



ZJU-UIUC INSTITUTE

Zhejiang University-University of Illinois at Urbana-Champaign Institute

浙江大学伊利诺伊大学厄巴纳香槟校区联合学院

ECE 459

COMMUNICATIONS SYSTEMS

FINAL PROJECT

PERFORMANCE ANALYSIS OF AMPLITUDE AND FREQUENCY MODULATION IN NOISE

Group 5

Name	Grade
Yiqin LI	
Peidong YANG	
Zhuohao XU	
Rongjian CHEN	
Tiantian ZHONG	

Abstract

Keywords Keyword 1, keyword 2, keyword 3

Contents

1	Introduction	1
2	Methodology	2
3	Results and Discussion	3
4	Conclusion	4
	Appendix A Python Scripts of This Project	6

1 Introduction

This task involves building an Amplitude Modulation (AM) system using Python (or another programming language) and performing a series of simulations and analyses.

The specific steps include:

1. **Simulating Envelope Modulation with a Sinusoidal Wave and a Modulation Index of 0.3:** First, we need to create a sinusoidal wave to act as the message signal and use this signal to modulate a carrier signal. The modulation index of 0.3 refers to the extent to which the message signal modulates the carrier signal. The goal here is to generate an AM signal whose envelope reflects the shape of the message signal.
2. **Including AWGN Noise:** Additive White Gaussian Noise (AWGN) is introduced into the modulated signal to simulate noise interference common in real communication environments. This makes the signal more representative of actual communication signals and tests the robustness of signal detection.
3. **Performing Envelope Detection:** Envelope detection is the process of recovering the original message signal from the modulated carrier signal. In this step, we will process the noisy AM signal to extract its envelope, representing the original message signal.

The specific tasks include:

1. **Plotting the Envelope Modulated Signal:** Show the waveform of the signal after envelope modulation in the time domain.
2. **Plotting Its Spectrum:** Create a spectrum plot of the modulated signal to display its frequency domain distribution.
3. **Plotting the Envelope-Detected Signal Before Low-Pass Filtering:** Display the waveform of the signal obtained through envelope detection before applying low-pass filtering to remove noise.
4. **Comparing Post-Detection Signal-to-Noise Ratio (SNR) to Theoretical Values:** Calculate the post-detection SNR for both low and high pre-detection SNR scenarios and compare these values to theoretical predictions. This comparison helps understand the performance of the signal detection algorithm under different noise levels.

2 Methodology

3 Results and Discussion

4 Conclusion

Statement of Contribution

Appendix A Python Scripts of This Project