ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

EE 436 Communications II

Term Project

Şevket Ege Yeşilyurt 2517233
Ömer Topcuoğlu 2743839

Introduction

- **Goal:** Joint channel estimation and signal detection in OFDM with QPSK using deep learning.

- **Phase I:**
  - Reviewed model in the paper.
  - Configured WINNERII channel.
  - Customized C2.
  - Generated OFDM–QPSK dataset.

- **Phase II:**
  - Implemented and trained DNN.
  - Compared with LS and MMSE.
  - Evaluated BER under:
    1. Pilot number variation
    2. Cyclic prefix omission
    3. Clipping noise
    4. Mismatched channel statistics

Literature Review

- No need for explicit channel state information (CSI).
- DNN learns channel behavior implicitly from data.
- Outperforms LS, and matches or exceeds MMSE.
- More robust under challenging conditions:
  - Sparse pilots
  - No cyclic prefix
  - Clipping noise
  - Validation Mismatch

Wireless Channel Modelling and Generation

Layout Parameter Configuration

- BS and MS,50 m apart
- WINNER II: "C2 – Typical Urban Macro-cell

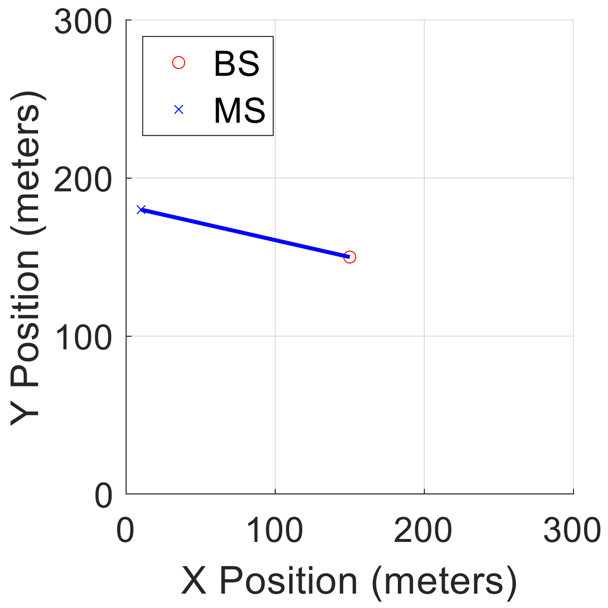
Model Parameter Configuration

- Cf = 2.6 GHz
- Shadowing & path-loss on
- Intra-cluster delay spread on: 24 paths

Delay-spread(DS) < 16 Ts Calculation

- C2's log-normal DS: $\mu = -6.6$, $\sigma = 0.32$
- $P(DS < k) = CDF(k) = 0.9999 \Rightarrow DS \approx 3.63\ \mu s$
- 16*Ts = 16/fs_max = 3.63μs, fs_max ≈ 4.4 MHz
- MS velocity = (fs_max * C) / (4000000*f_carrier)
- MS velocity = 0.127 m/s

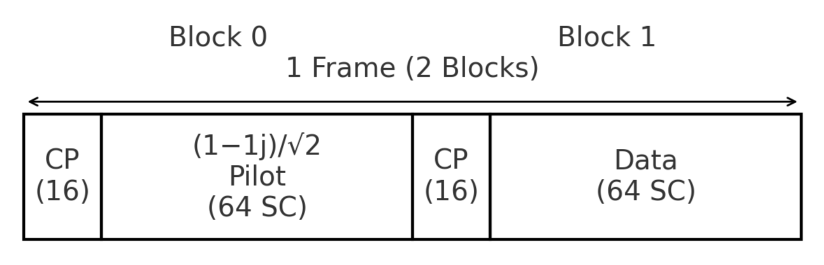
$(\text{max delay} - \text{min delay}) \times fs = \text{maximum delay in Ts}$

