ETH zürich

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

NextRAN-AI

ETH- Huawei Sweden

Marco Bertuletti	mbertuletti@iis.ee.ethz.ch
Yichao Zhang	yiczhang@iis.ee.ethz.ch
Mahdi Abdollahpour	mahdi.abdollahpout@unibo.it
Alessandro Vanelli-Coralli	avanelli@iis.ee.ethz.ch
Luca Benini	lbenini@iis.ee.ethz.ch

PULP Platform
Open Source Hardware, the way it should be!

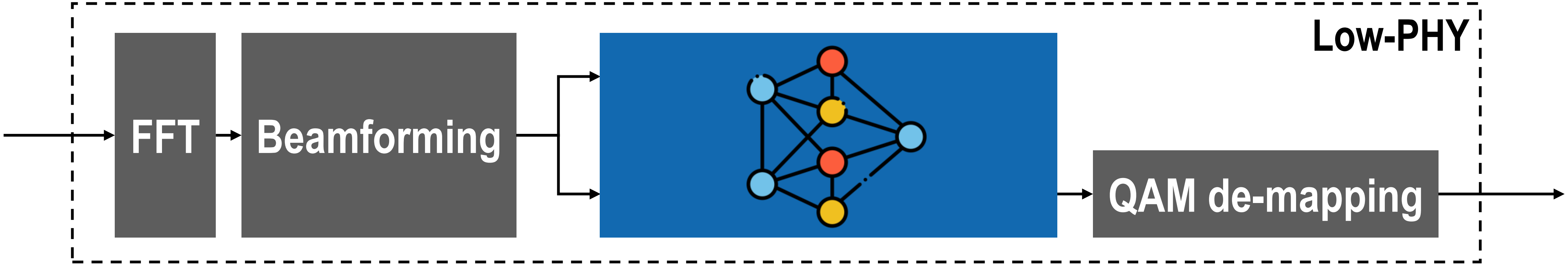
@pulp_platform
pulp-platform.org
youtube.com/pulp_platform

Outline

- Models currently under study
- Details on the model architecture
- Computational complexity of models

Focus on CSI and full MIMO AI-receivers

- **Channel State Information (CSI)**
  - Influences the performance of the receiver (BER vs SNR)
  - Must be performed **on the edge**, to avoid data transfer on the fronthaul and low-latency
  - Compute requirements scale with the MIMO-size (UEs/BW and number of antennas)
- **We target full MIMO receivers → full implementation of the low-PHY**
  - Direct comparison with the work on PUSCH
  - Partial **model-driven** and **data-driven** rx, depending on blocks with highest perf. gains



Models currently under study

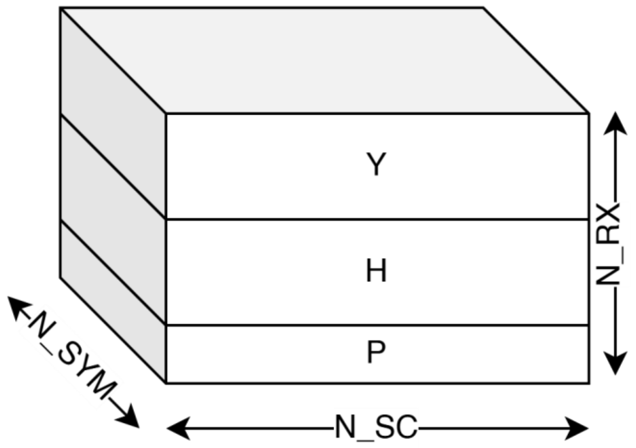
Name	Processing	NSC	NRXxNTX	Modulation	Model	Gain wrt conventional receiver @BER10^{-3}
Deep-RX SIMO	Ch.Est. + Det.	312	2x1	16QAM	ResNet	2.5 dB *
Deep-RX MIMO	Ch.Est. + Det.	312	16x4	16QAM	ResNet	2.5 dB *
Neural-RX	Ch.Est. + Det.	1584	4x2	16QAM	CGNN	1.7 dB *
Neural-RX RT	Ch.Est. + Det.	1584	4x2	16QAM	CGNN	1.0 dB *
… Extend to more subcarriers, RX, TX for B5G use-cases						

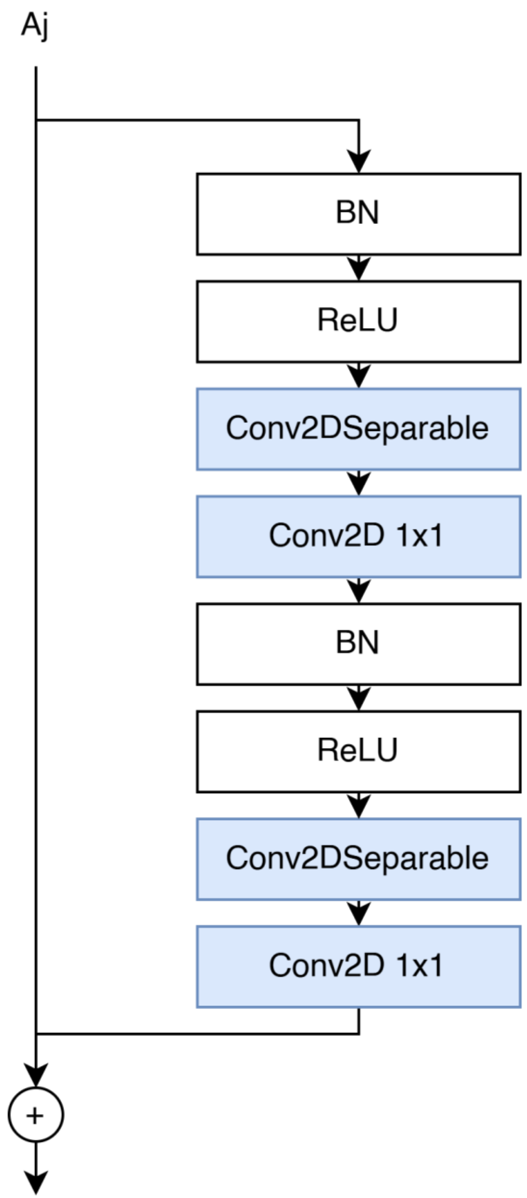
* LS Channel Estimation + LMMSE Detection

