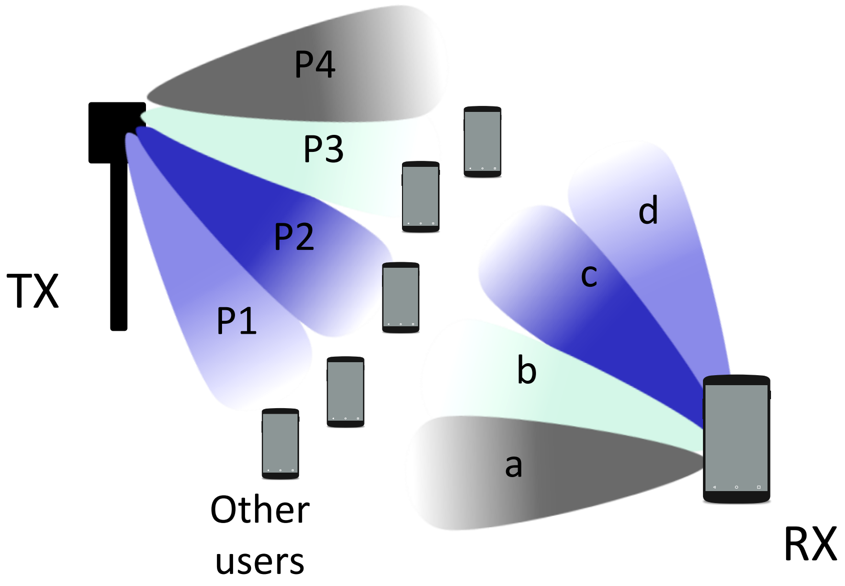


Z. Pi and F. Khan, "An introduction to millimeter-wave mobile broadband systems," in *IEEE Communications Magazine*, vol. 49, no. 6, pp. 101-107, June 2011.

Challenges for mobile mmWaves



Problem: Beam management in mmWave networks



TX and RX focus their energy in narrow beams

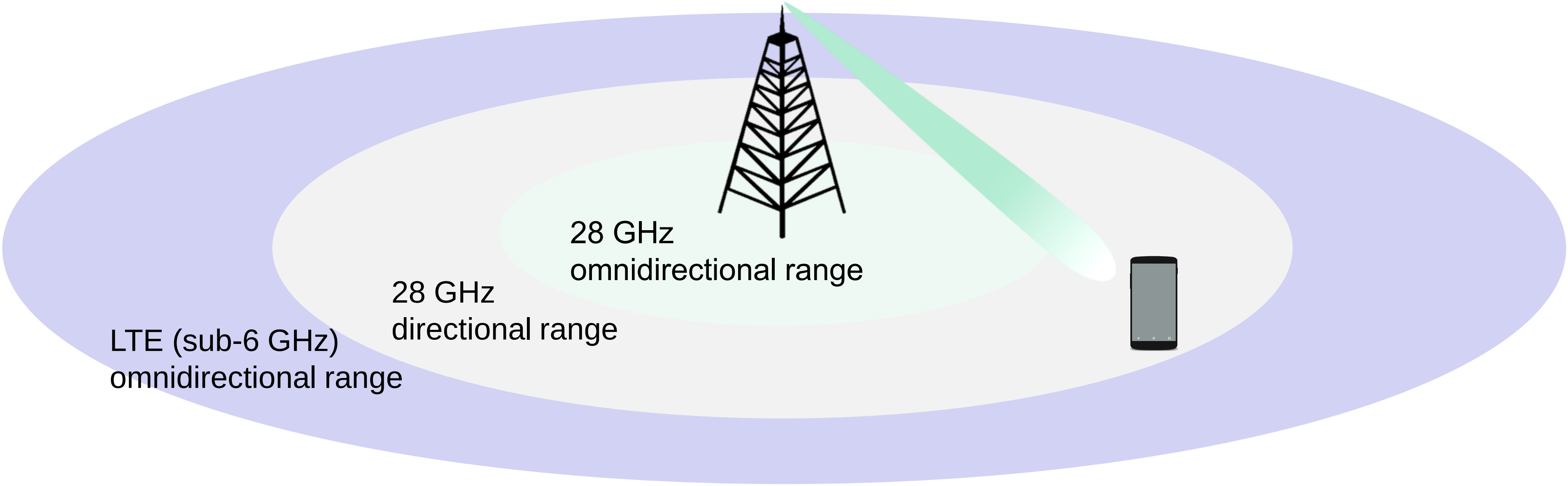
- They need to point the beams toward each other

