BLG 354E - Signal & Systems for Computer Engineering

Project-1 CRN: 21350

Student Information:

Name: Yusuf YILDIZ

Student ID: 150210006

Introduction:

This report documents the process and results of Project-1 conducted as part of the course BLG 354E. The project encompasses the extraction of a hidden message from an audio file, filtering of the extracted message, creating a game with a decision making algorithm based on audio signal filtering, and analysis of specific transfer functions.

1. The Radio (Part-1):

The code employs frequency domain manipulation to extract a hidden message from an audio file. The process entails the following steps:

- Loading the audio file and obtaining the sampling rate.
- Splitting the audio file into one-second segments.
- Computing the FFT (Fast Fourier Transform) of each segment and combining it with the FFT of the message, which is basically 0 padding since we have only 1 audio input.
- Performing inverse FFT to obtain the message in the time domain.
- Saving the extracted audio to a new WAV file.

After this process it is seen that the output audio is a message of a woman speaking clearly.

2. System Simulation (Part-2):

This section focuses on applying a filter to the extracted message. The steps involved are as follows:

- Definition of transfer function coefficients in the Z-domain.
- Creation of a TransferFunction object.
- Loading the extracted message.
- Application of the transfer function to the message signal.
- Saving the filtered audio to a new WAV file.

After this process it is seen that the output audio is a message of a woman speaking like in the first part, there is no signficant changes on the audio but there is a little enhancement in quality of the audio.

3. Listen yo Your Heart, Closely (Part-3):

In this section, we simulate the decision-making process of an agent (player) determining its movements. The goal is to detect danger zones and choose a safe path.

main() Function: This function is used to initiate the simulation. Initially, a check is made in the "down" direction to assess the environment's initial state. It's observed that when the agent moves down once at the beginning, the subsequent results are more reliable. Then, it calls the simulate() function to start the simulation.

simulate() Function: This function simulates the decision-making process. It repeatedly calls the make_decision function to determine the next action. It prints each action and stops when the make_decision function returns -1, indicating the end of the game.

make_decision(prev) Function: This function makes a decision based on the previous action. It takes the previous action as input and returns the next action to take. The actions are represented as integers: 0 for right, 1 for up, 2 for down, 3 for left, and -1 to indicate no action should be taken. It prioritizes safe movements based on the previous action and the environment's current state.

move(direction) Function: This function executes the agent's movement in the specified direction. It simulates key presses for the corresponding movement and introduces delays for reliable simulation.

check(direction) Function: This function checks for danger zones or valid paths in the specified direction. It simulates the agent's movement in that direction, listens for audio input, and determines if there is an obstacle based on the mean of the audio sample.

listen() Function: This function listens for audio input and determines if there is an obstacle based on the mean of the audio sample. It records audio using the microphone, normalizes the data, calculates the mean, and returns True if an obstacle is detected. After data normalization it is observed that the input signals are not in a shape of square or triangle correctly but there is a distinguishing pattern between them. If the mean of the sample is lower than -0.4 this means square wave, and otherwise it is triangle wave. This method is tested on various locations of the platform and it worked as expected robustly.

4. Bode Plots (Part-4):

Here, the frequency responses of specific transfer functions are examined. The process includes:

- Definition of coefficients for the relevant transfer functions.
- Creation of TransferFunctionDiscrete objects.
- Calculation of frequency responses.
- Generation of Bode plots for H1, H2, and H1 * H2 transfer functions.

The results of this part is given below with the explanations.