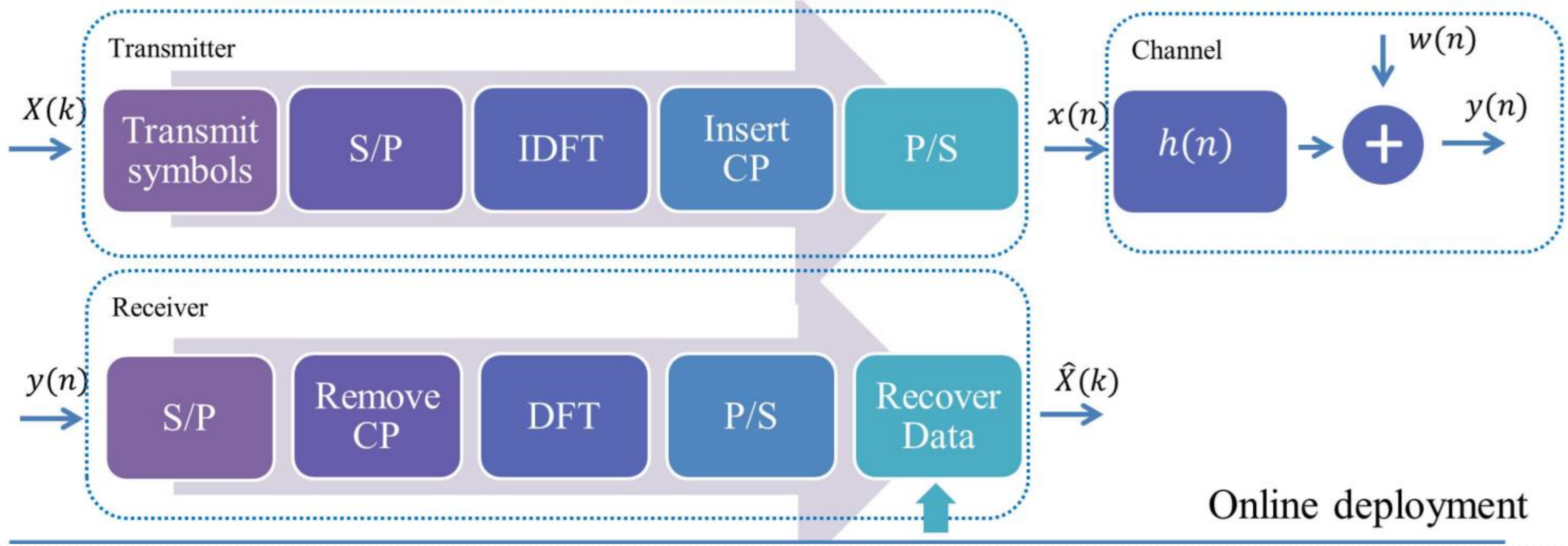


Generation

- Initialize channel object for each data (may decrease performance)
- Save channel impulse responses
- Generate data with convolution

System Architecture





DNN Architecture and Training

SNR dB	Data Pair Size	Training Ratio
5:5:25	200k	0.8

- ⟨received symbols (X[k] H[k]), original symbols(X[k])⟩

- Input is (64+64)*2 = (1 frame)*2 = 256, since complex
- Output is 8 QPSK, so 16 bits

$$x' = \frac{x - \mu}{\sigma}$$

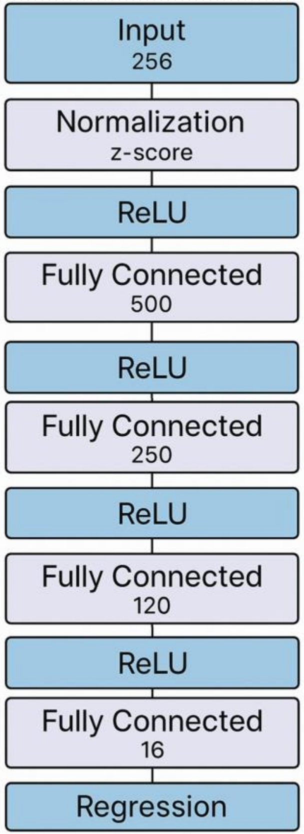
z-score to input (256)
mean and std of input (256)

$$\hat{x}^{(i)} = \frac{x^{(i)} - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

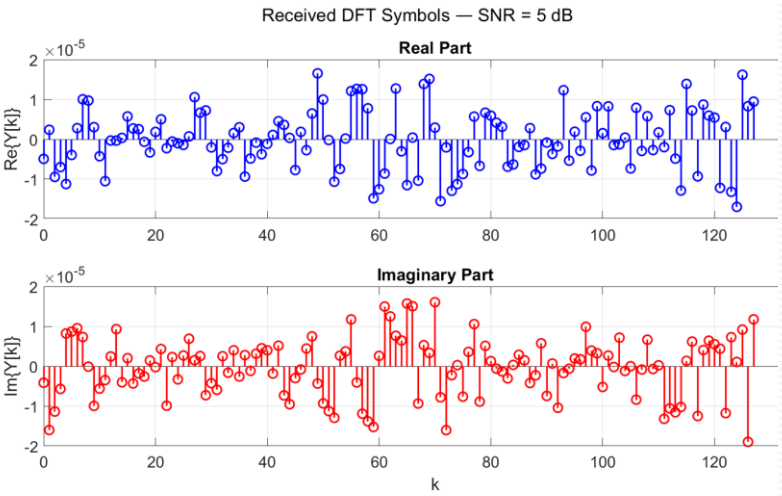
z-score to mini-batch

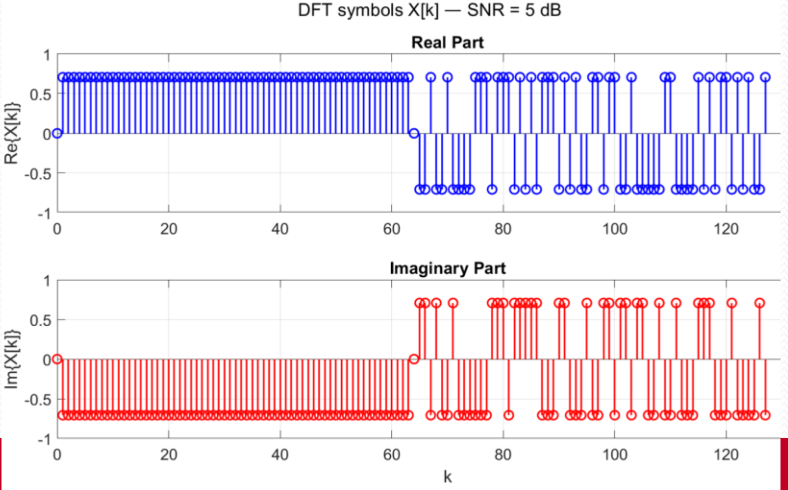
$$y^{(i)} = \gamma \cdot \hat{x}^{(i)} + \beta$$