- Otherwise, the **gain** introduced by using beamforming could disappear

Directionality Challenges

need beamforming gain even during the cell search of initial access

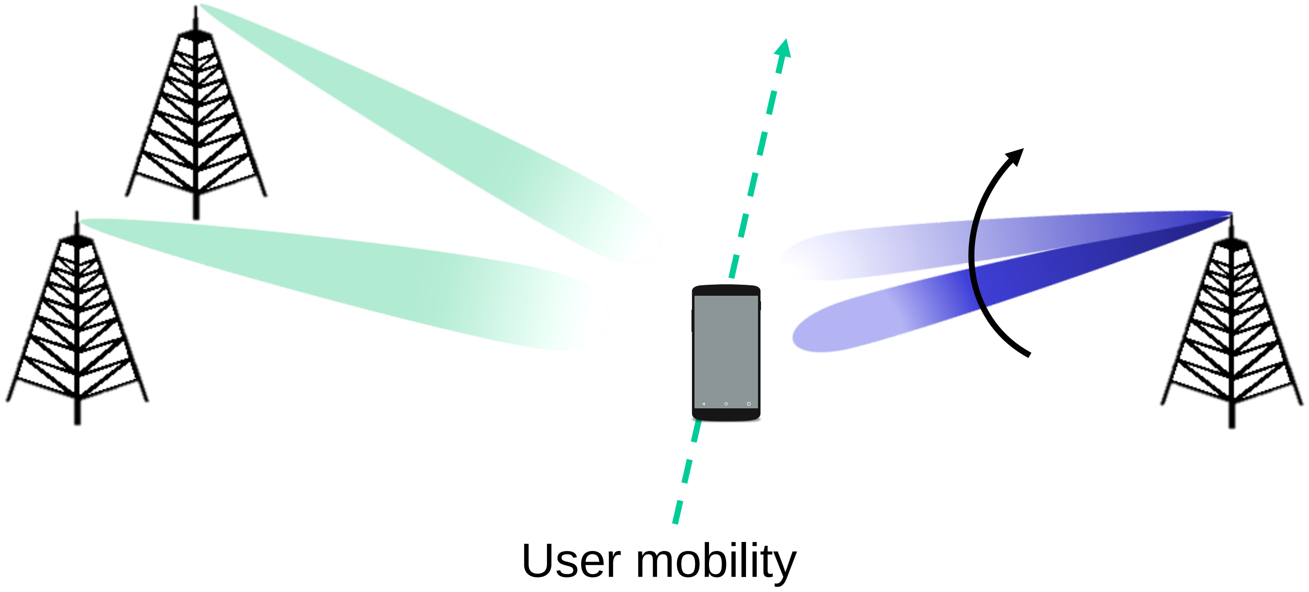
Discovery range mismatch



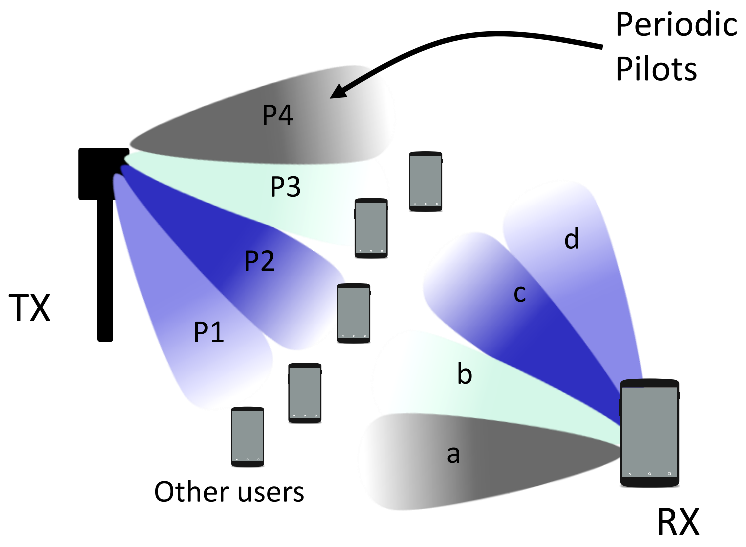
Directionality Challenges

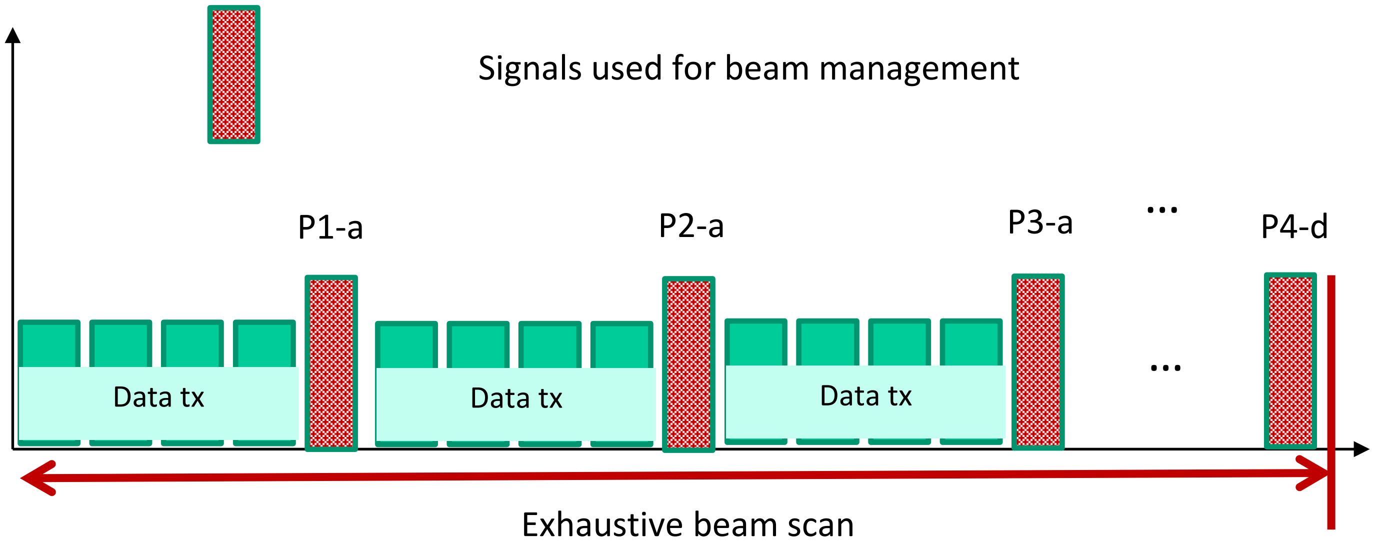
Need for tracking

need to track beams (and, in case, update access point/BS) as the user moves



Traditional beam management





Traditional operations:
- Need for pilots
- Need for exhaustive scan

High latency and overhead

In IEEE 802.11ad, beams are distributed in 128 sphere sectors, with beam widths as small as 3 degrees (**Nitsche et al, Steering With Eyes Closed: Mm-wave Beam Steering Without In-band Measurement, INFOCOM 2015**).
A beam sweep is performed by the TXer plus intra-sector fine-tuning is used to refine the selection (**Nitsche et al., IEEE 802.11 ad: Directional 60 GHz Communication for Multi-Gigabit-per-second Wi-Fi, IEEE Comm. Mag, 2014**.)

Typical 3GPP NR configuration can take up to **164 ms** for 24-beam codebooks at TX and RX

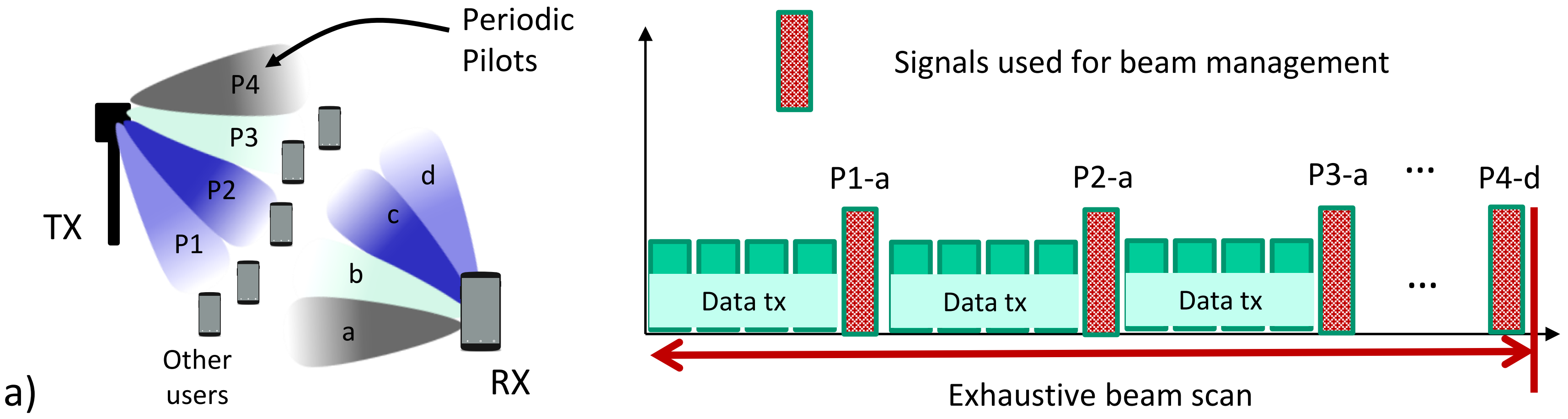
Deep Learning for mmWaves

Directionality | High data rates | Blockage

- Complex control procedures (e.g., beam management)
- Need for coordination among network nodes
- Need for quick reactions

AI can play a crucial role to optimize mmWave operations, with predictive and/or autonomous control policies

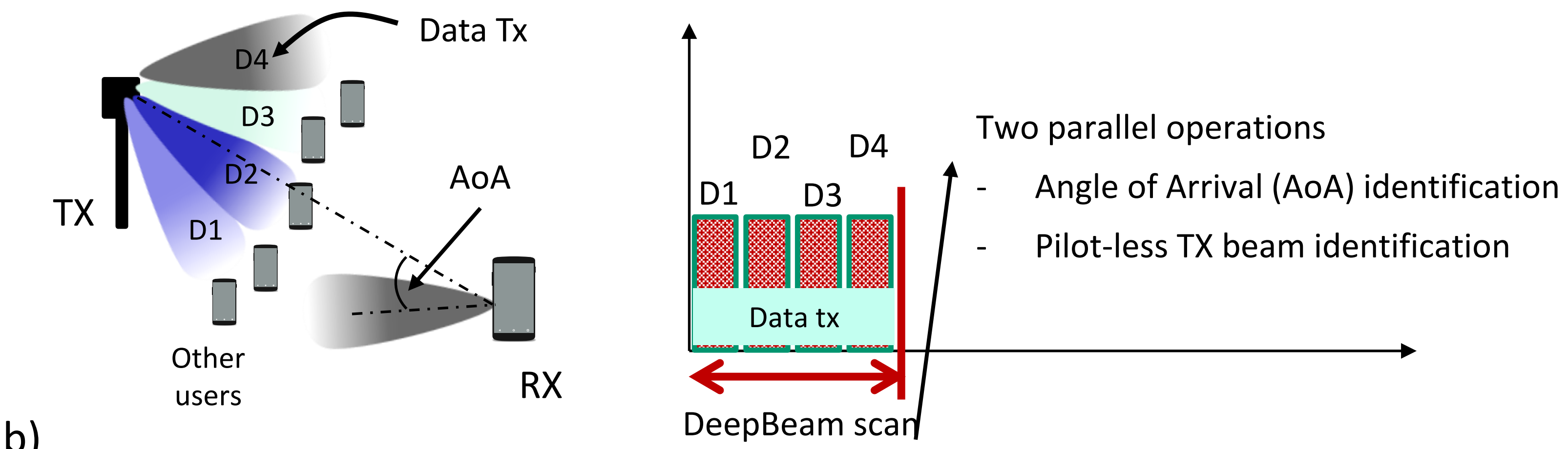
AI-enabled Beam Management



Traditional operations:

- Need for pilots
- Need for exhaustive scan

High latency and overhead



Deep-learning-enabled operations:

- Exploit ongoing data transmissions (no pilots)
- No need for exhaustive scan at RX

Reduce latency and overhead

Contributions

1. **First waveform-learning framework for mmWaves**
   - Speed up initial access and tracking
   - No need for pilots
2. **Experimental validation**
   - Dataset with 4TB of raw waveform, to be released
   - Multiple radios (NI/SiBeam and Pi-Radio)
   - Multiple TX/RX combinations and spatial configurations

M. Polese, F. Restuccia and T. Melodia, "DeepBeam: Deep Waveform Learning for Coordination-Free Beam Management in mmWave Networks," Proc. of ACM MobiHoc 2021. Preprint available at https://arxiv.org/abs/2012.14350.