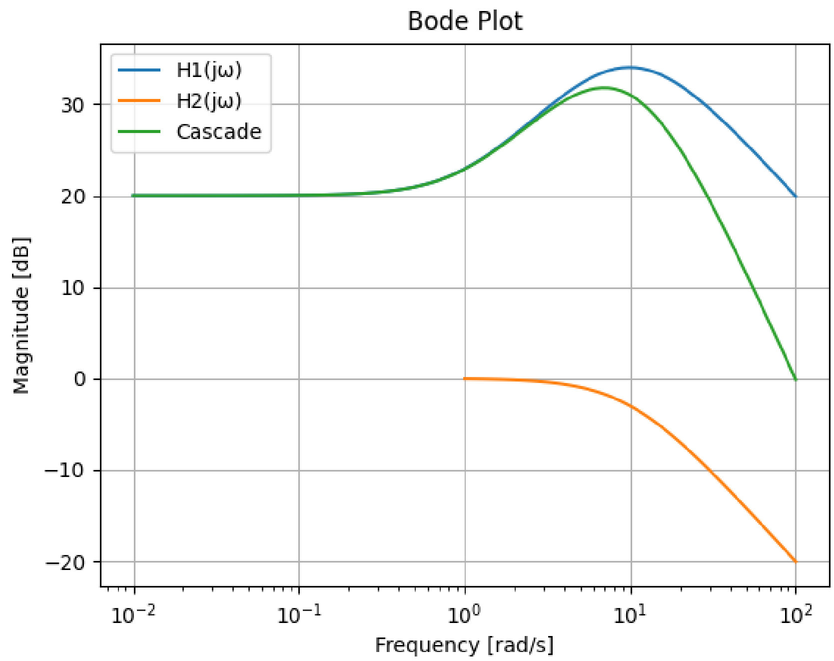

Figure 1: Magnitude Response of the Functions

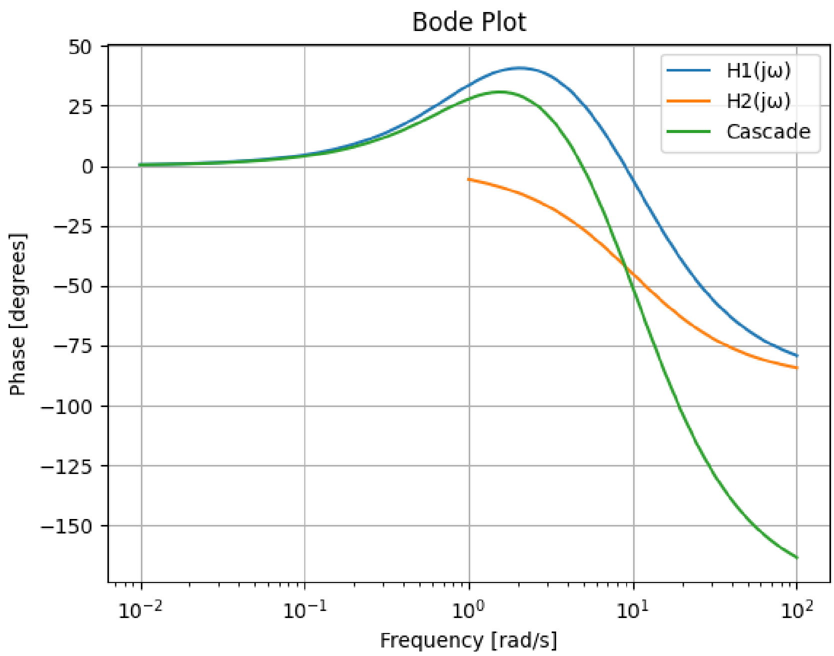

Figure 2: Phase Response of the Functions

The results are saved in phase_response.png and magnitude_response.png files. Magnitude responses and phase responses of these transfer functions are given in Figure-1 and Figure-2 respectfully.

Only the H1 and cascaded transfer functions have initial coefficient K = 10. This causes an initialization at 20dB in the magnitude response.

H2 has no such initialization so it starts from 0dB.

In H1, there is a zero at w = 1 rad/s and this causes a 20dB increment. But it has 2 poles at in w = 10 rad/s so one of them causes finish the increment of the zero and other pole causes 20 dB decrement resulting in 20dB total.

As can be seen in H2 it has a pole at w = 10 rad/s and this causes a 20dB decrement.

When H1 and H2 are cascaded, it has zero at w = 1 rad/s and 3 poles at w = 10 rad/s, one more than the H1. As a result, the response is lower than the H1 response with resulting in 0dB total. All the observations are consistent with the theory.

Moreover, the phase responses are also consistent and are as expected.

Conclusion:

This project offers a comprehensive exploration of signal processing, system analysis, and decision-making within the realm of computer engineering. Divided into four distinct parts, the project contributes to a deeper understanding of these concepts. In the first part, a hidden message is extracted from an audio file using frequency domain manipulation. The second part involves filtering the extracted message to enhance its quality. Part three simulates the decision-making process of an agent navigating through an environment, integrating signal processing techniques with real-world applications. Finally, in part four, the frequency responses of specific transfer functions are analyzed, providing insights into their behavior and characteristics. Overall, this project showcases the practical relevance of signal processing, system analysis, and decision-making in various engineering domains, deepening understanding and paving the way for further research.