1. DeepRX-SIMO: architecture

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9345504

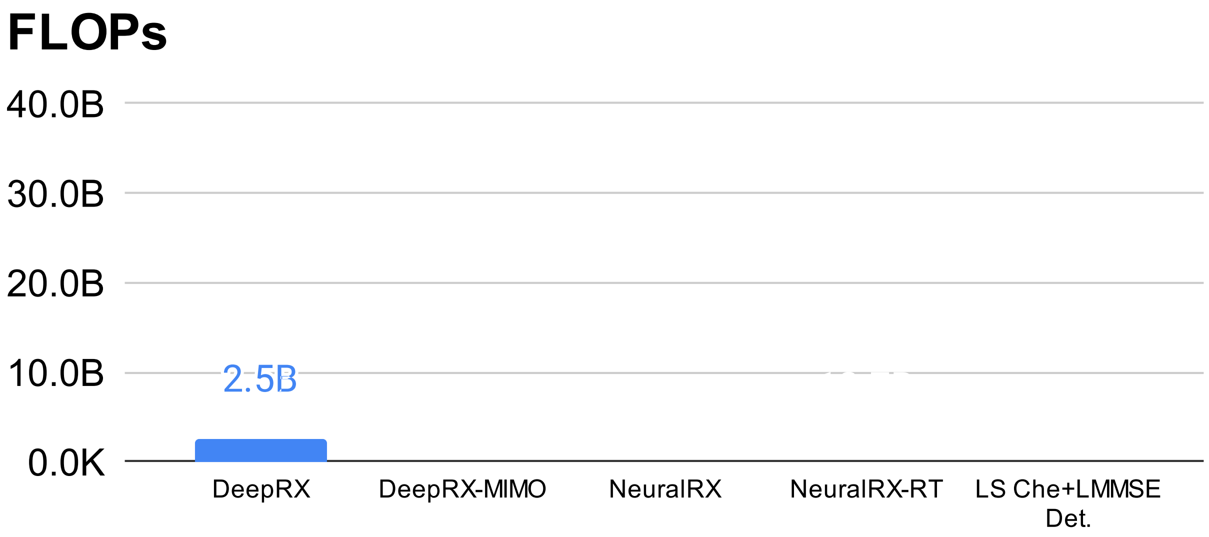
- Channel Estimation + DMR extraction + Detection
- Concatenate inputs, channel and pilots
- Based on **ResNet** (Depthwise separable convolutions + ReLU activation)

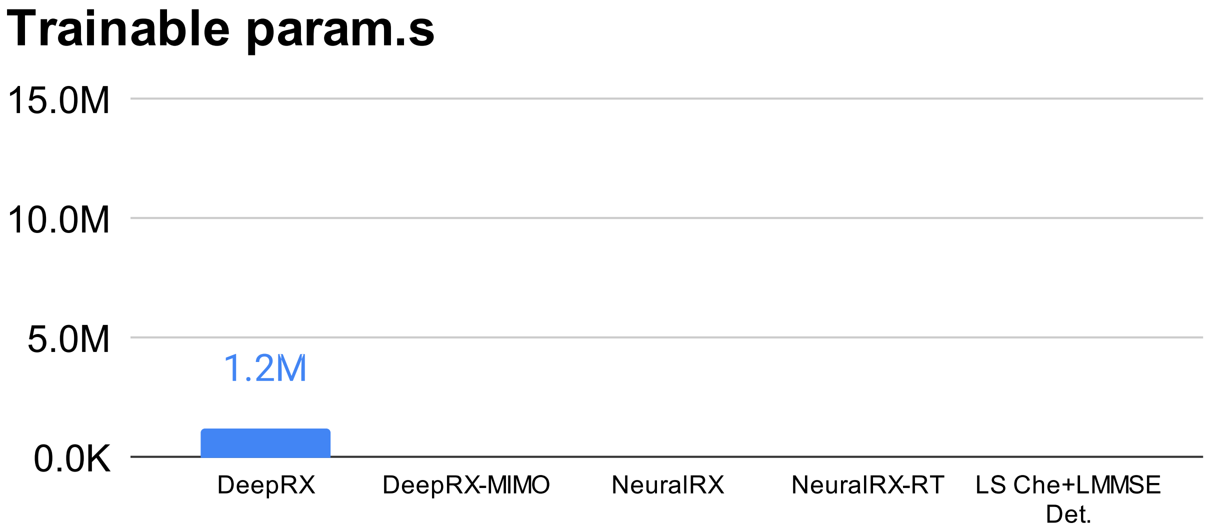




1. DeepRX-SIMO: summary

Parameters	
NRX x NTX	2x1
NSC	312
Modulation	16QAM
Channel Evaluation	TDL-A, TDL-E