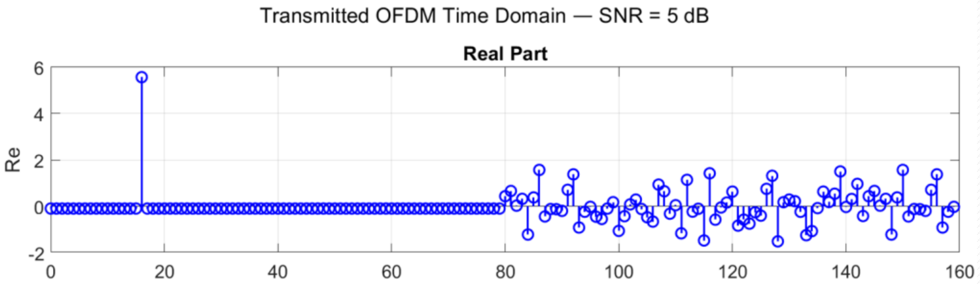
learnable scale and bias for adjusting normalization

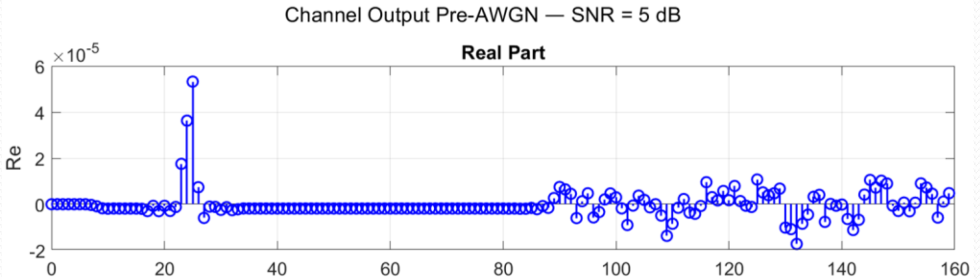


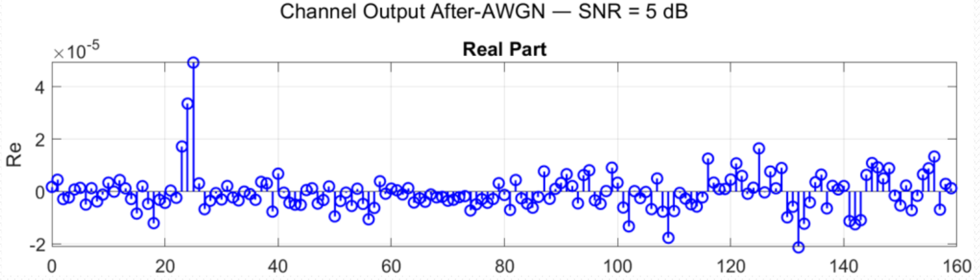
Synthetic Data Generation Example



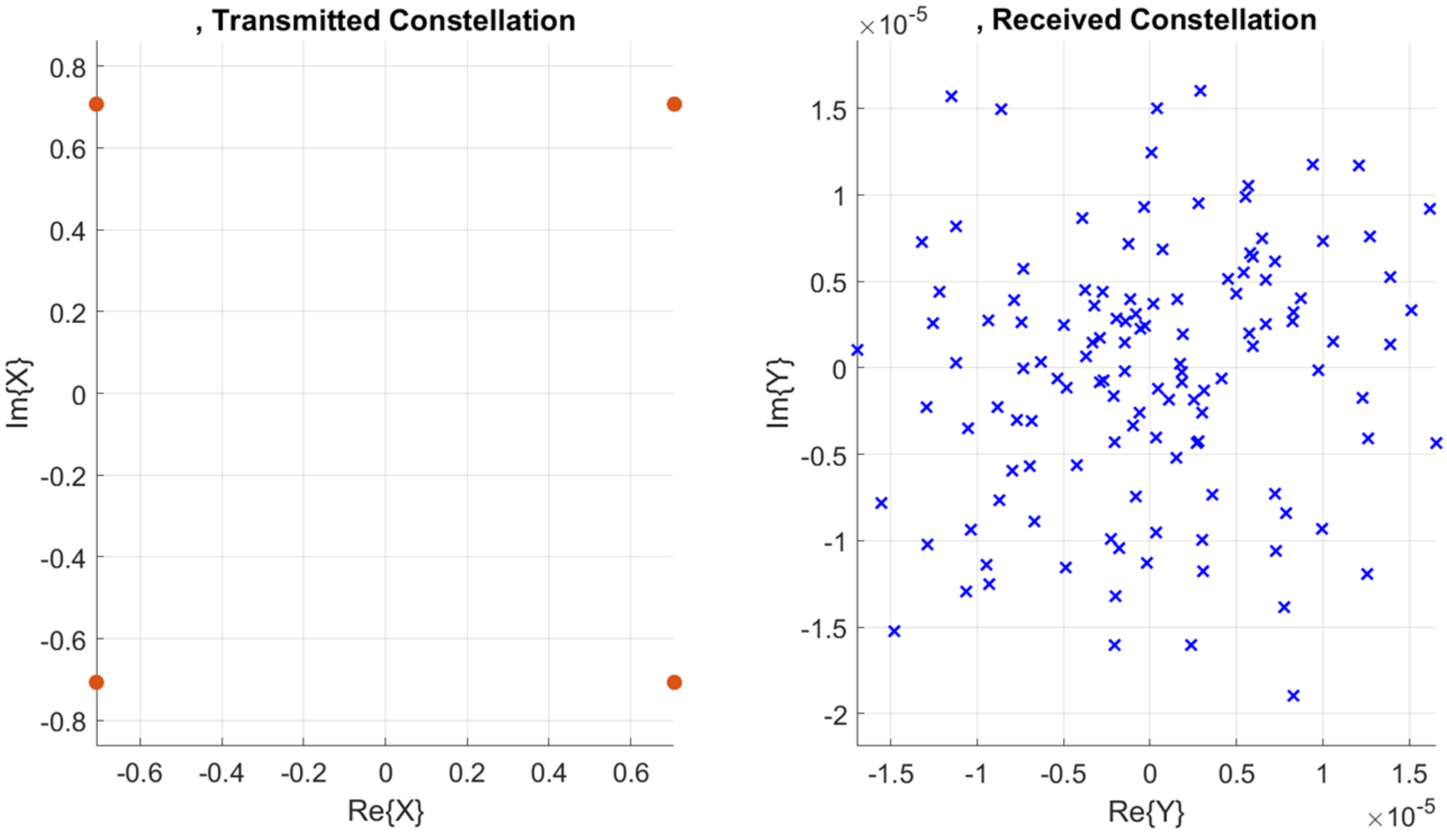




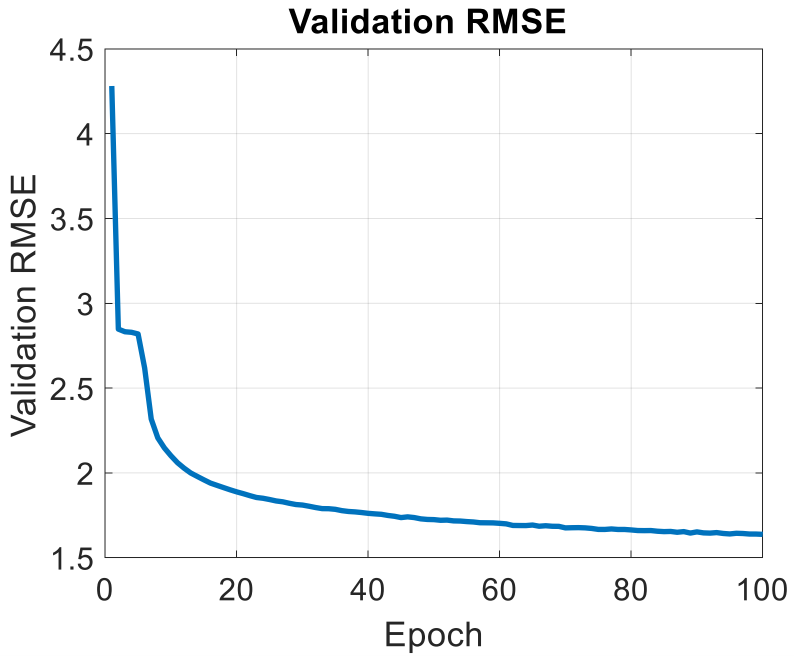




Synthetic Data Generation Example



Network Convergence Analysis



Training Loss in the paper (MSE)

$$L_2 = \frac{1}{N}\sum_k (\hat{X}(k) - X(k))^2,$$

= regressionLayer

$$\text{Validation RMSE} = \sqrt{\frac{1}{m}\sum_{j=1}^{m}(y_j - \hat{y}_j)^2}$$

Steadily decreased and converged to ~1.6

Low relative to DNN output symbol amplitude

Good generalization to unseen data

No overfitting observed: no divergence

Recovering Methods

LS Channel Estimation:

$\hat{H}_{LS} = X^{\dagger}Y$ X: Transmitted Pilot, Y: Received Pilot

ZF Recovering

$$\hat{X}_{\text{ZF}}^{(k)} = \frac{Y(k)}{\hat{H}(k)_{\text{LS}}}$$

MMSE Recovering:

Given actual channel response (h):

$$\hat{\mathbf{H}}_{\text{MMSE}} = \mathbf{R}_{HH_p}\left(\mathbf{R}_{H_pH_p}\right)^{-1}\hat{\mathbf{H}}_{\text{LS}}$$

$\hat{\mathbf{H}}_{\text{LS}}$: The initial channel estimate obtained via the Least Squares method on pilot subcarriers.

$\mathbf{R}_{HH_p}$: The cross-correlation matrix between all subcarriers and the pilot subcarriers.