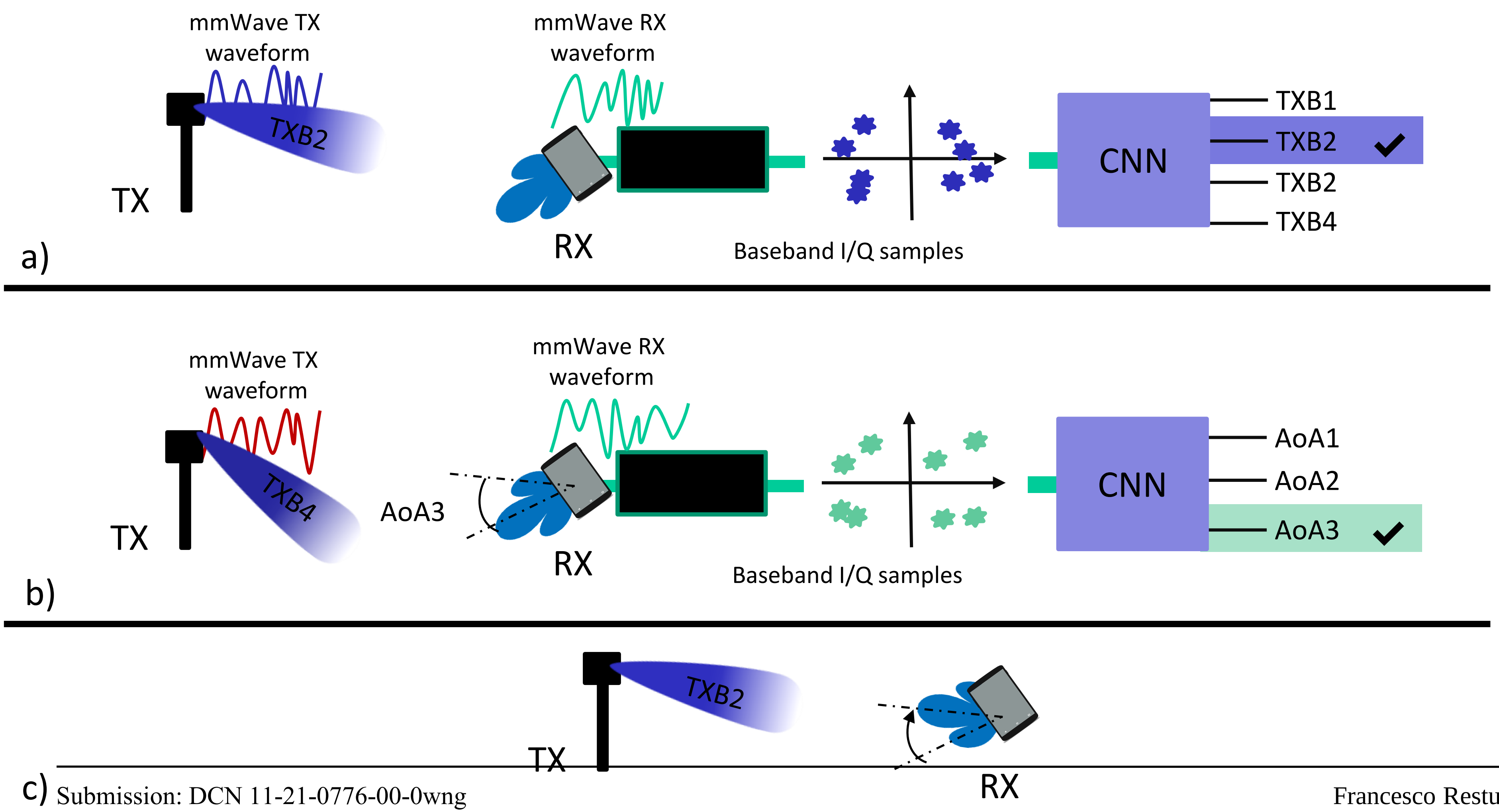
Contributions

Our approach achieves accuracy of up to 96%, 84% and 77% with a 5-beam, 12-beam and 24-beam codebook

Our approach reduces latency by up to 7x with respect to the 5G NR initial beam sweep

DeepBeam in a nutshell

What does DeepBeam learn?

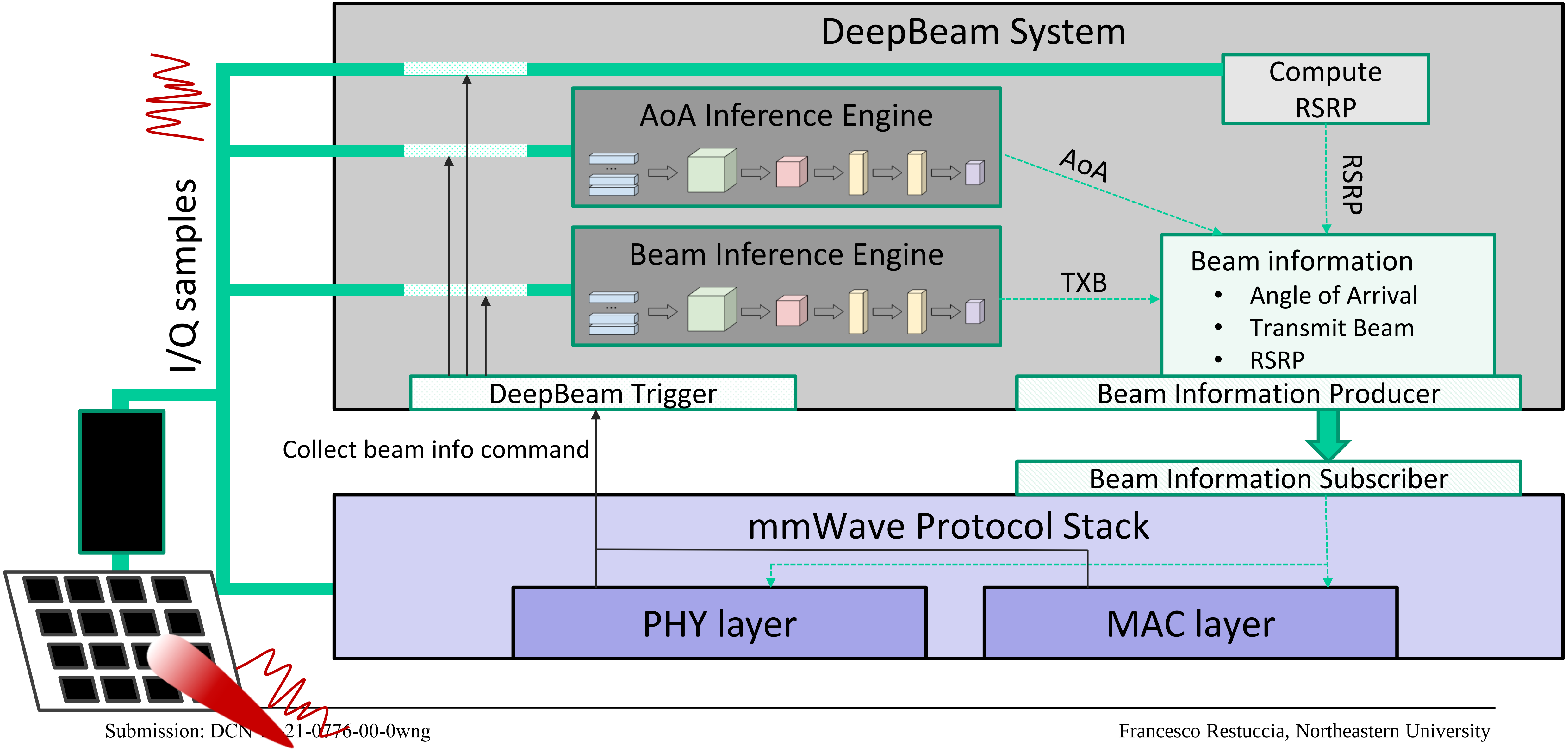


1 – Which beam is the transmitter using?