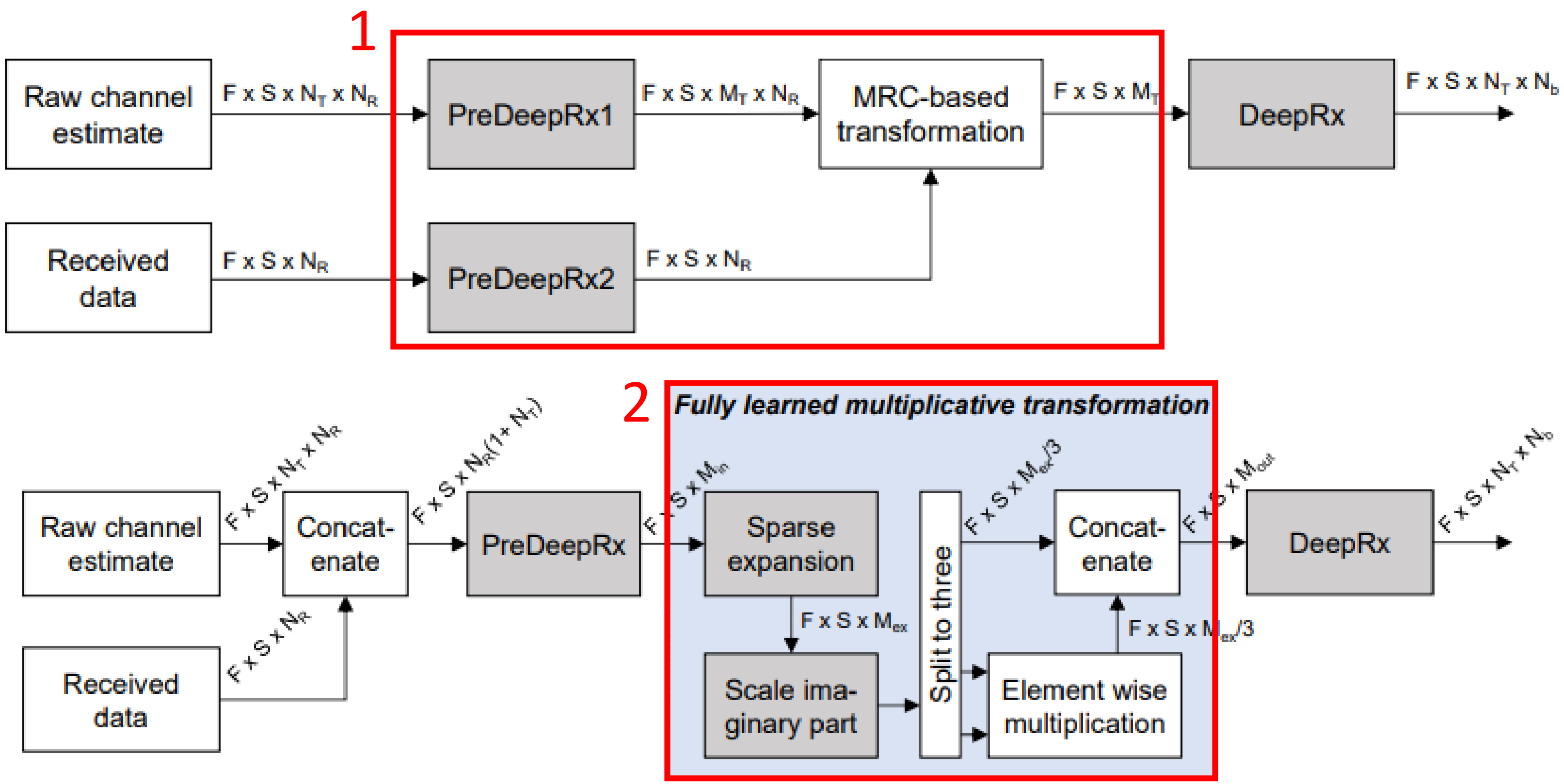




2. DeepRX-MIMO: architecture

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9500518

- Channel Estimation + DMR extraction + Detection
- Extension of DeepRX to handle multiple spatial streams

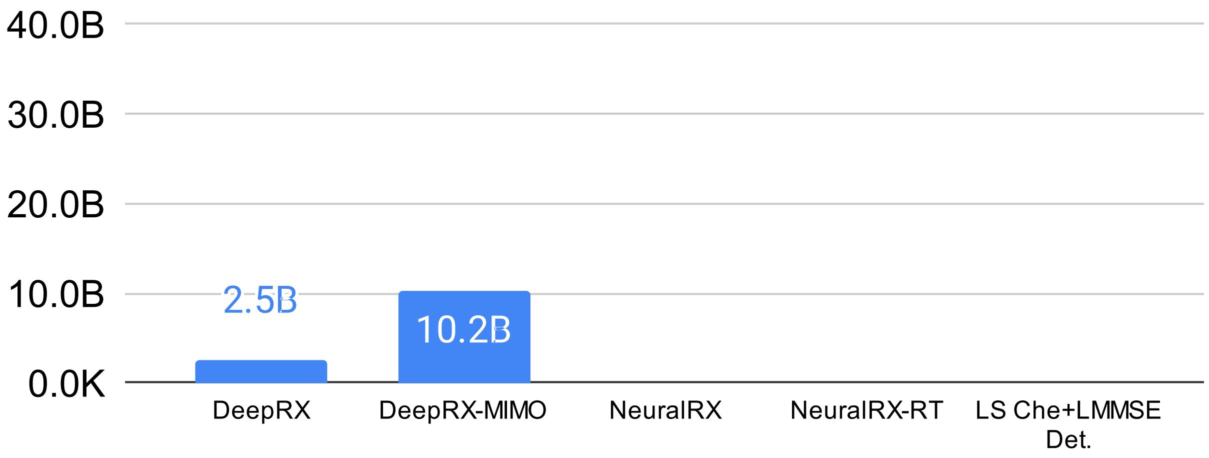


1. **MRC (Minimum-Ratio combining)** = partial equalization, hypothesis that TX experience orthogonal channel realizations.
2. **Learned sparse multiplication** + partition in 3 streams and multiplication.

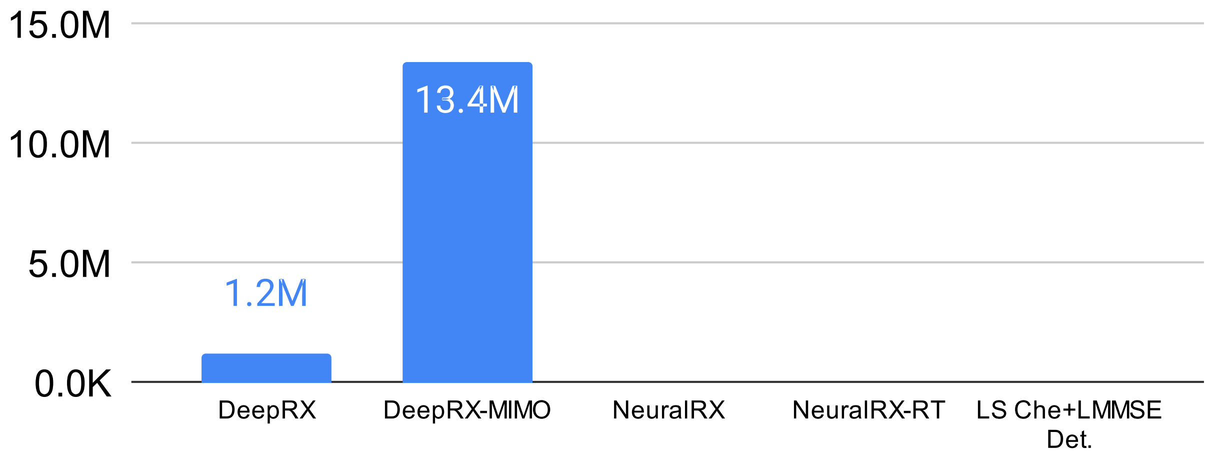
2. DeepRX-MIMO: summary

Parameters	
NRX x NTX	16 x 4
NSC	312
Modulation	16QAM
Channel Evaluation	TDL-A, TDL-E

