# Running Steps:

- Run the IDX_Generation.m in order to genertae the dataset indices, training dataset size, and testing dataset size.
- Run the main.m file two times as follows:
  1. Specify all the simulation parameters like: the number of OFDM symbols, channel model, mobility scenario, modulation order, SNR range, etc.
  2. Specify the path of the generated indices in step (1).
  3. The first time for generating the training simulation file (set the configuration = 'training' in the code).
  4. The second time for generating the testing simulations files (set the configuration = 'testing' in the code).
  5. After that, the generated simulations files will be saved in your working directory.
- Run the LSTM_Datasets_Generation.m also two times by changing the configuration as done in step (2) in addition to specifying the channel estimation scheme as well as the OFDM simulation parameters. This step generates the LSTM training/testing datasets.
- Run the LSTM.py file also two times in order to perform the training first then the testing as mentioned in the LSTM.py file description.
- After finishing the previous step, the LSTM results will be saved as a .mat files. Then you need to run the LSTM_Results_Processing.m file in order to get the NMSE and BER results of the studied channel estimation scheme.
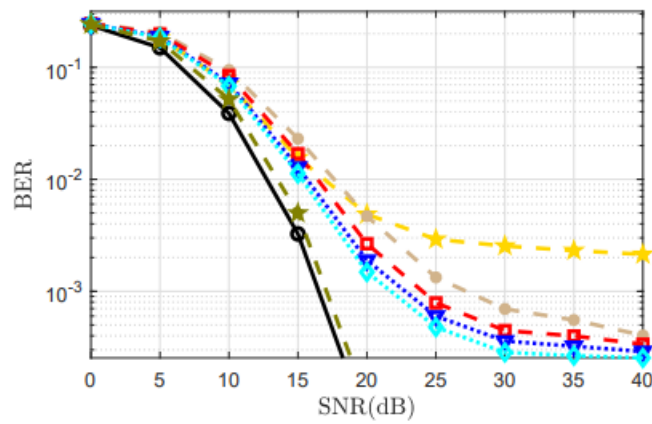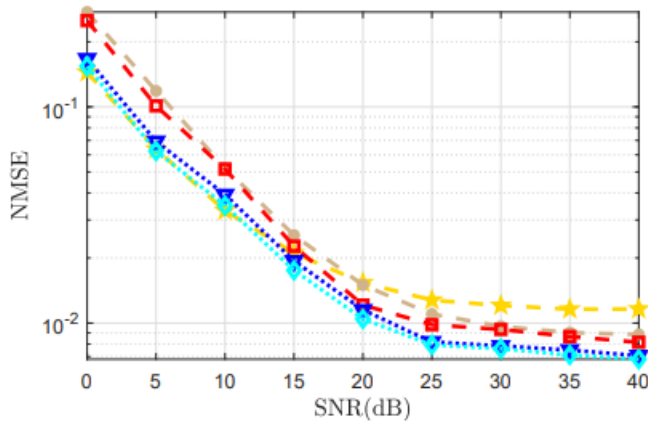
# Results:
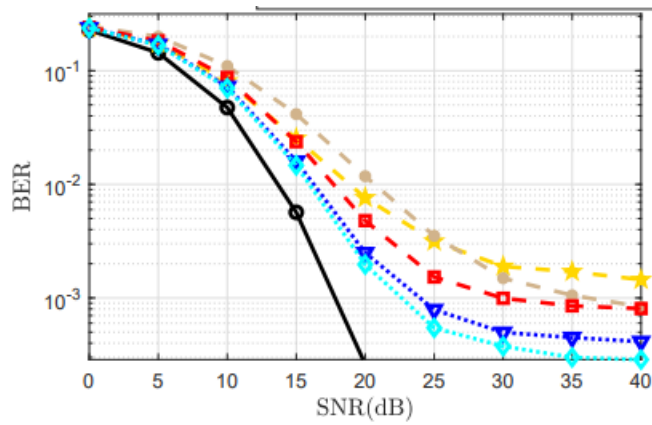
BER performance employing high mobility scenario
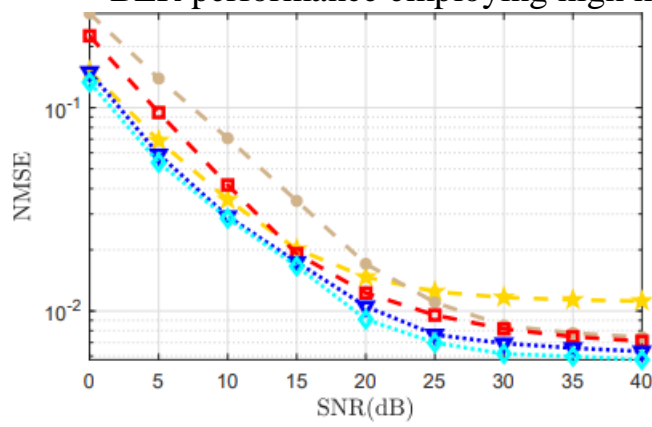


NMSE performance employing high mobility scenario



BER performance employing very high mobility scenario

NMSE performance employing very high mobility scenario

- Data-pilot aided-(DPA)
- Temporal averaging (TA)
- Spectral temporal averaging-(STA)
- Time domain reliable test frequency domain interpolation-(TRFI)

## REFERENCE:

[1] A. K. Gizzini, M. Chafii, S. Ehsanfar and R. M. Shubair, "Temporal Averaging LSTM-based Channel Estimation Scheme for IEEE 802.11p Standard," 2021 *IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 01-07, doi: 10.1109/GLOBECOM46510.2021.9685409.