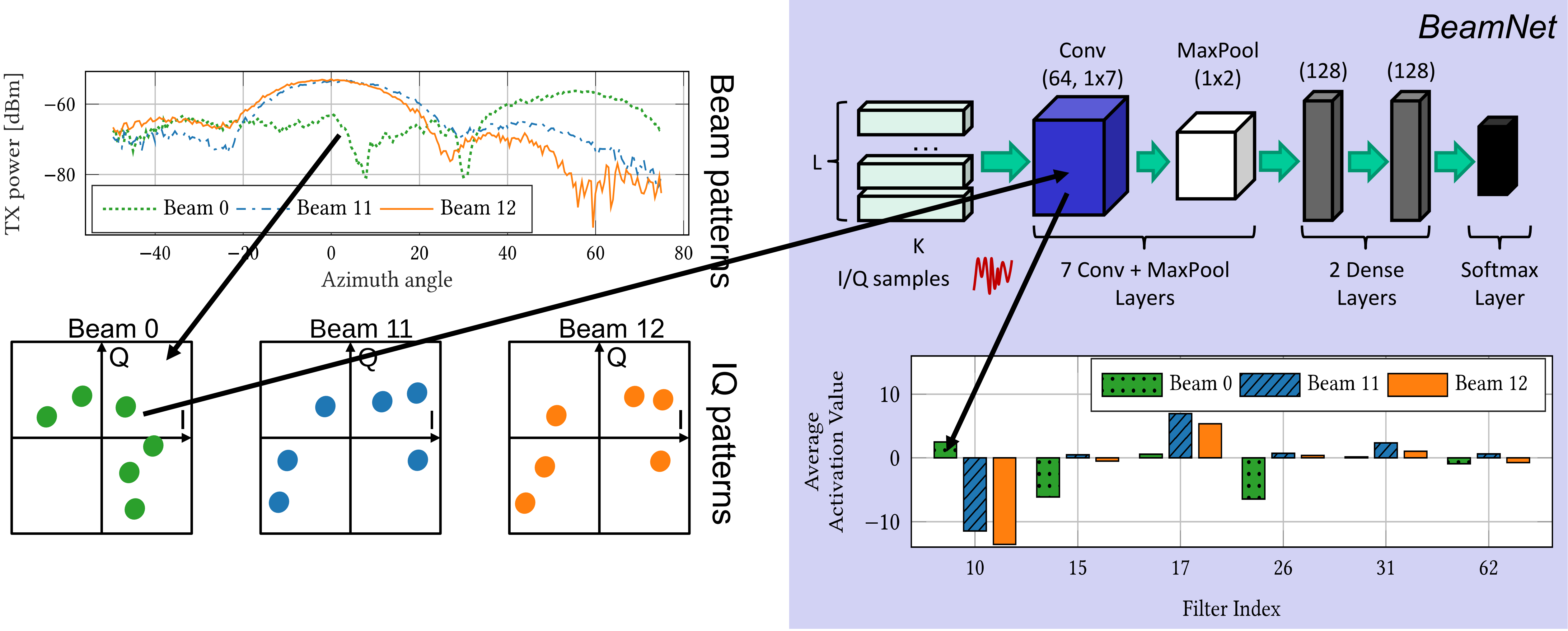
2 - What is the angle of arrival?