FLOPs

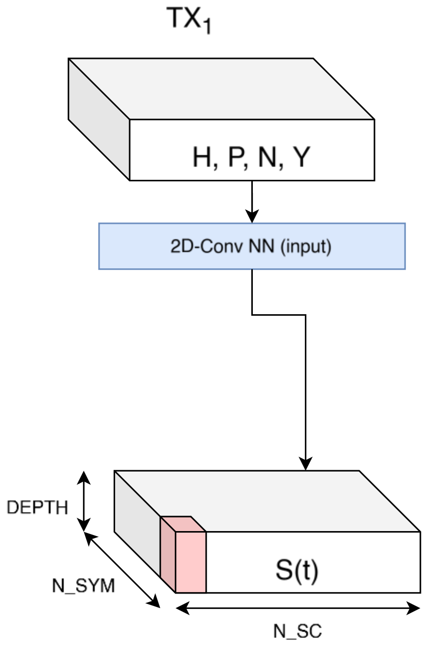


Trainable param.s



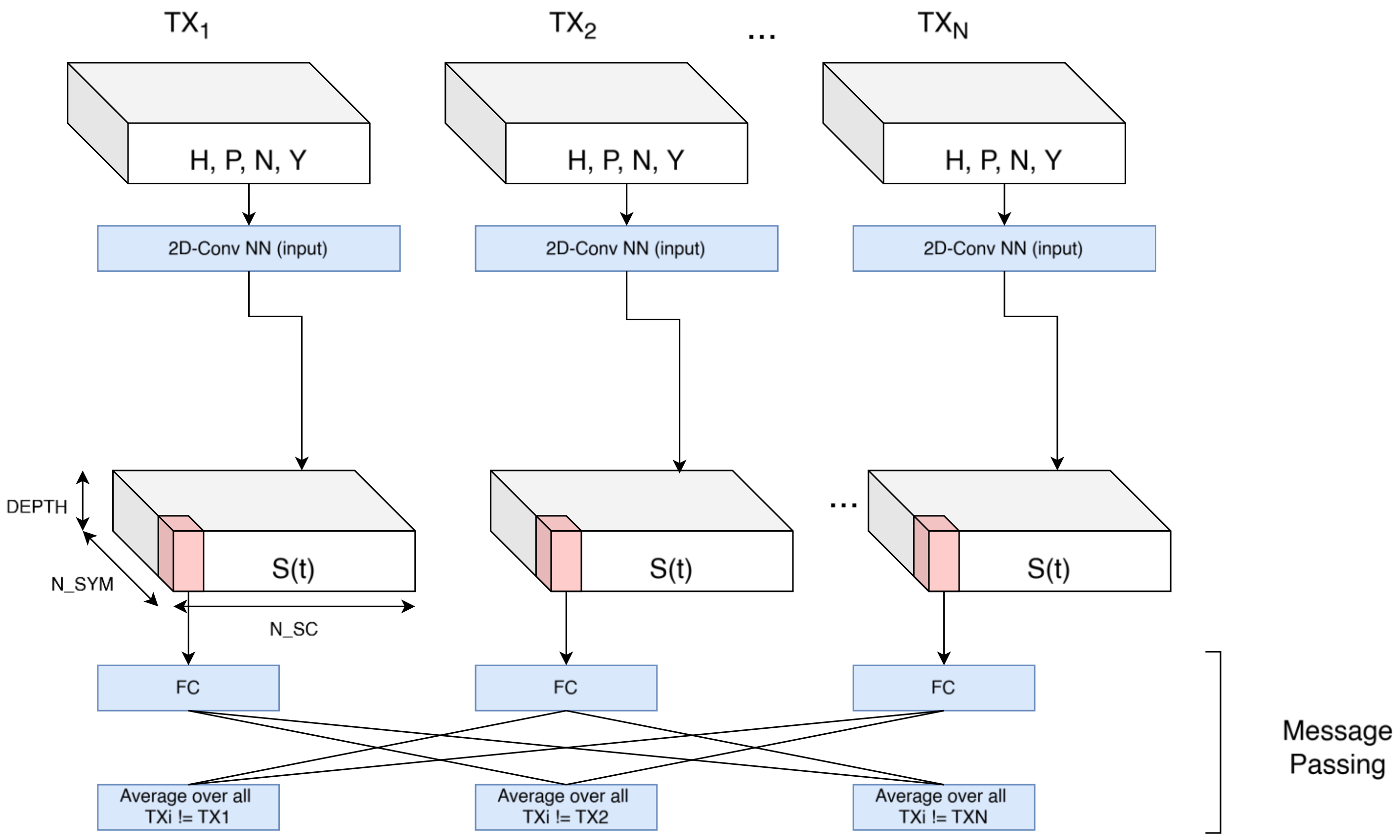
3. NeuralRX: architecture

https://arxiv.org/pdf/2312.02601



1. Concatenation of inputs and input CNN (ResNet based)

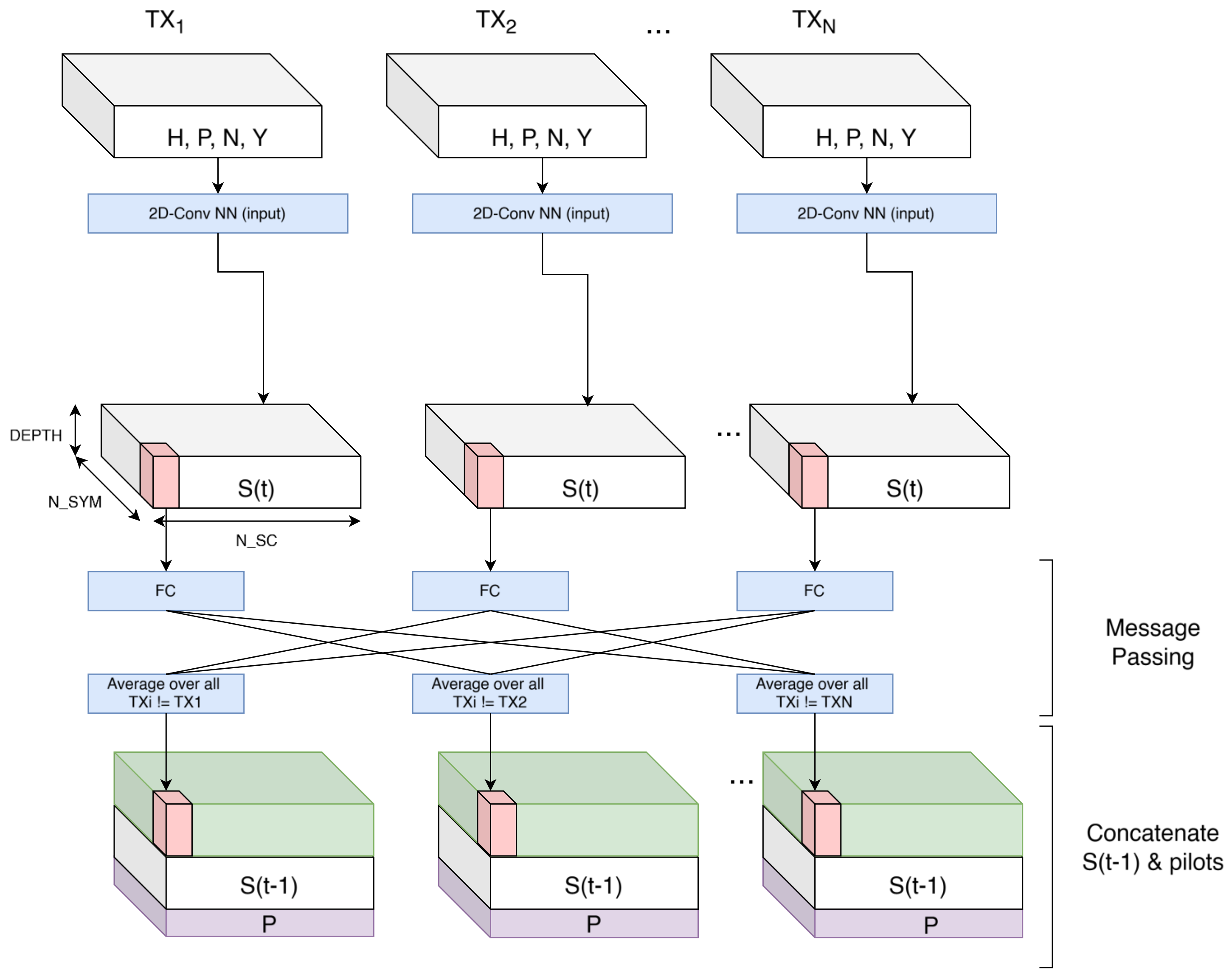
3. NeuralRX: architecture



https://arxiv.org/pdf/2312.02601

1. Concatenation of inputs and input CNN (ResNet based)