$\mathbf{R}_{H_nH_n}$: The auto-correlation matrix of the pilot subcarriers.

$$\hat{X}_{\text{MMSE}}^{(k)} = \frac{Y(k)}{\hat{H}(k)_{\text{MMSE}}}$$

- **Alternative MMSE**
  From channel impulse responses obtain the 2nd order statistics and obtain MMSE estimator

$$\hat{\mathbf{h}}_{\text{MMSE}} = \boldsymbol{\mu}_h + \mathbf{R}_h\mathbf{X}^H\left(\mathbf{X}\mathbf{R}_h\mathbf{X}^H + \sigma_n^2\mathbf{I}\right)^{-1}(\mathbf{y} - \mathbf{X}\boldsymbol{\mu}_h)$$

$\mathbf{X} \in \mathbb{C}^{M\times L}$: convolution matrix

$$\tau_{\text{mean}} = \frac{\sum_k |h(k)|^2 \cdot k}{\sum_k |h(k)|^2} \quad \tau_{\text{rms}} = \sqrt{\frac{\sum_k |h(k)|^2 \cdot k^2}{\sum_k |h(k)|^2} - \tau_{\text{mean}}^2}$$

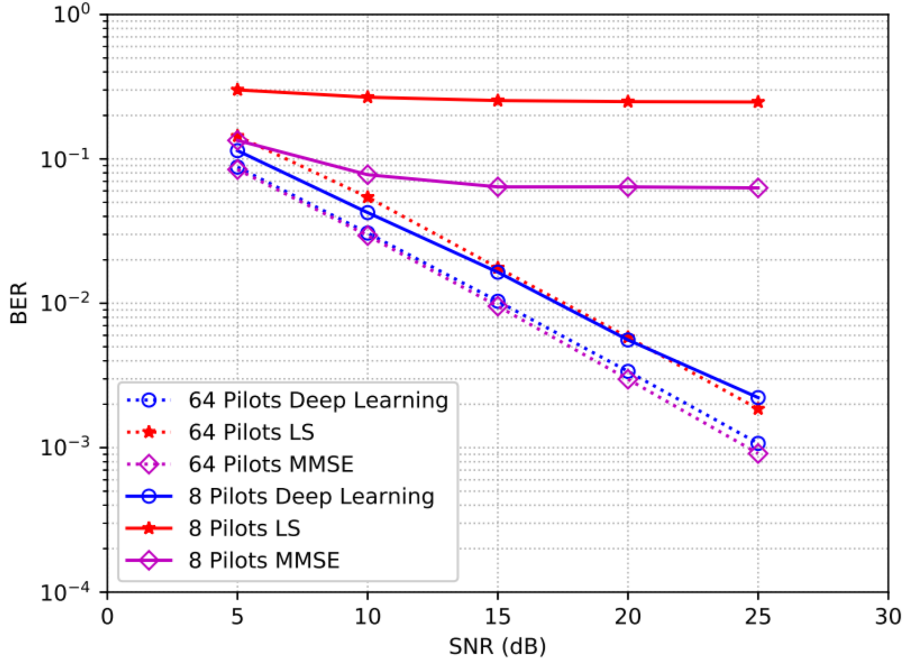
$$df = \frac{1}{N_{\text{fft}}}, \quad j2\pi\tau df = j2\pi\tau_{\text{rms}} \cdot df$$

$$[R_{hp}]_{n,m} = \frac{1}{1 + j2\pi\tau_{\text{rms}} \cdot df \cdot (n - mN_{ps})}$$

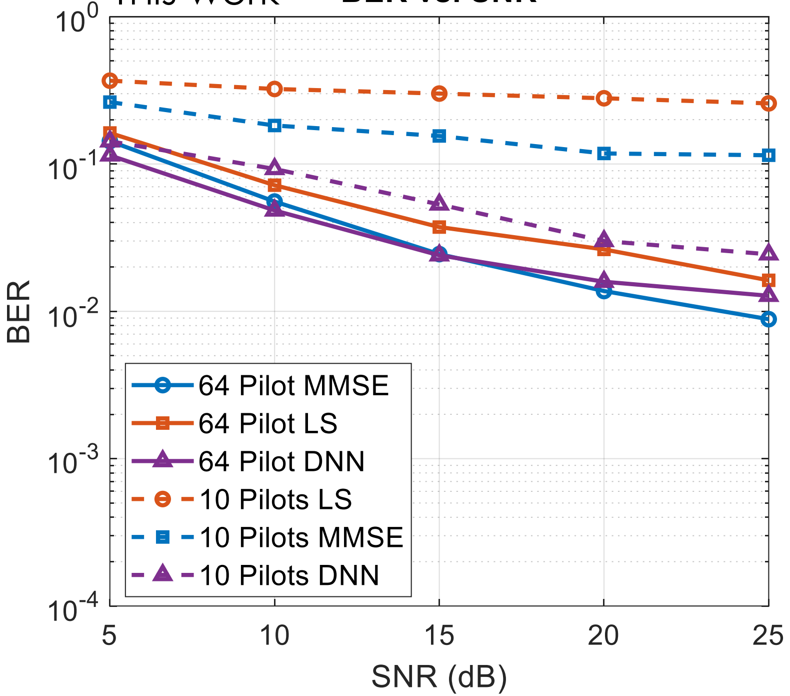
$$[R_{pp}]_{m_1,m_2} = \frac{1}{1 + j2\pi\tau_{\text{rms}} \cdot df \cdot N_{ps}(m_1 - m_2} + \frac{1}{\text{SNR}_{\text{lin}}}$$