3 – Adaptation step

DeepBeam framework



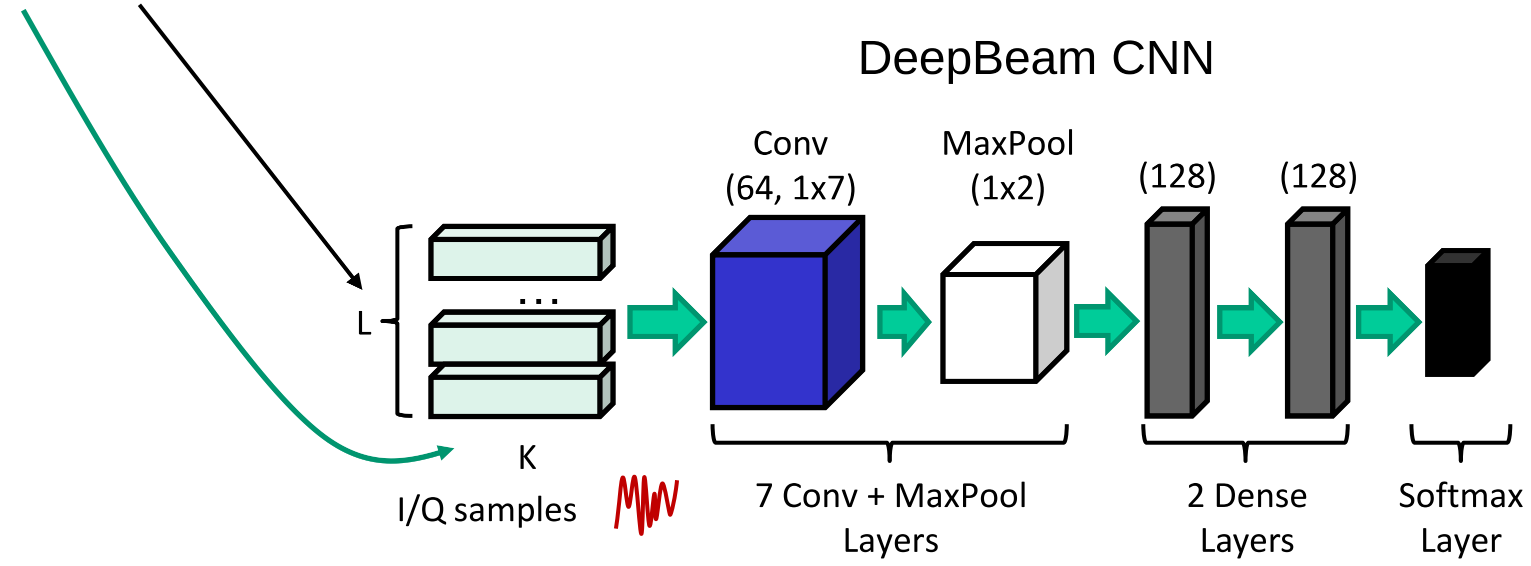
DeepBeam Inference Engine



Initial Access Latency for DeepBeam

DeepBeam eavesdrop ongoing transmissions

Need to collect $\xi = K \cdot L$ I/Q samples to perform classification



Initial Access Latency for DeepBeam

We also model the latency for the classification in the CNN

Exploit pipelining

$$T_{\mathrm{DB,c,e2e}} + (N_{tx} - 1)T_{\mathrm{DB,c,max}}$$

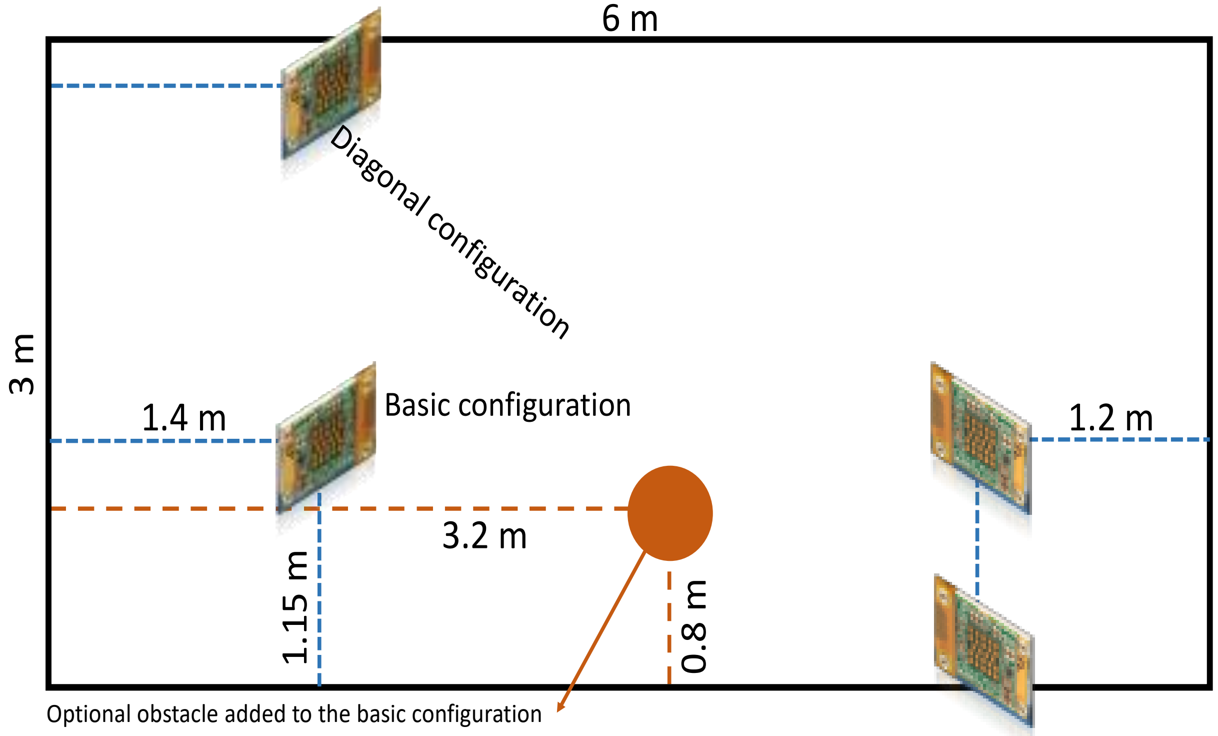
End-to-end CNN latency

Latency of the slowest layer

DeepBeam - Experimental results

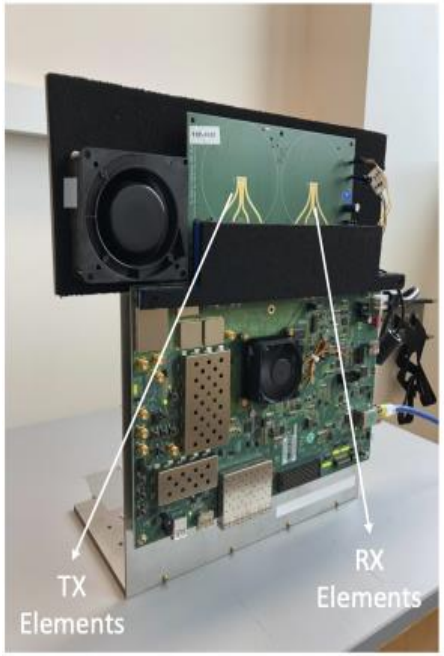
Multi-radio data collection at 60 GHz

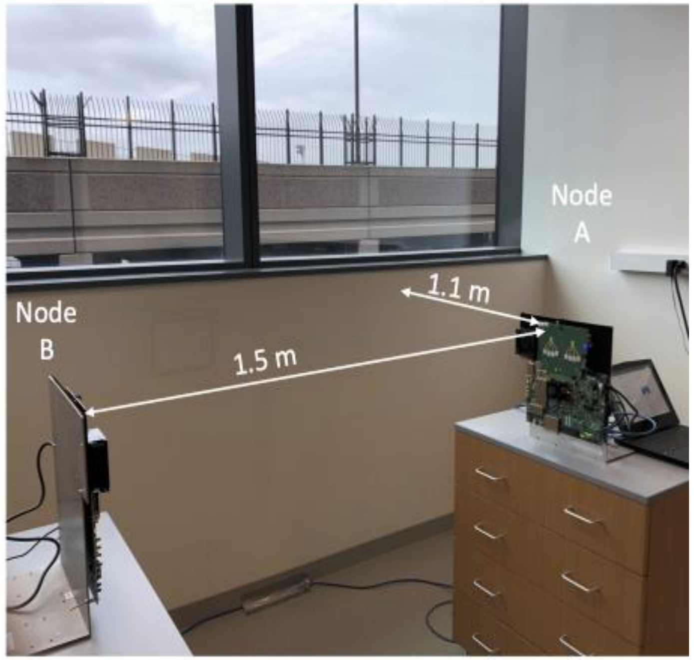
SiBeam/NI with analog phased arrays



4 different SiBeam frontends

Pi-radio SDR with digital beamforming





DeepBeam – Dataset

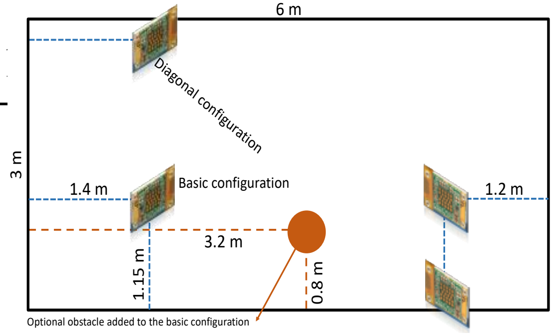
Classification target	TX Codebook	Testbed	Configuration	(TX, RX) antenna combinations
TXB	24-beams codebook	Single-RF-chain	Basic, with obstacle, diagonal	SiBeam (0, 1), (1, 0), (2, 1), (3, 1)
TXB	12-beams codebook	Single-RF-chain	Basic, with obstacle, diagonal	SiBeam (0, 1), (1, 0), (2, 1), (3, 1)
AoA	24-beams codebook	Single-RF-chain	Basic, with obstacle, diagonal	SiBeam (0, 1), (1, 0), (0, 2), (0, 3)
TXB	5-beams codebook	Multi-RF-chain	Multi-RF-chain basic	Node A, Node B

Table 1: Setups for the I/Q data collection.

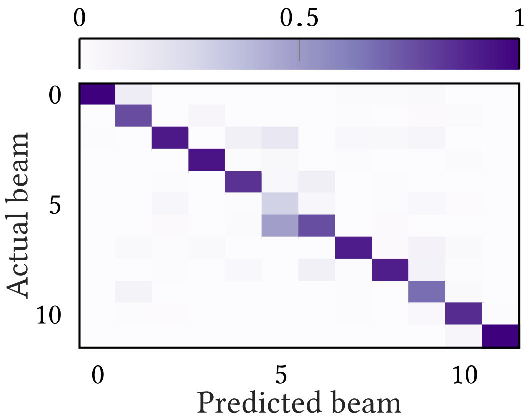
4 TB of raw I/Q samples

Different TX codebooks, antenna frontends, spatial configurations

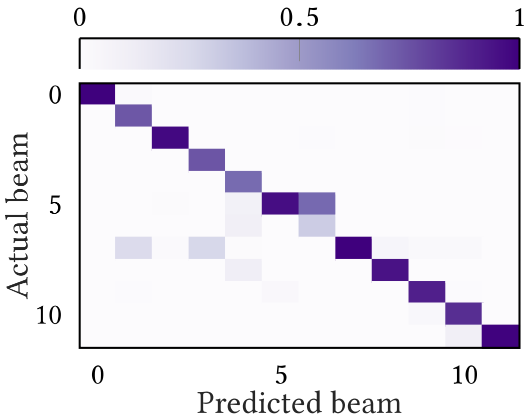
CNN trained with Adam optimizer, 60% training 40% testing