2. Fully-Connected layer over the depth of «state-tensor»
3. Message-Passing = averaging on the TX dimension

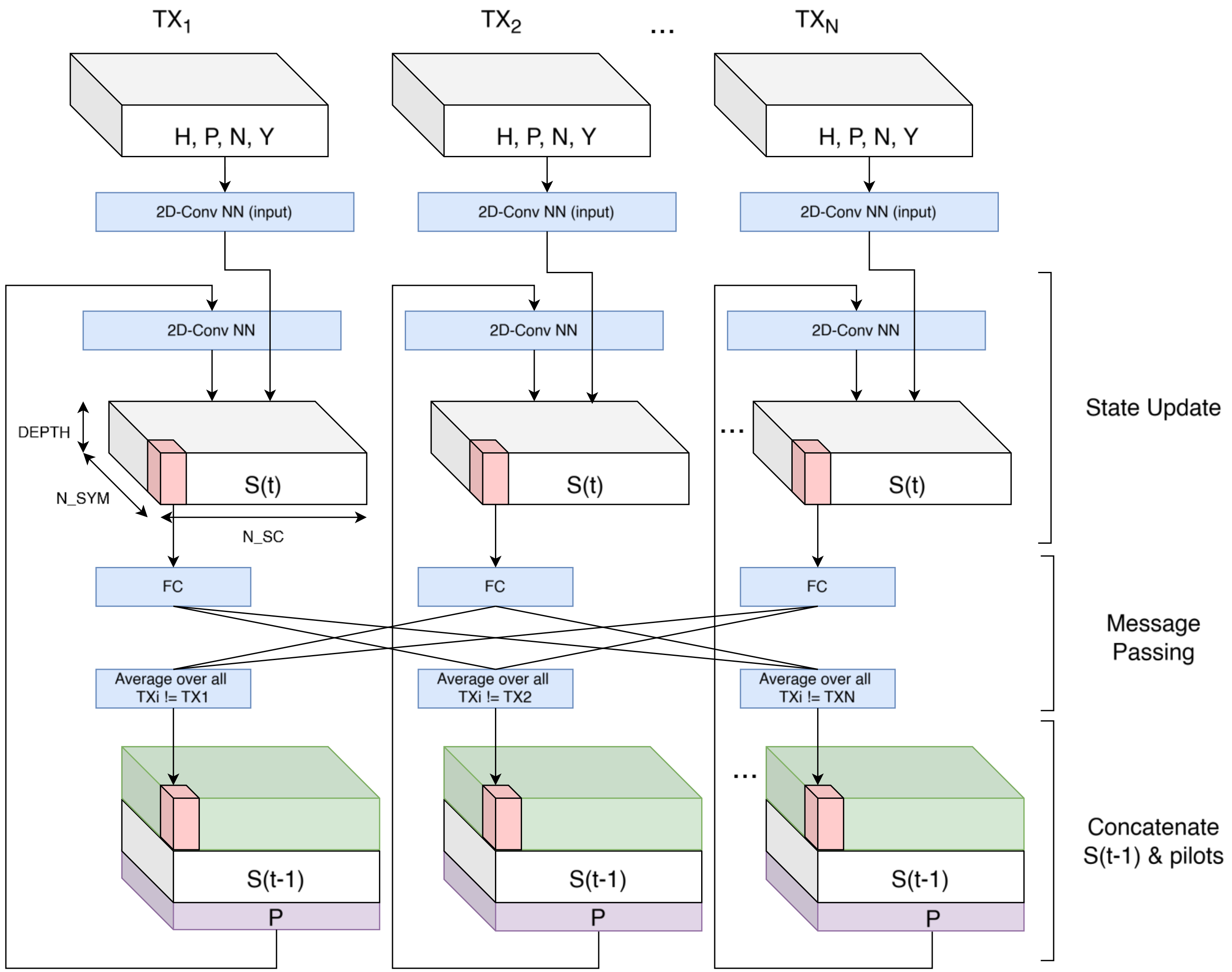
3. NeuralRX: architecture



https://arxiv.org/pdf/2312.02601

1. Concatenation of inputs and input CNN (ResNet based)
2. Fully-Connected layer over the depth of «state-tensor»
3. Message-Passing = averaging on the TX dimension
4. Concatenation with previous state + pilots