The Effect of Pilot Number

Reference work



This work **BER vs. SNR**

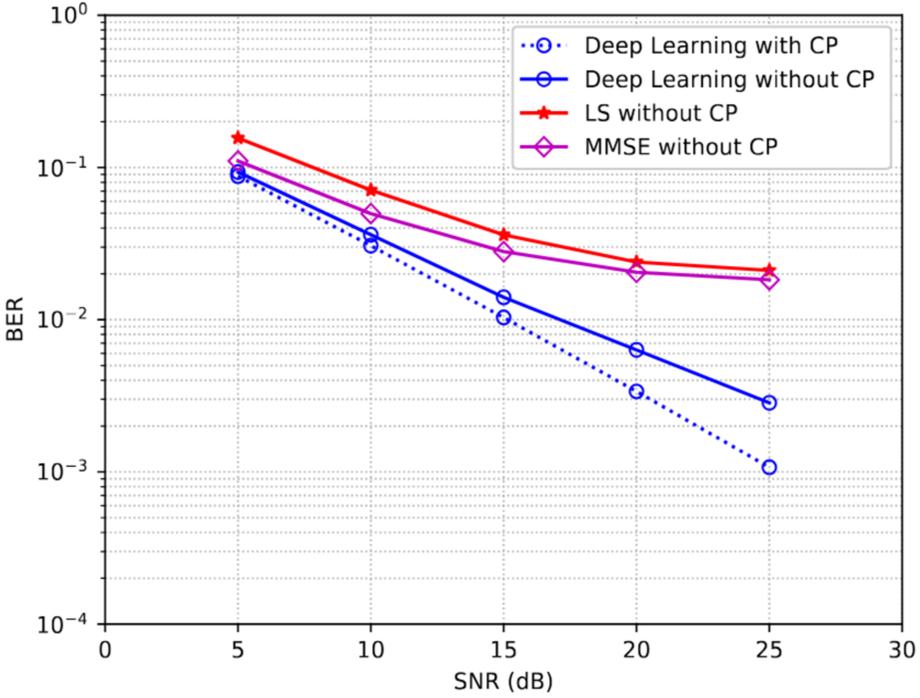


64 pilot, MMSE achieves the **best** (prior statistics utilized) performance, while DNN performs **better** than LS (no prior statistics) and is close to MMSE.

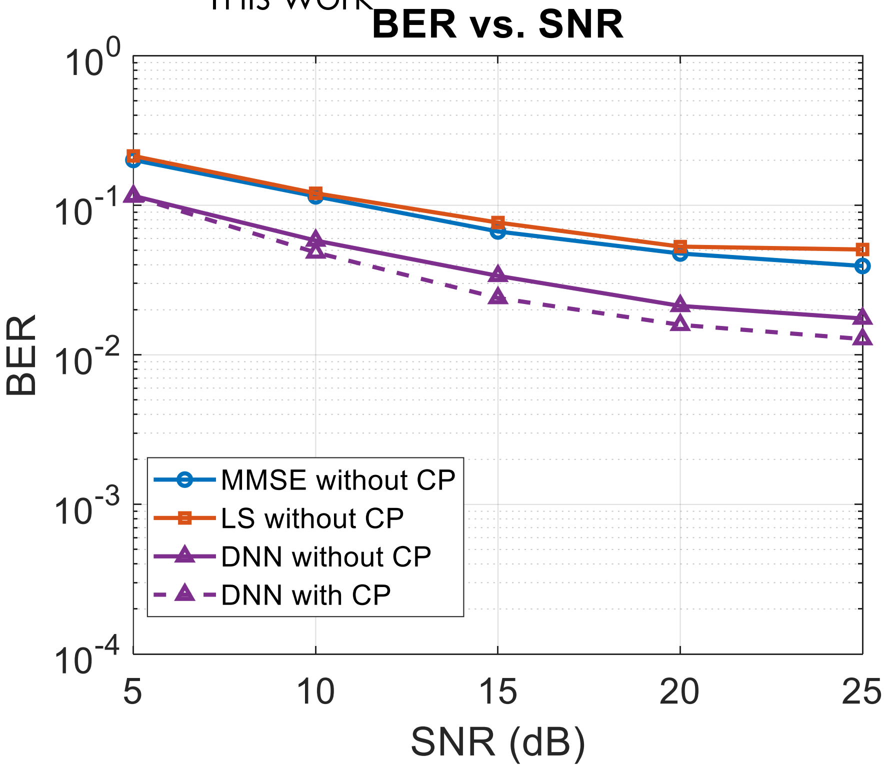
10 pilot, DNN **surpasses** LS and MMSE which saturate, maintaining lower BER as SNR increases, shows **higher** robustness.