Classification Accuracy

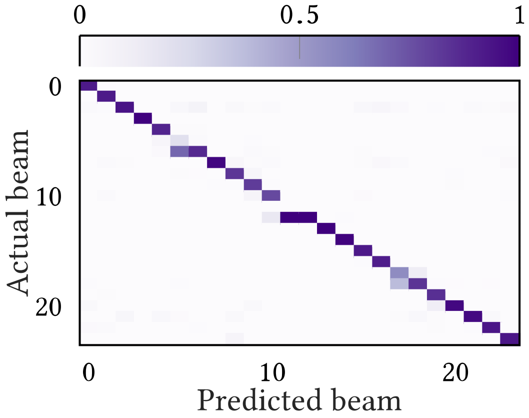


(a) 12-beam, $L = 1$, Accuracy: 81.02%

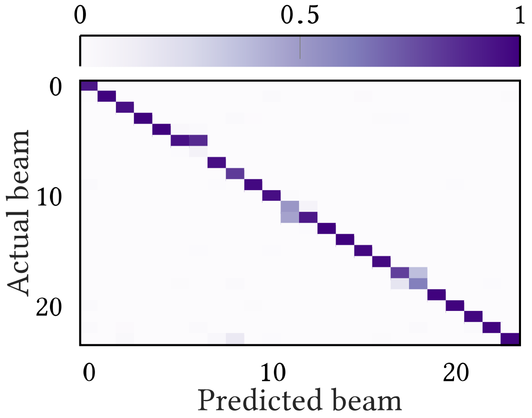


(b) 12-beam, $L = 5$, Accuracy: 84.02%

12 beam – 80% accuracy



(c) 24-beam, $L = 1$, Accuracy 68.77%

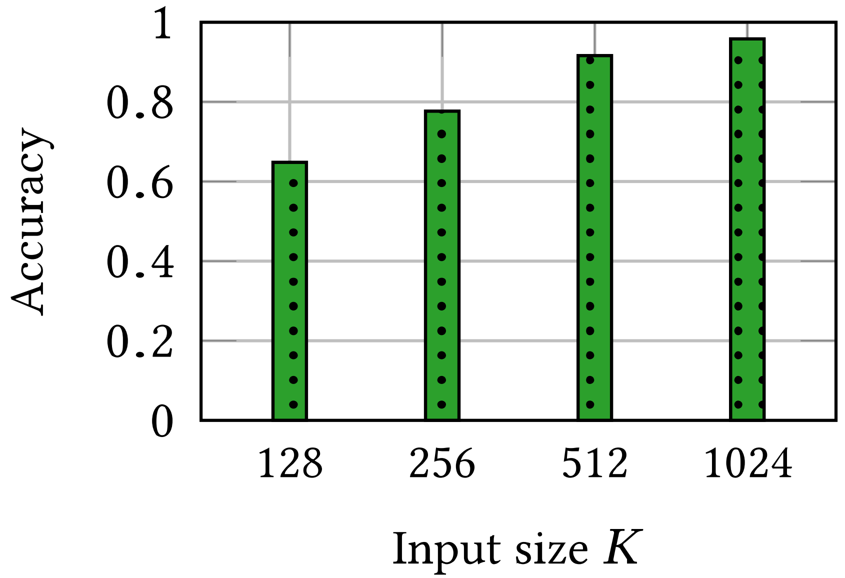


(d) 24-beam, $L = 5$, Accuracy: 77.46%

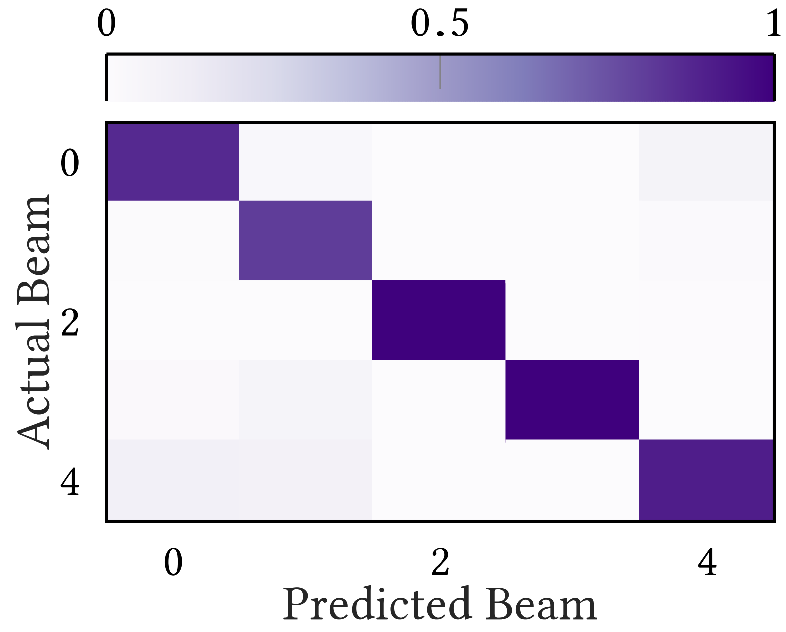
24 beam – 77% accuracy

NI/SiBeam radios, basic configuration, mixed SNR

Impact of input size



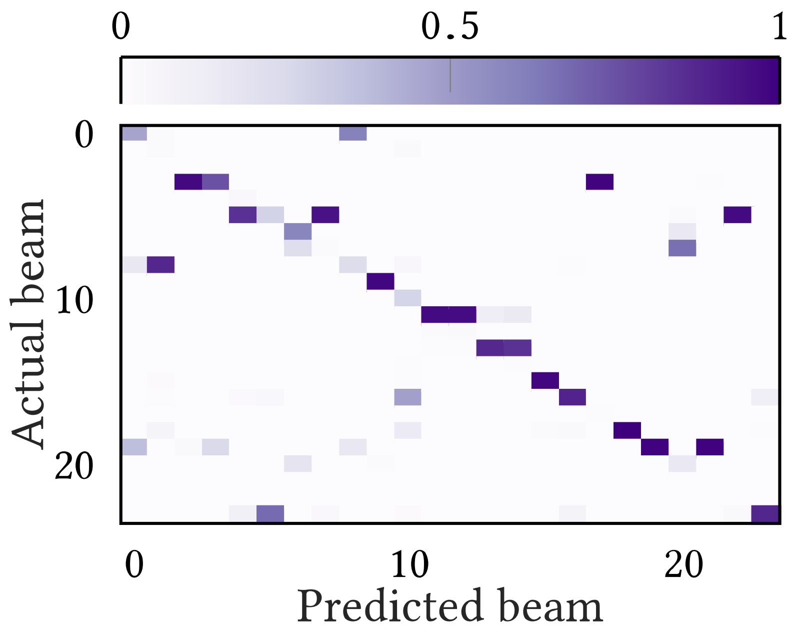
(a) Accuracy vs input size K.