3. NeuralRX: architecture

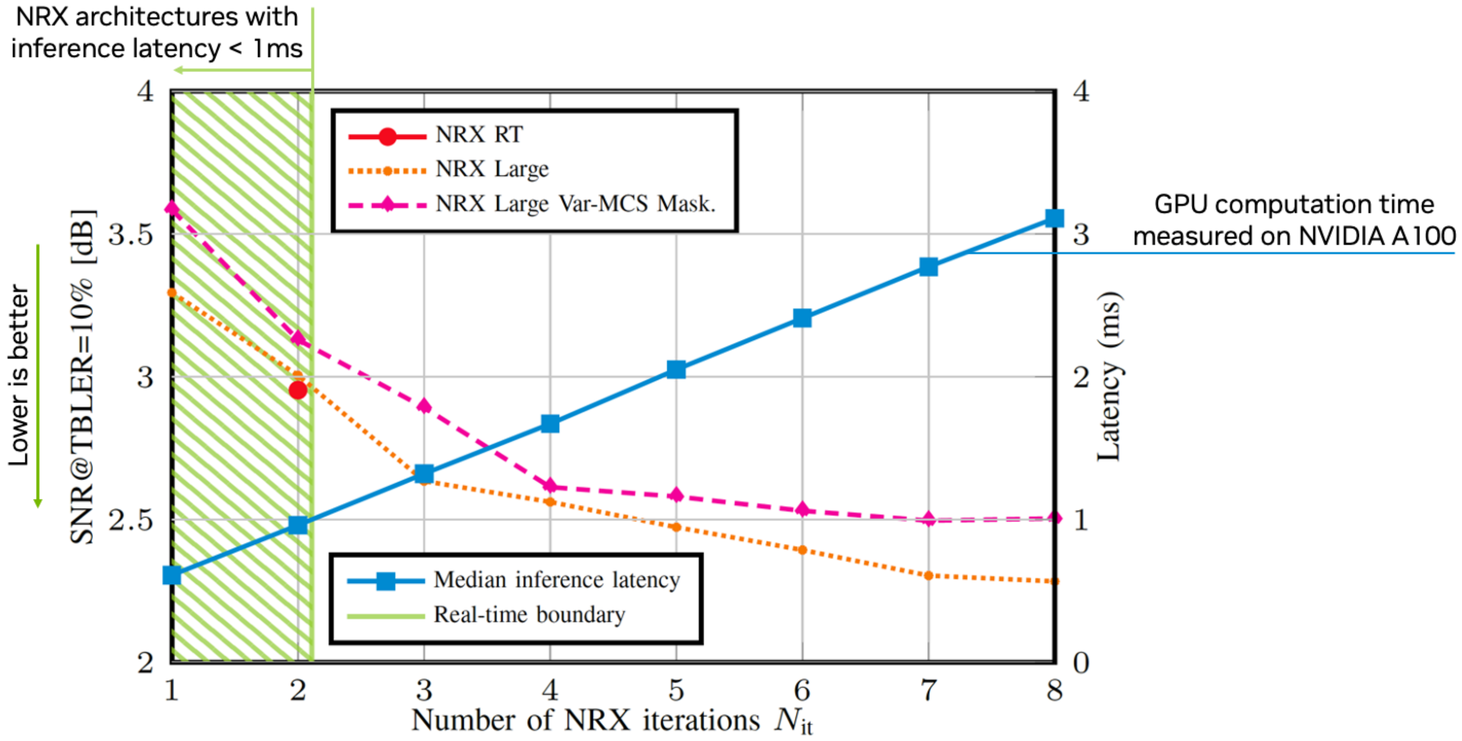


https://arxiv.org/pdf/2312.02601

1. Concatenation of inputs and input CNN (ResNet based)
2. Fully-Connected layer over the depth of «state-tensor»
3. Message-Passing = averaging on the TX dimension
4. Concatenation with previous state + pilots and «state-update»

