SIMULATION MATLAB CODE for PMI-Based Codebook Enhancement

Advanced Codebook Generation Using High-Resolution k-Means Clustering to PMI Data

VERSION 1.2

2025.02.20

Ph.D. Sangchun Park

Sogang University, Republic of Korea

Required Applications

Main Application

- Application Name: MATLAB

- Version: R2023a (Other versions are also highly likely to be compatible.)

- Description: A high-level language and interactive environment for numerical computation, visualization, and programming, widely used for data analysis, algorithm development, and modeling.

- Download Link: https://mathworks.com/products/matlab.html

Required MATLAB Toolbox

- Toolbox Name: Statistics and Machine Learning Toolbox

- Description: Provides functions and tools for data analysis, statistical modeling, machine learning, and predictive analytics. (In this simulator, the function "ksdensity" is used for kernel density estimation.)

- Download Link: https://mathworks.com/products/statistics.html

Version History

[2024-07-12] Version 1.0

- Initial release of the code.

[2024-07-15] Version 1.1

- Updated the code for generating figures 6 and 7.

[2025-02-20] Version 1.2

- Refactored the code for improved structure and usability.

Overview

The paper **"Advanced Codebook Generation Using High-Resolution k-Means Clustering to PMI Data"** presents a novel method for generating advanced codebooks for beamforming in 5G and beyond wireless systems. The proposed approach is based solely on **Precoding Matrix Indicator (PMI) feedback** from user equipment (UE) and does not require additional signaling or channel state information (CSI). The method enhances the resolution of PMI data using **Kernel Density Estimation (KDE)** and refines the beamforming codebook through $\boldsymbol{k}$**-Means clustering**. By iteratively updating the codebook, the system adapts autonomously to changing wireless environments, improving beamforming performance while remaining compliant with **3GPP standards**.

Working Mechanism

1. Standard Codebook and PMI Collection:

- The base station (BS) transmits reference signals using an initial DFT-based codebook.
- User equipment (UE) selects the best beam vector based on received signal strength and reports a PMI index.

2. Kernel Density Estimation (KDE) for PMI Enhancement:

- The PMI feedback data is discrete and sparse.
- KDE is applied to smooth and generate a higher-resolution representation of the probability distribution of PMI values.

3. High-Resolution k-Means Clustering for Codebook Refinement:

- Using the refined PMI distribution, a spherical k-Means++ clustering algorithm is applied.
- The clustering process extracts Q optimal beam vectors, forming a new and environment-optimized codebook.

4. Iterative Codebook Updating:

- The refined codebook is applied in the next round of PMI feedback collection.
- This iterative process continues, further optimizing beamforming vectors without the need for large datasets or deep learning models.

5. Performance Evaluation:

- The proposed method improves beamforming gain and spectral efficiency, achieving a higher sum-rate than 3GPP DFT-based codebooks.
- It dynamically adapts to changing channel conditions, ensuring stable performance even in high-mobility scenarios.
- Additionally, it reduces computational overhead while maintaining competitive accuracy compared to deep learning-based approaches.

Contents

- **Folders/files**

- **Structures Diagram**

- **Code Explanation** [Demo.m, Figure_3.m, Figure_6and7.m]

Folders and files

Folder

File

PMI-based Codebook Enhancement Github Code

generateChannel.m – Generate channels based on Saleh-Valenzuela channel model

enhanceCodebook.m – Generate enhanced codebook using the proposed codebook enhancement method

generateDFT.m – Generate DFT codebooks

generateVQ.m – Generate perfect location information-based codebooks

Sum_rate.m – Evaluate sum-rate performance using codebooks

rank_adaptive_sum_rate.p – Evaluate sum-rate using the rank adaptive scheme

KDE_setup.m – Set up for kernel density estimation (KDE)

Plot1.m
Plot2.m
Plot3.m
} Plot the process of proposed codebook enhancement

location_xs.mat – UE locations generated by SUMO

Demo.m – Demo code for explanation this simulator.

Figure3.m – Draw figure 3 in the paper

function

- The files in function folder are used in simulations
- File list

ARV_UPA.p	sph2xyz.p
Codebook_DFT.p	SV_channel.p
Codebook_VQ.	xyz2sph.p
inverse_rotation.p	
kmeans_plus_clustering.p	
laprnd.p	
PMI2sph.p	
rotation.p	

channel

- This folder is generated and populated by generateChannel.m file.

