Adaptive Digital Modulation Using ML

Dr. JAISOORAJ
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
j_jaisooraj@cb.amrita.edu

Dr. DIVYA SANDEEP
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
s_divya1@cb.amrita.edu

Dr. ASHIK SURESH
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
s_aashiq@cb.amrita.edu

V. SAI VED
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
Cb.sc.u4aie23262@cb.students.amrita.edu

HARI VAARTHAN T.D
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
Cb.sc.u4aie23228@cb.students.amrita.edu

K.V.NIHAL REDDY
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
Cb.sc.u4aie23239@cb.students.amrita.edu

U. ADITYA SANDEEP
Amrita Vishwa Vidyapeetham
Coimbatore , Tamil Nadu
Cb.sc.u4aie23257@cb.students.amrita.edu

***Abstract*— The major contribution of this project is to propose a new real-time system for the classification of BPSK and QPSK digital modulation schemes in Wireless communication systems. This system makes use of GNU Radio, an open-source software toolkit for software-defined radio (SDR), combined with the power of machine learning. This methodology includes processing complex valued wireless signals in both clean and noisy environments. Statistical characteristics important in performing modulation classification are extracted from these signals, such as SNR (Signal to Noise Ratio), Bit Error Rate (BER) Phase Spread. These features are used to train and evaluate a pre-trained K-Nearest Neighbors (KNN) classifier. The system is integrated into GNU Radio for real-time signal acquisition, processing, and classification. This tight coupling allows for dynamic evaluation using live RF signals.**

***IndexTerms---* Digital Modulation, Machine Learning, GNU Radio, KNN Classifier, Signal Processing, SDR**

I) INTRODUCTION

Modern wireless systems demand high data rates and reliable transmission, especially with the growth of IoT and 5G technologies. Digital modulation schemes like BPSK and QPSK play a key role in balancing robustness and efficiency. BPSK is more noise resistant, while QPSK offers better bandwidth efficiency However, traditional systems use fixed modulation schemes, leading to performance issues when channel conditions change. Using QPSK in noisy environments increases errors, and using BPSK in clean conditions wastes bandwidth. Current communication systems lack real time adaptability in choosing the modulation scheme based on signal quality. There's a need for a smart, software-based solution that can dynamically switch between BPSK and QPSK depending on live signal values like SNR, BER, and Spread.

This project solves this by implementing an adaptive modulation system using Machine Learning in GNU Radio, which extracts real-time features and uses a K-Nearest Neighbours classifier to switch modulation schemes in real time improving efficiency and reliability.

The primary objective of this project is to develop an adaptive digital modulation system capable of dynamically switching between BPSK and QPSK based on real time conditions using machine learning. This system aims to enhance communication efficiency by automatically selecting the best modulation scheme according to key signal parameters such as Signal to Noise Ratio (SNR), Bit Error Rate (BER), and Phase Spread. The project also seeks to implement a complete software only solution using GNU Radio, without relying on any external hardware, thus making it highly accessible. Additionally, the project focuses on combining a trained machine learning model into the GNU Radio flowchart to enable seamless and intelligent modulation switching in real time. Finally, the goal is to improve the reliability and bandwidth utilization of wireless communication systems in varying environmental conditions.

Machine Learning plays a main role in this project by enabling the system to make intelligent decisions in real time. Traditional modulation schemes in communication systems are static, which leads to inefficient performance under fluctuating signal conditions. By integrating ML, the system can learn from signal characteristics such as SNR, BER, and phase spread, and classify the optimal modulation type dynamically. This not only improves adaptability but also ensures efficient use of bandwidth and better transmission reliability.

The Internet of Things (IoT) connects various devices and systems, facilitating real time monitoring and control. In this project, the concept of IoT is reflected through the use of a software defined radio environment where signal transmission and adaptation occur in real time. By simulating a wireless communication setup, the project explains how smart devices can dynamically adapt to network conditions without human intervention ideal for modern IoT deployments where reliability and autonomy are key.