4. NeuralRX-RT

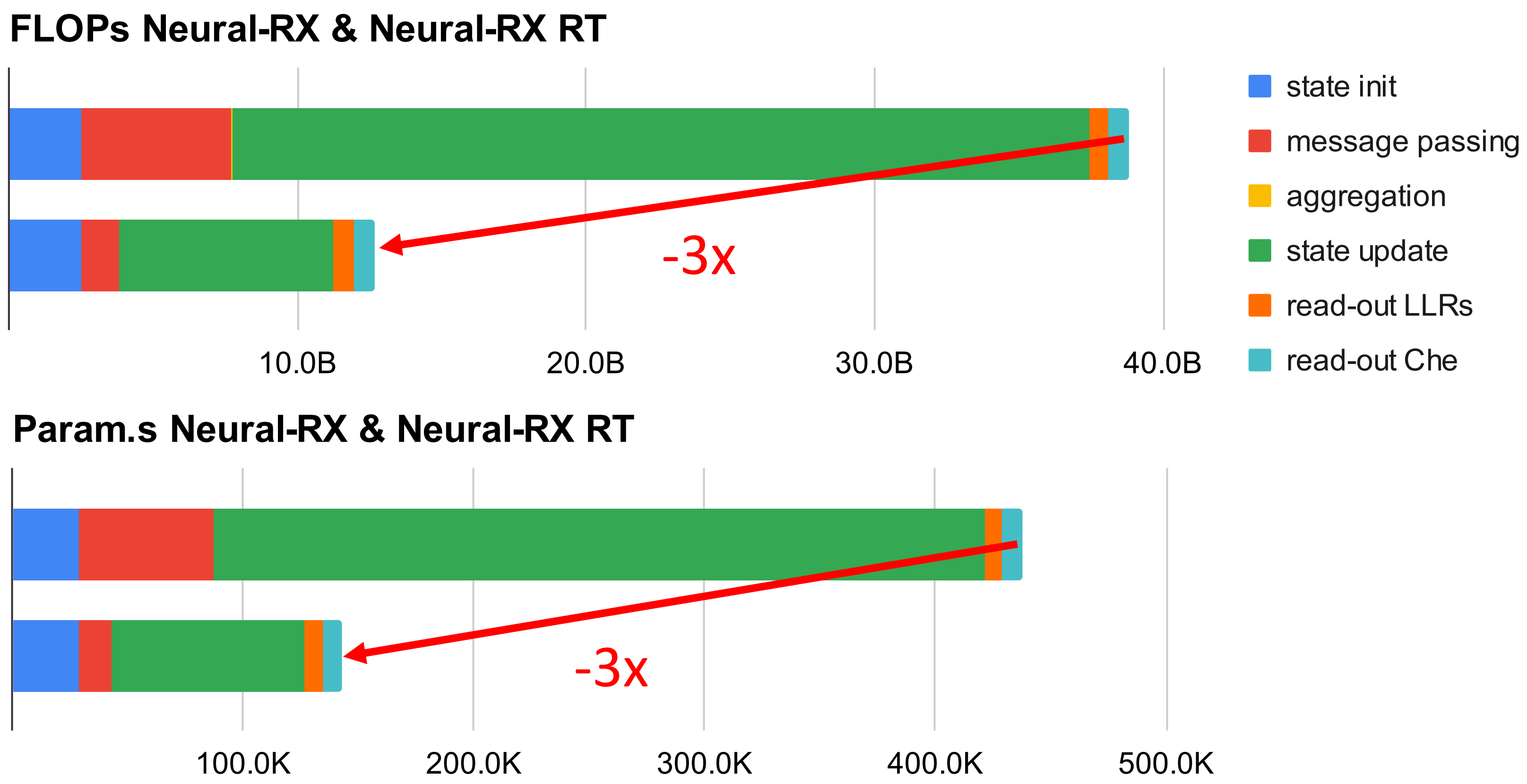
https://arxiv.org/pdf/2409.02912



Extension of NeuralRX for Real-Time execution:

- Target 1ms latency → reduce number of state-update iterations (higher BER)
- Add site-specific fine-tuning (few thousands iterations and data-samples)

4. NeuralRX-RT



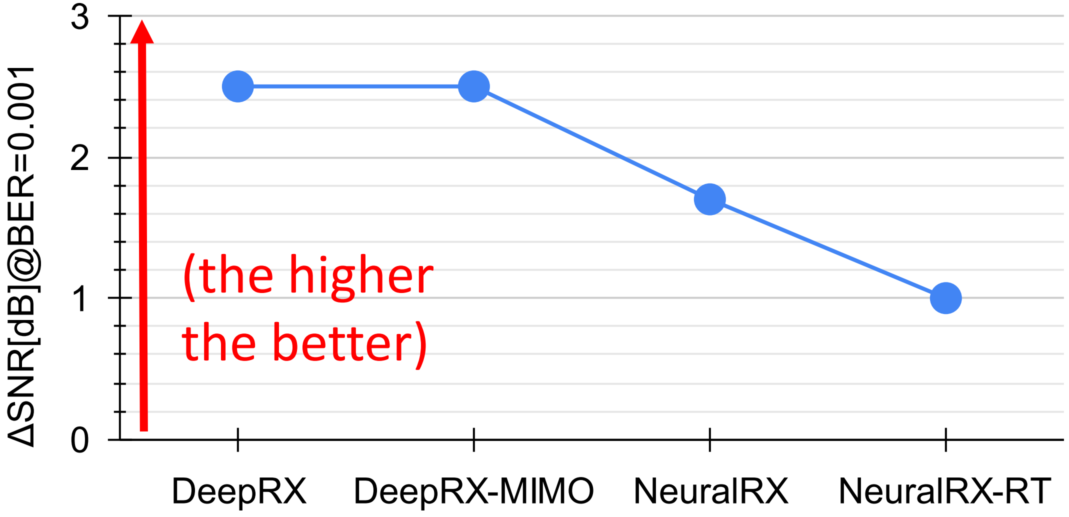
Extension of NeuralRX for Real-Time execution:

- Target 1ms latency → reduce number of state-update iterations (higher BER)
- Add site-specific fine-tuning (few thousands iterations and data-samples)

3/4. NeuralRX: summary

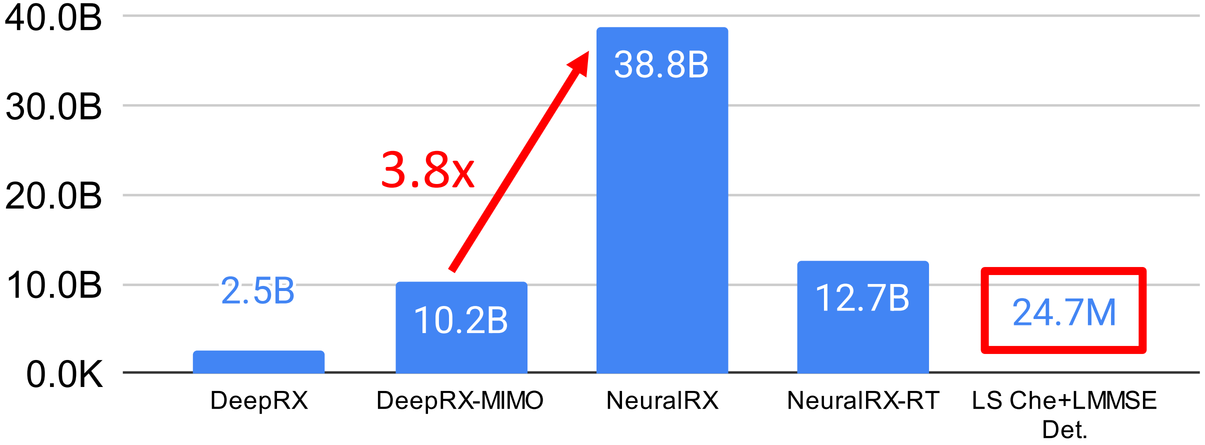
Parameters	
NRX x NTX	4 x 2
NSC	1584
Modulation	16QAM
Channel Evaluation	TDL-B, TDL-C

ΔSNR[dB]@BER=0.001 vs LS-Che + LMMSE-Det.