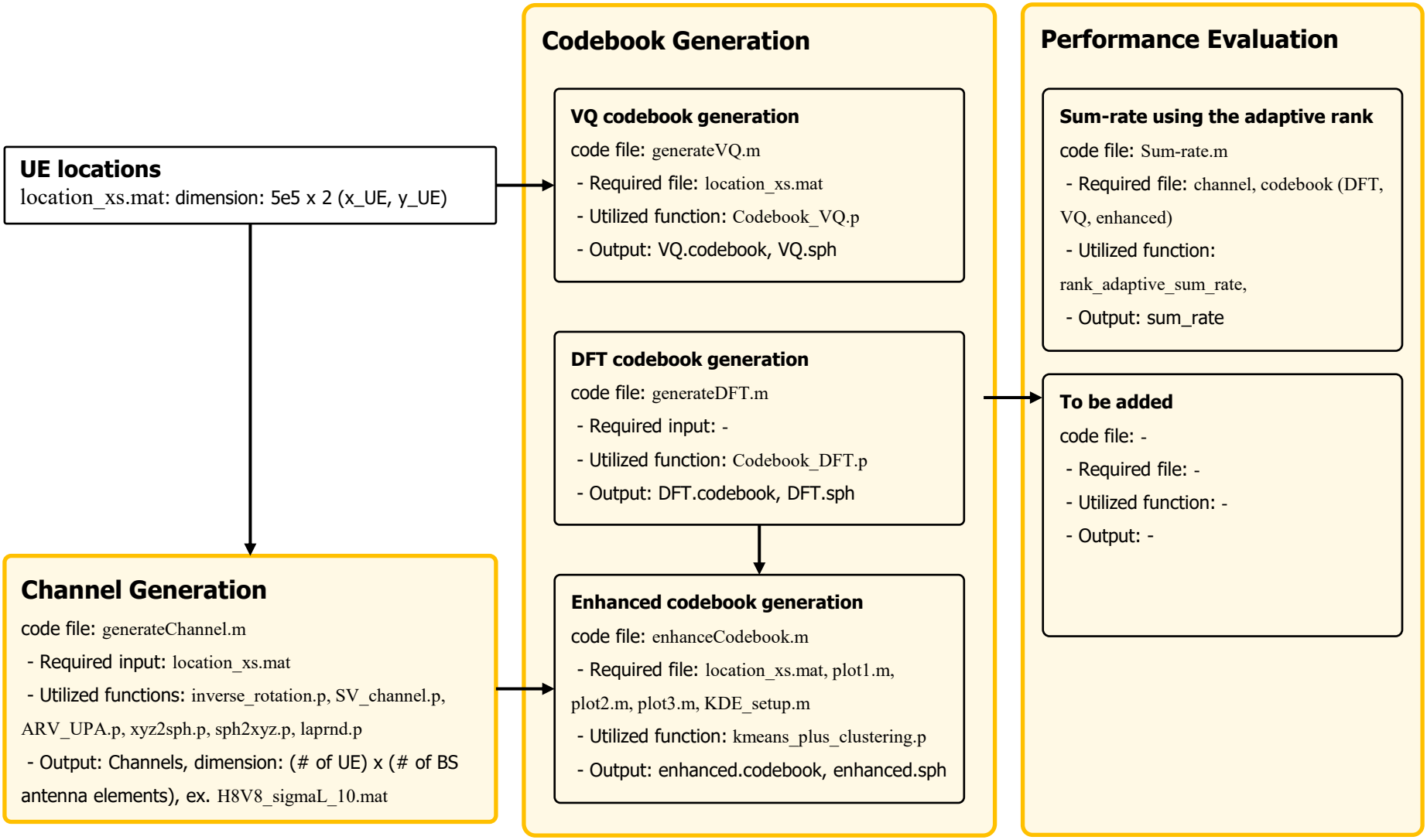
codebook

- This folder is generated and populated by enhanceCodebook.m, generateDFT.m, and generateVQ.m files.

sum_rate

- This folder is generated and populated by Sum_rate.m file.