II) DATASETS

The dataset used in this project is for training and evaluating the machine learning model that drives the modulation switching mechanism. It consists of 1800 samples, each characterized by three key features: Signal to Noise Ratio (SNR), Bit Error Rate (BER), and Phase Spread, along with a corresponding modulation label (either BPSK or QPSK). These features were extracted in real time using a custom built Feature Extraction block in GNU Radio, which receives both clean and noisy versions of the transmitted signal. The dataset effectively captures the variation in signal conditions under different noise scenarios, which is crucial for teaching the ML model how to distinguish and adapt between modulation schemes. It represents a balanced and practical set of real world like transmission conditions, making it ideal for training classifiers such as K-Nearest Neighbors for accurate and reliable modulation prediction in live wireless environments.

III)METHODOLOGY

A) Feature Extraction

This part is an important part of the system where the raw signals are converted into meaningful numerical representations that the machine learning model can understand. In this project, both clean and noisy signals are generated using BPSK and QPSK modulation schemes in GNU Radio. These signals are then given into a Feature Extraction block that processes each signals to calculate key parameters such as Signal to Noise Ratio (SNR), Bit Error Rate (BER), and Spread (a measure of phase variance). These three features capture the essential characteristics of the signal's quality and behaviour under noise. The extracted values are written into a CSV file for training purposes and also passed forward in real time for live modulation classification. This step enables the machine learning model to make decisions based on signal conditions, making it foundational to the adaptive modulation system.

B) Model Training

Once the dataset of features (SNR, BER, Spread) and their corresponding modulation labels (BPSK or QPSK) is prepared, the model training part begins. Using Python and scikit-learn, the dataset is first cleaned to ensure robustness during training. The features are then standardized using scaling techniques, and the categorical labels are encoded into numerical format for compatibility with the machine learning algorithms. Multiple models Logistic Regression, K-Nearest Neighbours (KNN), and Multi-Layer Perceptron (MLP) Classifier are trained and evaluated based on their accuracy and generalization performance. Among these, the KNN model has the best classification accuracy and was selected for deployment. This model was then converted and saved in .pkl format for integration into GNU Radio. The training process ensures that the system can accurately recognize signal patterns and make correct decisions during live execution.

C) *Modulation Prediction and Switching*

The final and most important part of the system is modulation prediction and switching, which is implemented in real time within GNU Radio using the trained KNN model. In this part, the ML Modulation Switch block loads the saved .pkl model and continuously receives real time feature vectors (SNR, BER, Spread) from the Feature Extraction block. These incoming features are scaled and then passed to the model, which predicts whether the current signal conditions are better suited for BPSK or QPSK modulation. Based on the prediction, the system writes the result in a file. This dynamic switching ensures that the modulation type can adapt in realtime to optimize performance, making the communication system more intelligent and useful for varying channel conditions. It reflects how machine learning and IoT concepts come together to build a smarter, self-correcting communication system.

D) ML models

In this project, three different machine learning models were explored and evaluated for the task of modulation classification: Logistic Regression, K-Nearest Neighbours (KNN), and Multi-Layer Perceptron (MLP) Classifier. Each model was chosen for its distinct learning approach and capability to handle small, feature-based datasets efficiently

- Logistic Regression is a linear classification algorithm that estimates the probability of a signal belonging to a modulation class based on the input features. It used as a reliable baseline due to its simplicity and fast training time, performing decently on linearly separable data.
- K-Nearest Neighbors (KNN) is a non-parametric model that classifies a signal by analyzing the majority class of its nearest data points in the feature space. In this project, KNN delivered the highest classification accuracy, as it effectively leveraged the spatial closeness of feature vectors (SNR, BER, and Spread) to separate BPSK and QPSK classes.
- Multi-Layer Perceptron (MLP) is a type of feedforward artificial neural network that learns complex non-linear patterns. It was included to explore the potential of deep learning techniques on the dataset. The MLP model achieved good performance, especially in capturing non-linear boundaries between classes, thanks to its multiple hidden layers and adaptive learning.

After comparing the models using accuracy scores, classification reports, and cross-validation, KNN was selected as the final model for deployment in the GNU Radio flow. Its simplicity, interpretability, and strong performance made it the ideal choice for real-time signal classification in this adaptive modulation system.

E) Dataflow

The dataflow of this project begins with the Random Source block in GNU Radio, which generates a stream of random binary bits simulating real world data. These bits are passed to the Modulator block, where they are modulated using either BPSK or QPSK, based on the selected modulation scheme. This modulated signal is then duplicated one copy remains as the clean signal, while the other is passed through a Noise block to simulate real world channel conditions, resulting in a noisy signal. Both the clean and noisy signals are then forwarded to the Feature Extraction block. This custom block calculates three key signal features:

- Signal-to-Noise Ratio (SNR) – measuring signal quality.
- Bit Error Rate (BER) – indicating the rate of bit mismatches between clean and noisy signals.
- Phase Spread – affects the phase variability, especially relevant for QPSK.