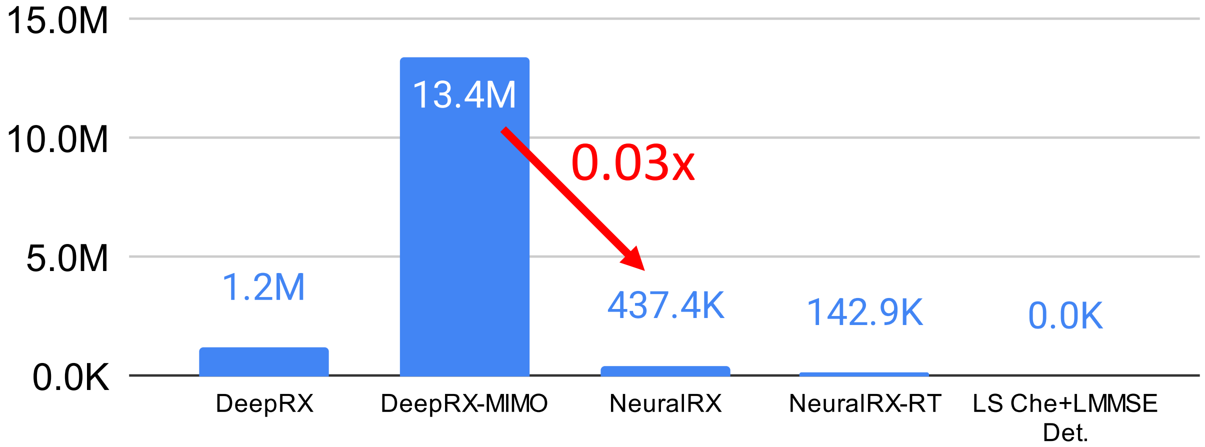


Computational complexity for conventional algorithms far below AI-models

FLOPs



Trainable param.s



We choosed to explore NeuralRX

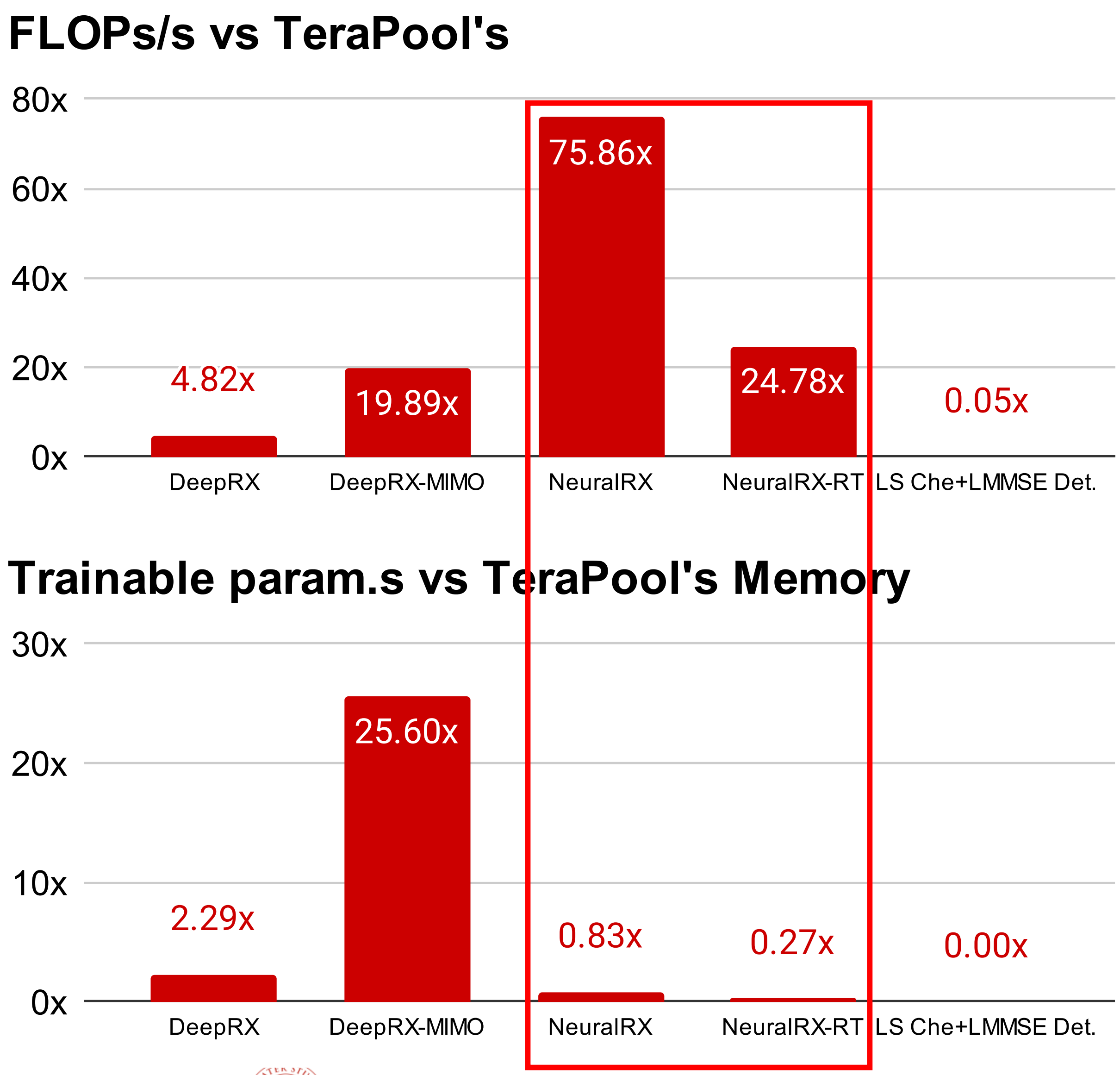
Advantages of NeuralRX over other models

- **Flexible** = the same trained model supports different number of users, different number of subcarriers, different modulation schemes
- It generalizes well to many different channel models
- It is open-sourced and tested already on a real-time and standard compliant scenario (NeuralRX RT)

Open-issues & Next Steps:

- Reduce model size and computational complexity for edge-deployment
- Possibly extend to more subcarriers, transceivers

3/4. NeuralRX on TeraPool



The number of operations per cycle required to TeraPool skyrockets.

However the memory required to store the trainable parameters is adequate.

→ **Need to push the performance**

We choosed to explore NeuralRX

Advantages of NeuralRX over other models

- **Flexible** = the same trained model supports different number of users, different number of subcarriers, different modulation schemes
- It generalizes well to many different channel models
- It is open-sourced and tested already on a real-time and standard compliant scenario (NeuralRX RT)

Next Steps:

- Reduce model size and computational complexity for edge-deployment
- Possibly extend to more subcarriers, transceivers
- **Adequate TeraPool's computation per cycle**