The Effect of Cyclic Prefix

Reference work



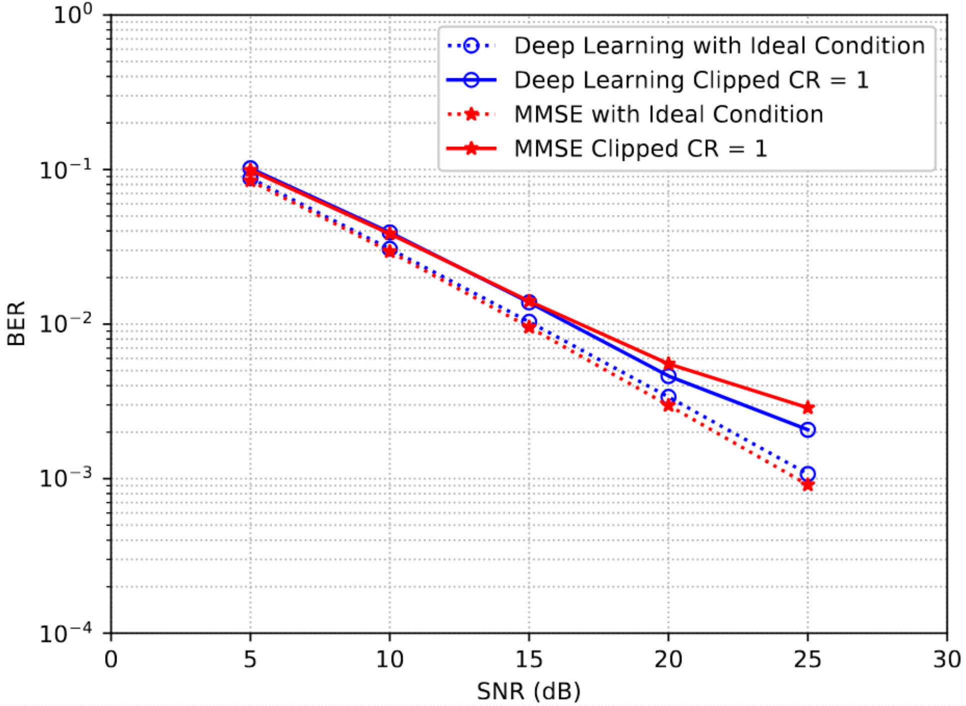
This work

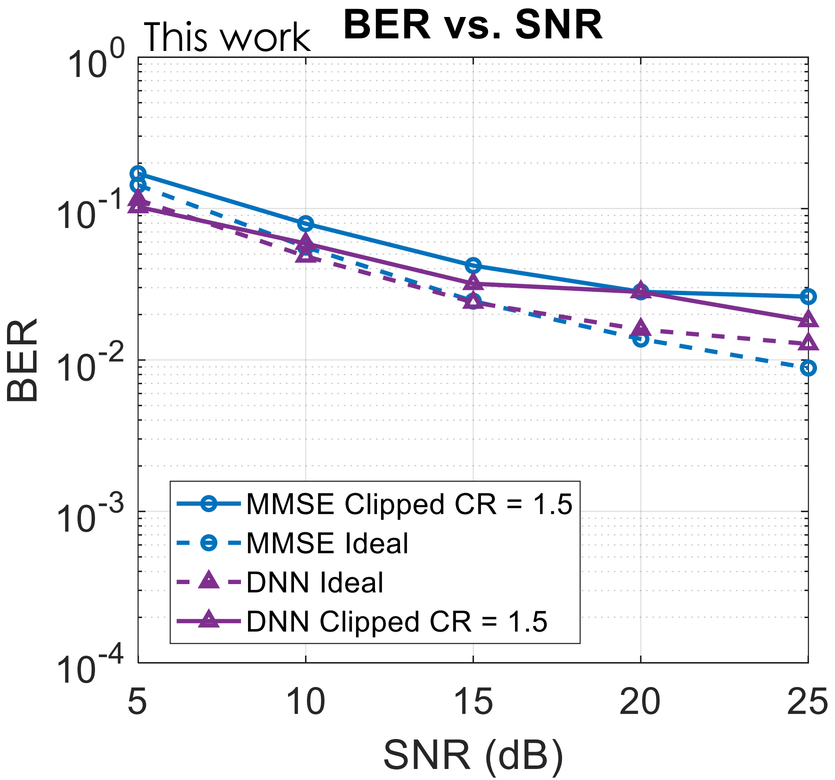


Without CP, MMSE and LS perform **worse** and saturate after 15 dB, while DNN remains robust and performs **better**.

