**Signals and Systems Project 1 Writeup**

Jingguo Liang and Chang Yan

**Part 3: Digital Message Transmission**

For this part, we implemented two functions: encode and transmit\_noise. The function encode takes as input a string which is the message that will be sent, and a pulse signal that represents a ‘1’. It will return the signal that represents the message. The function transmit\_noise takes as input a string which is the message that will be sent, the pulse signal that represents a ‘1’, the sample rate, and the level of noise to be added to the signal. It constructs the signal by calling encode and add noise to it. Then it will call decode and try to retrieve the message from the noisified signal. Finally, it will compare if the decoded message matches the original one.

**3.1.** For 3.1 and 3.2 we will use sample rate 100 samples/s and use rectangular pulse with duration 0.2s to represent a binary 1. We will try to send the message “Acknowledged”. The noise level will be set to 0.5. Enter the following command in command window:

>> pulse = ones(1, 21);

>> msg = 'Acknowledged';

>> transmit\_noise(msg, pulse, 100, 0.5);

We have the following output from the command window:

Original message is: Acknowledged

Decoded message is: Acknowledged

Two messages match.

We also have the following graphs that shows the original and the noisified signal:

Chart

Description automatically generatedChart, bar chart

Description automatically generated

**3.2.** From the command window output we see that the decoded message is “Acknowledged”, and it matches the message that is sent.

**3.3.** First, we will construct the four types of pulse signals. Take sample rate 100, and all signals will have duration 0.2s. Enter the following command in command window:

>> pulse1 = ones(1, 21);

>> pulse2 = 0:0.05:1;

>> pulse3 = sin((0:20)\*0.1\*pi);

>> pulse4 = [1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0];

Then, we will call the function transmit\_noise using those pulses and the message “Acknowledged”, with the noise level increasing 0.1 every step.

At noise level 2.2, type1 pulse signal (rectangular) begin to unmatch.

At noise level 1.2, type2 pulse signal (triangular) begin to unmatch.

At noise level 1.3, type3 pulse signal (sinusoidal) begin to unmatch.

At noise level 1.3, type4 pulse signal begin to unmatch.

Type1 pulse (rectangular pulse) is most robust to noises.