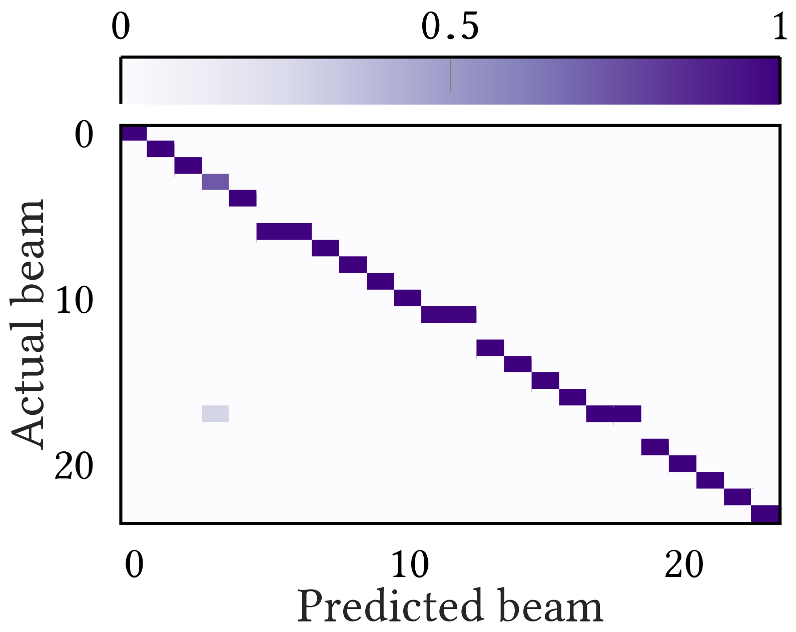
(b) Accuracy: 91.56%

Pi-Radios, basic configuration, mixed SNR

Impact of SNR

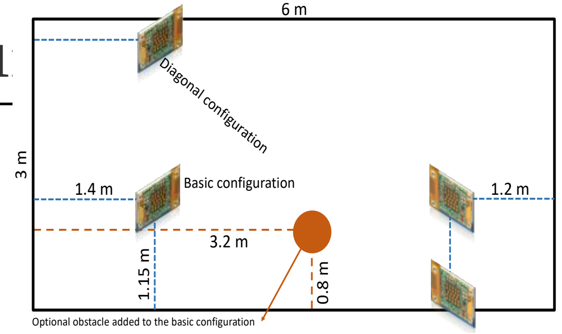


(a) Low SNR. Accuracy 43.47%

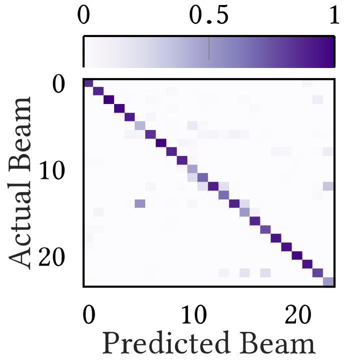


(b) High SNR. Accuracy: 86.36%

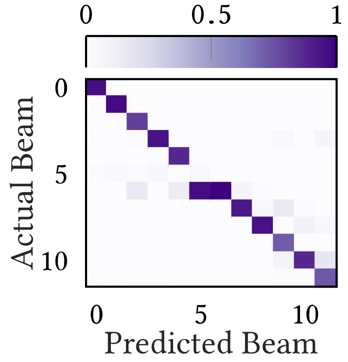
NI/SiBeam radios, basic configuration



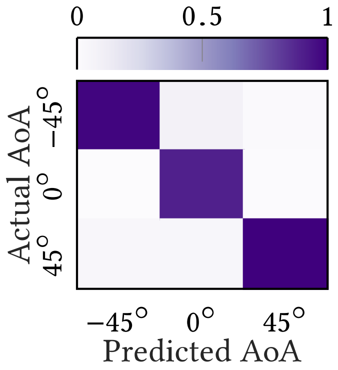
Impact of location



(a) 24-beam, TX Ant. 0
Accuracy: 81.84%

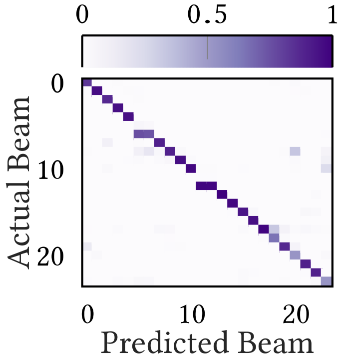


(b) 12-beam, TX Ant. 0
Accuracy: 80.89%

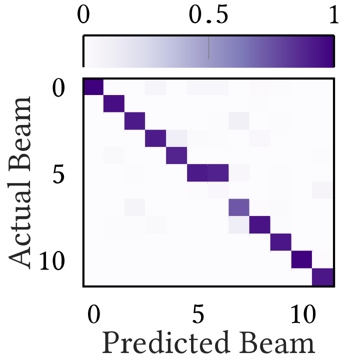


(c) AoA, RX Ant. 0
Accuracy: 92.95%

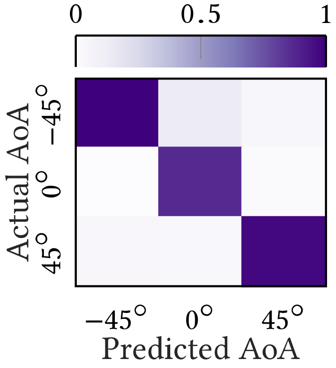
Diagonal configuration



(d) 24-beam, TX Ant. 0
Accuracy: 84.69%



(e) 12-beam, TX Ant. 0
Accuracy: 84.41%

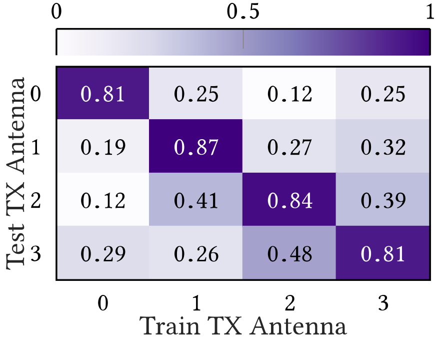


(f) AoA, RX Ant. 0
Accuracy: 88.13%

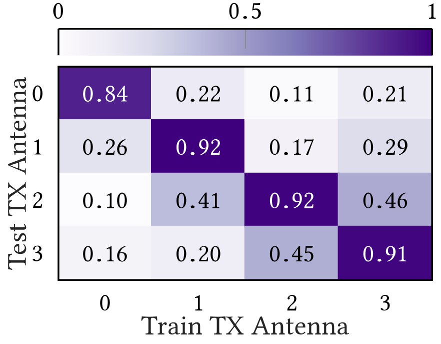
Obstacle configuration

NI/SiBeam radios, mixed SNR

Training and testing on different devices

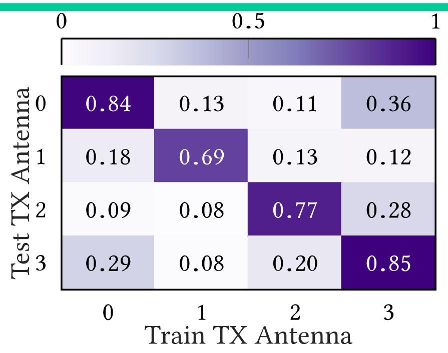


(a) **12-beam**, $L = 1$

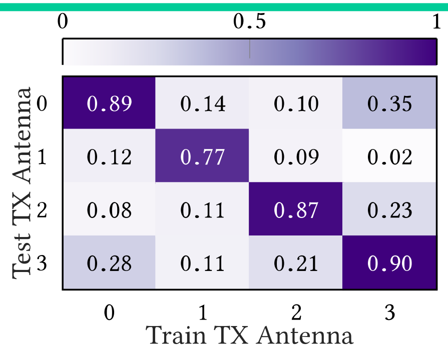


(b) **12-beam** $L = 5$

12 beam

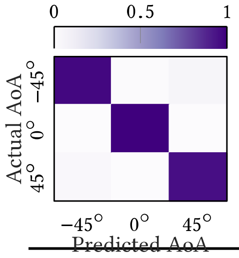


(c) **24-beam**, $L = 1$

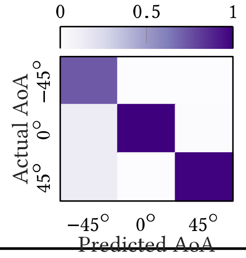


(d) **24-beam**, $L = 5$

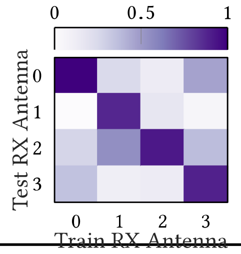
24 beam



(a) AoA, RX Ant. 0



(b) AoA, RX Ant. 1

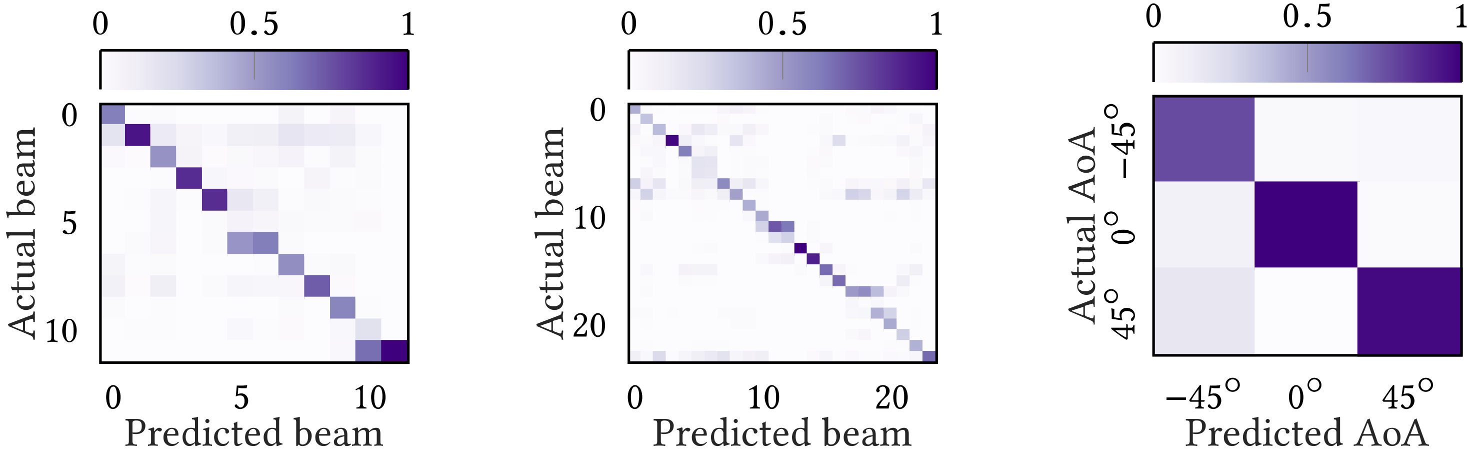


(c) TOTA, AoA

AoA

- Train on one device, test on another
- Features learned by the CNN are a mixture of antenna-based and antenna-independent
  - Accuracy decreases with mixed testing and training
  - Accuracy does not drop to random classification
  - More than 3x better than random

Training and testing with mixed dataset



(a) 12-beam, Accuracy: 62.69% **(b) 24-beam, Accuracy: 49.41%** **(c) AoA, Accuracy: 83.38%**

Mixed dataset with I/Qs from all 4 NI/SiBeam antennas

Increase accuracy of 124% (24-beam), 191% (12-beam), and 44% (AoA) with respect to TOTA

Conclusions and main takeaways

Deep waveform learning is effective at mmWaves

Enables pilot-less approaches that can improve overall performance

Future work

Develop **fine tuning** solutions to improve **generalization** capabilities	Test different **scenarios** and more **AoA** values

Dataset to be released soon, stay tuned!

Francesco Restuccia

Institute for the Wireless Internet of Things

Northeastern University

frestuc@northeastern.edu

https://restuccialab.org

https://northeastern.edu/wiot

Thanks! Questions?