The Effect of Clipping Noise

Reference work

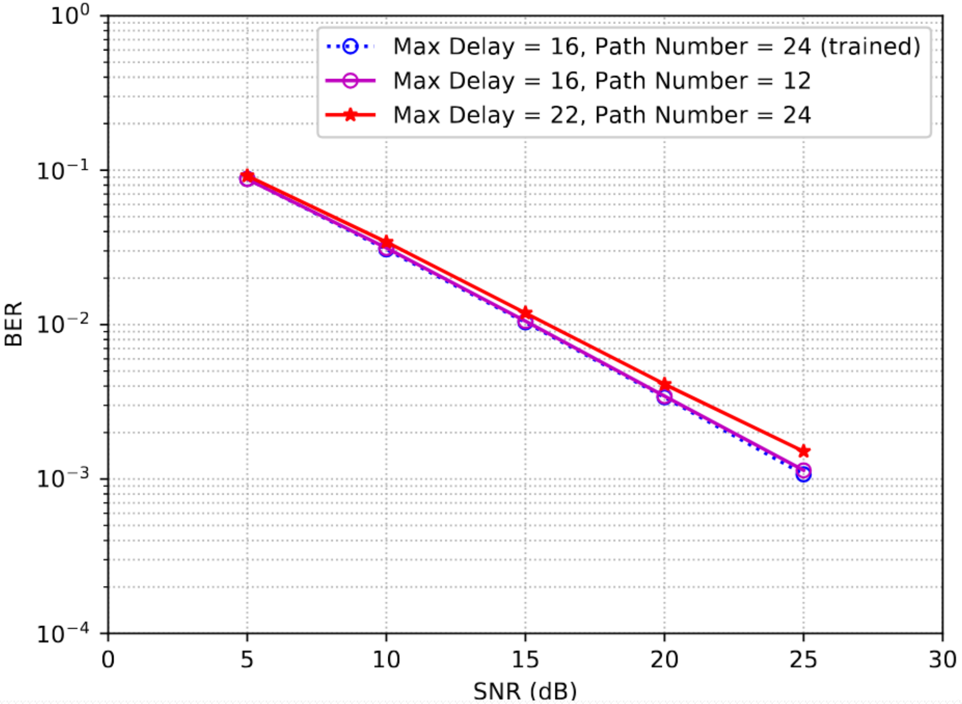




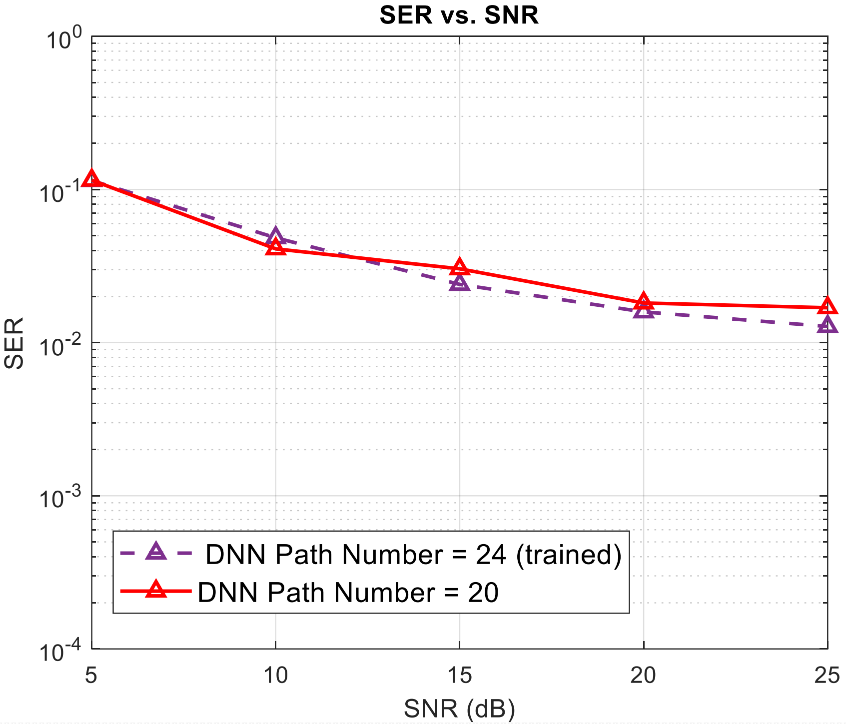
For CR=1.5, MMSE performs **better without** clipping, but DNN **outperforms** MMSE with clipping for **SNR>20 dB**, showing higher robustness.

Mismatched Training and Test

Reference work



This work



Trained with 24 paths, tested with 20 paths.

DNN's symbol detection performance is **not significantly damaged** by variations on the statistics of channel.

Conclusion

DNN-based receiver was implemented for joint channel estimation and symbol detection in OFDM systems.
The model was trained on synthetic data using the WINNER II channel and evaluated under various SNR levels.

Key Results:

- **Low Pilot Count:** DNN achieved the lowest BER and outperformed LS and MMSE.
- **No Cyclic Prefix:** DNN remained effective, while LS and MMSE saturated after 15 dB.
- **Clipping Noise (CR = 1.5):** DNN outperformed MMSE at high SNR values.
- **Channel Mismatch:** DNN maintained stable performance despite differences in path count.

Overall, the DNN model demonstrated strong robustness and generalization across all tested scenarios.

References

[1] H. Ye, G. Y. Li, and B.-H. Juang, “Power of deep learning for channel estimation and signal detection in OFDM systems,” IEEE Wireless Communications Letters, vol. 7, no. 1, pp. 114–117, 2017.

[2] P. Ky¨osti, “WINNER II channel models,” IST, Tech. Rep. IST-4-027756 WINNER II D1.1.2 V1.2, 2007

Thank you for your attention.