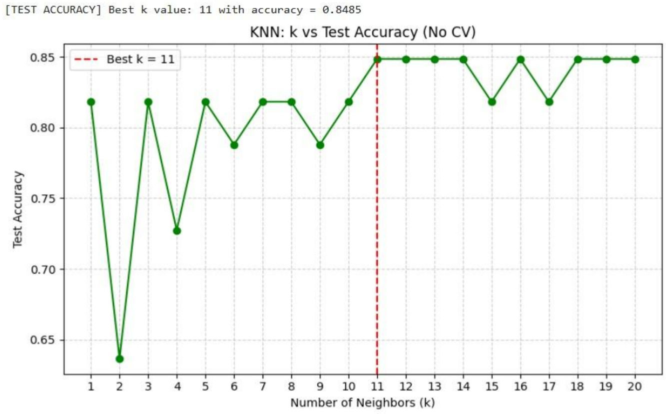
These features are not only written into a CSV file for offline training but also streamed in real time to the ML Modulation Switch block. This block loads a trained K-Nearest Neighbors (KNN) model along with a Standard Scaler and Label Encoder to classify the modulation type either BPSK or QPSK based on live signal conditions. The predicted modulation label is then written and used to decide the next modulation type for transmission, effectively enabling adaptive modulation.

IV)RESULTS AND CONCLUSION

```
Logistic Regression Accuracy: 0.8788
Best Parameters: {'C': 0.1, 'class_weight': None, 'dual': False,
Classification Report:
              precision    recall  f1-score   support

           0       0.75      0.75      0.75         8
           1       0.92      0.92      0.92        25

    accuracy                           0.88        33
   macro avg       0.83      0.83      0.83        33
weighted avg       0.88      0.88      0.88        33
```

```
K-Nearest Neighbors Accuracy: 0.8485
Best Parameters: {'algorithm': 'auto', 'leaf_size': 30, 'metric'
Classification Report:
              precision    recall  f1-score   support

           0       0.64      0.88      0.74         8
           1       0.95      0.84      0.89        25

    accuracy                           0.85        33
   macro avg       0.80      0.86      0.82        33
weighted avg       0.88      0.85      0.86        33
```

```
MLP Classifier Accuracy: 0.7879
Best Parameters: {'activation': 'relu', 'alpha': 0.001, 'batch_size'
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.12      0.22         8
           1       0.78      1.00      0.88        25

    accuracy                           0.79        33
   macro avg       0.89      0.56      0.55        33
weighted avg       0.83      0.79      0.72        33
```



V)REFERENCES

- 1] T. J. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, Dec. 2017.
- [2] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, Sep. 2018.
- [3] M. Afgani, H. Haas, and A. T. Koc, "Adaptive modulation for wireless communications in varying environments," in *Proc. Int. Conf. Wireless Communications and Mobile Computing*, Vancouver, Canada, Jul. 2006, pp. 1327–1332.
- [4] R. Srinivasan and M. R. Sarma, "Modulation classification using machine learning," in *Proc. Int. Conf. Signal Processing and Communications (SPCOM)*, Bengaluru, India, Jun. 2016, pp. 1–5.
- [5] O. Gnawali, "Software-defined radio using GNU Radio and USRP," Dept. of Computer Science, University of Houston, Houston, TX, USA, Tech. Rep., 2010. [Online]. Available: https://gnuradio.org
- [6] S. Lalitha and T. Gireeshkumar, "Automatic modulation classification using KNN algorithm," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 1, pp. 217–221, May 2019.
- [7] S. Mathur, R. Patidar, and A. Sharma, "Real-time modulation recognition using SDR and machine learning," in *Proc. IEEE Ubiquitous Computing, Electronics & Mobile Communication Conf. (UEMCON)*, New York, USA, Oct. 2020, pp. 80–85.
- [8] GNU Radio Project, "GNU Radio Documentation," [Online]. Available: https://wiki.gnuradio.org/index.php/Main_Page