DeepBeam – 3GPP NR use case

Initial access in 3GPP NR – cell search

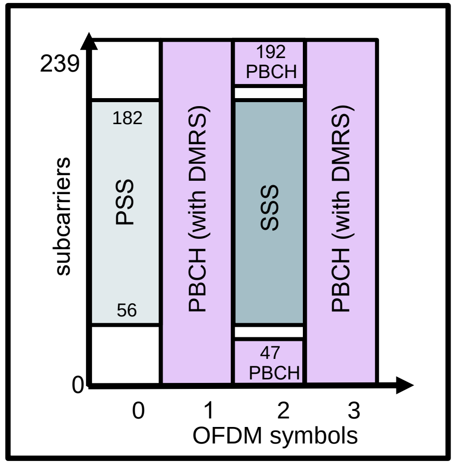
N_{SS}

Base stations transmit periodic sync signal (SS blocks)

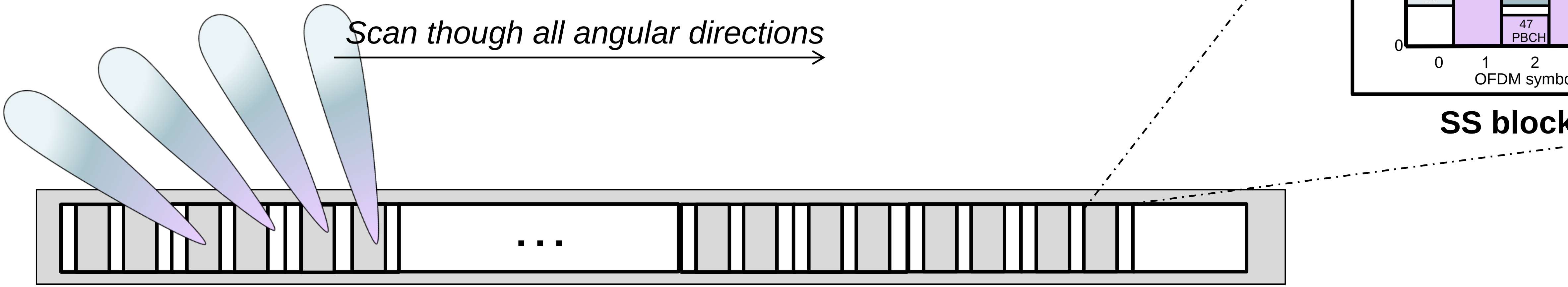
SS blocks are grouped in SS bursts (max 5 ms)

SS bursts are repeated with a certain periodicity

T_{SS}



SS block



SS burst

Initial Access Latency for 3GPP NR

Defined as *latency to perform a full IA/cell search scan*

Number of beams in TX codebook

Number of beams in RX codebook

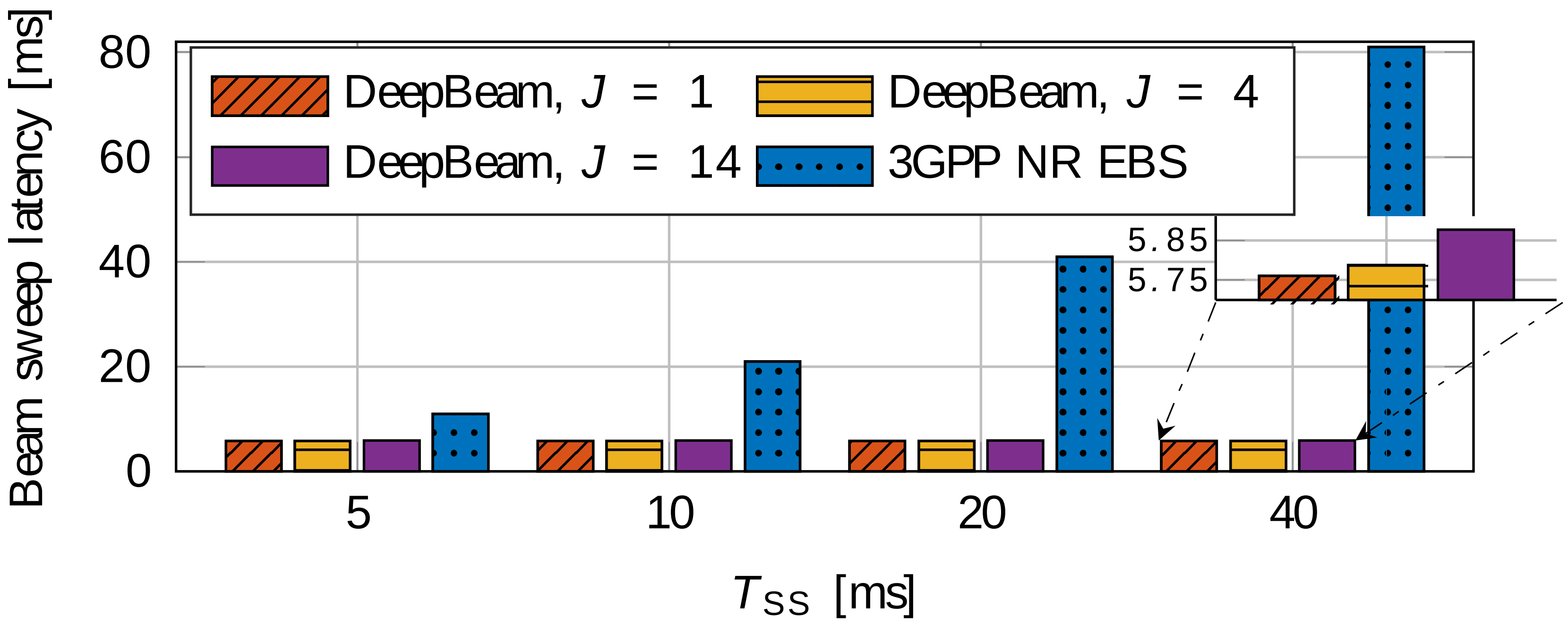
$$T_{\mathrm{EBS}} = T_{\mathrm{SS}}\left(\left\lceil \frac{N_{tx}M_{rx}}{N_{\mathrm{SS}}} \right\rceil - 1\right) + \hat{T}_{\mathrm{EBS}}$$

SS burst periodicity

SS blocks per burst

Initial Access Latency Results

- FPGA implementation of CNN (0.492 ms for e2e delay, 0.34 ms for slowest layer)
- Comparison with 12 beams at TX and RX, 3300 subcarriers (400 MHz bandwidth), 3GPP numerology 3, J is the number of symbols allocated to each user. 802.11ad is 0.2554



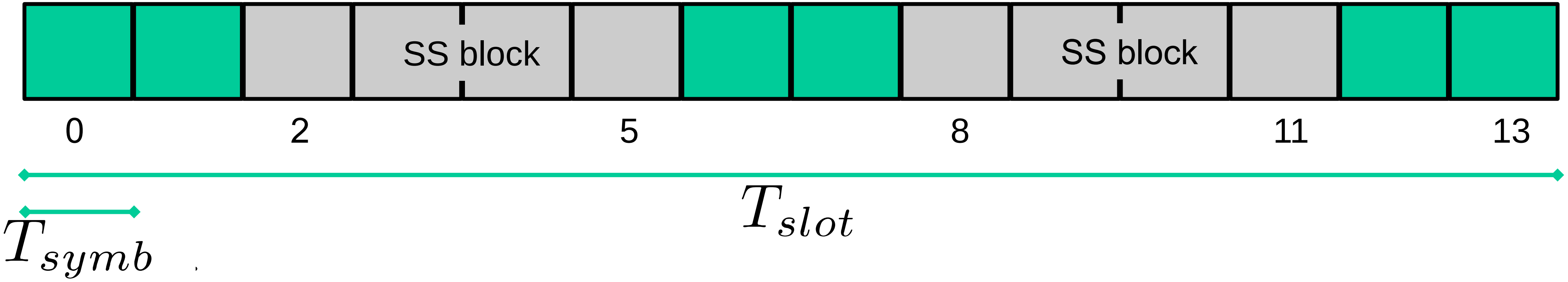
Initial Access Latency for 3GPP NR

$$T_{\mathrm{EBS}} = T_{\mathrm{SS}}\left(\left\lceil \frac{N_{tx}M_{rx}}{N_{\mathrm{SS}}} \right\rceil - 1\right) + \hat{T}_{\mathrm{EBS}} \longrightarrow$$

Time to scan the remaining SS blocks in the last SS burst

$$\hat{N}_{SS} = N_{tx}M_{rx} - (\lceil N_{tx}M_{rx}/N_{\mathrm{SS}} \rceil - 1)N_{\mathrm{SS}}$$

SS burst in a 3GPP NR slot



$$\hat{T}_{\mathrm{EBS}} = \begin{cases} \frac{\hat{N}_{SS}}{2} T_{slot} - 2T_{sym} & \text{if } \hat{N}_{SS} \bmod 2 = 0 \\ \left\lfloor \frac{\hat{N}_{SS}}{2} \right\rfloor T_{slot} + 6T_{sym} & \text{otherwise.} \end{cases}$$