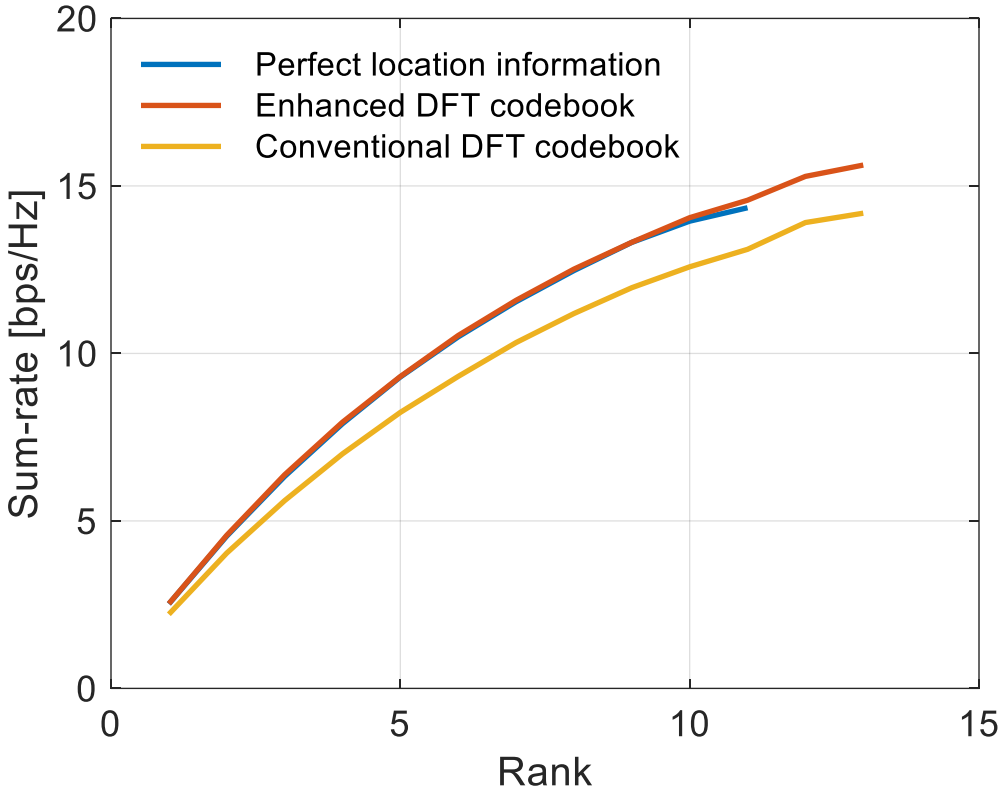
Structure Diagram



Code Explanation [Demo.m]

- Execute all parts of the structure diagram to plot the sum-rate results.

- Output figure:

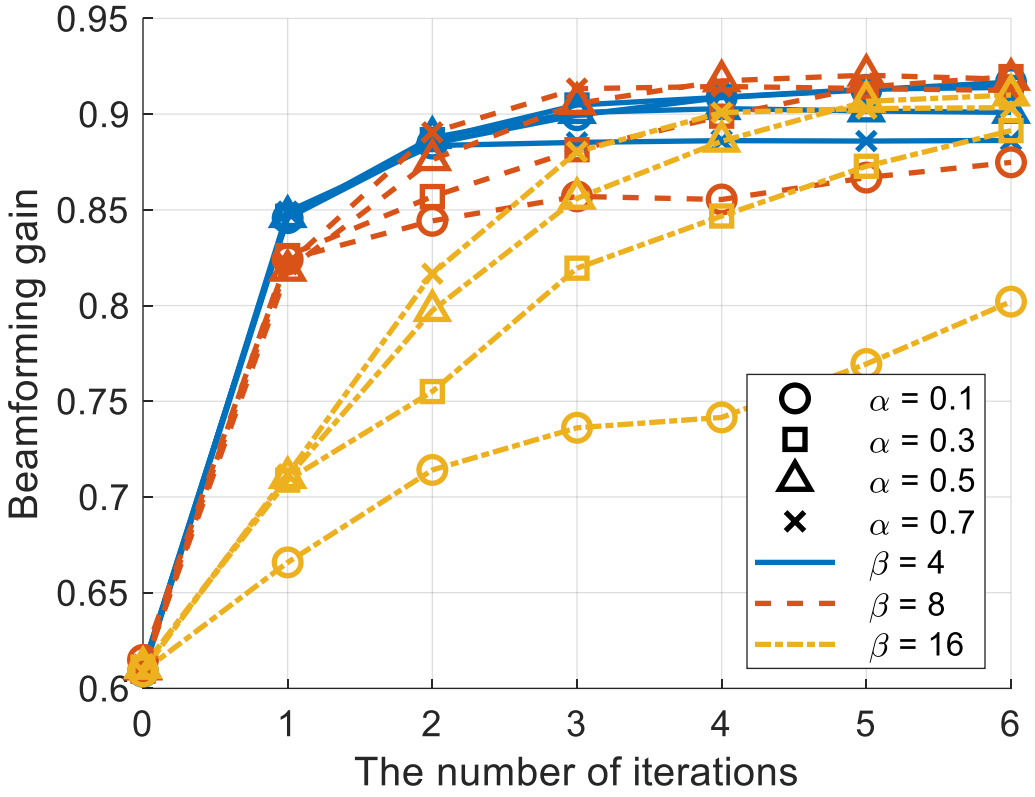


Code Explanation [Figure_3.m]

[CPU: i5-11400F, RAM: 32GB]
Required time: 1870 sec
Required memory: 3GB

- Plot the figure 3 in the paper

- Output figure:



Code Explanation [Figure_6and7.m]

[CPU: i5-11400F, RAM: 32GB]
Required time: 680 sec
Required memory: 3GB

- Plot the figure 6 and figure 7 in the paper

- Output figure: