UEE4502 Principle of Communication Engineering Term Project Report

109101013 Cho-Yuan Lee
School of Medicine
National Yang Ming Chiao Tung University
Email: rick.109101013.md09@nycu.edu.tw

***Abstract*—In this project, we are given (in .txt format) samples of modulated message signals with additive Gaussian noise. The message signals are modulated by one of these four modulation schemes: DSB-C, DSB-SC, SSB, and FM. The goal is to implement demodulation with MATLAB to recover the message signals in each data file.**

I. Setup

The system architecture on [1, pp. 2] is used in this project. Each message signal is generated from a 32-bit binary sequence. For each bit, a '1' is sent as $\sin(2\pi t)$ and a '0' is sent as $-\sin(2\pi t)$, both for a duration of 1 second.

The message signal $m(t)$ is modulated and up-converted to generate

$$s(t) = \mathrm{Re}\{\tilde{s}(t)e^{j2\pi f_c t}\} \tag{1}$$

where $\tilde{s}(t)$ is the baseband modulated signal of $m(t)$ and $f_c = 100$ Hz is the carrier frequency. The channel is corrupted by an additive Gaussian noise $w(t)$ with a carrier-to-noise ratio of 10dB. So the channel output, received by the demodulator, is

$$x(t) = s(t) + w(t). \tag{2}$$

The samples of the signal are given (in *.txt):

$$x[n] = x(nT) \tag{3}$$

where $n = 0,\ 1,\ 2,\ ...,\ N-1$, and $T = \frac{1}{1000}$ second is the sampling period.

In addition, the samples of the corrupted channel output with its modulated and up-converted signal $s(t)$ having a phase distortion of ϕ due to the asynchronization between the transmitter carrier and the receiver carrier, are also given (in *_pd.txt) to demonstrate the effects of asynchronization, where eq.(1) becomes

$$s(t) = \mathrm{Re}\Big\{\tilde{s}(t)e^{j(2\pi f_c t + \phi)}\Big\}. \tag{4}$$

II. Modulation Schemes

This section describes the four possible modulation schemes used in this project. For a more detailed description see [2].

The purpose of a communication system is to transmit baseband signals (i.e., information-bearing signals) through a communication channel separating the transmitter from the receiver. A modulator shifts the range of baseband frequencies into other frequency ranges suitable for transmission.

A. Double Sideband - Carrier (DSB-C)

$$s(t) = [1 + k_a m(t)]\cos(2\pi f_c t) \tag{5}$$

where k_a is a constant called the modulation sensitivity of the modulator.

B. Double Sideband - Suppressed Carrier (DSB-SC)

$$s(t) = m(t)\cos(2\pi f_c t) \tag{6}$$

C. Single Sideband (SSB)

$$s(t) = m(t)\cos(2\pi f_c t) \mp \hat{m}(t)\sin(2\pi f_c t) \tag{7}$$

where $\hat{m}(t)$ is the Hilbert transform of $m(t)$.

D. Frequency Modulation (FM)

$$s(t) = \cos\left[2\pi f_c t + 2\pi k_f \int_0^t m(\tau)d\tau\right] \tag{8}$$

where the constant k_f represents the frequency sensitivity of the modulator.

III. Demodulation

In a communication system, a demodulator is used to recover the information content from the modulated carrier wave.

Demodulation for the four modulation schemes mentioned in the previous section, if we neglect the effects of channel noise and asynchronization between the carriers of the transmitter and the receiver, can be done respectively as follows:

A. Double Sideband - Carrier (DSB-C)

For DSB-C, demodulation can be accomplished using an envelope detector [1, pp. 16-18]. We first take the square of the modulated signal $s(t)$ to get

$$\begin{aligned} y(t) &= [s(t)]^2 \\ &= [1 + k_a m(t)]^2 \cos^2(2\pi f_c t) \\ &= \frac{1}{2}[1 + k_a m(t)]^2\,[1 + \cos(4\pi f_c t)]\,. \end{aligned} \tag{9}$$

The resultant signal, $y(t)$, is then passed through a low-pass filter:

$$y(t) \xrightarrow{\text{Low-pass}} \frac{1}{2}[1 + k_a m(t)]^2. \tag{10}$$

Using a square-rooter, we obtain

$$w(t) = \sqrt{\frac{1}{2}[1 + k_a m(t)]^2} = \frac{1}{\sqrt{2}}[1 + k_a m(t)] \tag{11}$$

assuming that $1 + k_a m(t) \geq 0$. Finally, by blocking the DC component in $w(t)$, we get a scaled version of the message signal

$$w(t) \xrightarrow{\text{DC blocker}} \frac{1}{\sqrt{2}} k_a m(t). \tag{12}$$

B. Double Sideband - Suppressed Carrier (DSB-SC)

For DSB-SC, coherent demodulation [1, pp. 25-26] is done by multiplying the modulated signal with the carrier wave:

$$\begin{aligned} y(t) &= s(t)\cos(2\pi f_c t) \\ &= m(t)\cos^2(2\pi f_c t) \\ &= \frac{1}{2} m(t)\left[1 + \cos(4\pi f_c t)\right] \end{aligned} \tag{13}$$

The resultant signal, $y(t)$ is then passed through a low-pass filter to obtain a scaled version of the message signal:

$$y(t) \xrightarrow{\text{Low-pass}} \frac{1}{2} m(t). \tag{14}$$

C. Single Sideband (SSB)

For SSB demodulation, the procedure is the same as DSB-SC demodulation. We first multiply the modulated signal with the carrier wave:

$$\begin{aligned} y(t) &= s(t)\cos(2\pi f_c t) \\ &= m(t)\cos^2(2\pi f_c t) \mp \hat{m}(t)\sin(2\pi f_c t)\cos(2\pi f_c t) \\ &= \frac{1}{2} m(t)\left[1 + \cos(4\pi f_c t)\right] \mp \frac{1}{2}\hat{m}(t)\sin(4\pi f_c t) \end{aligned} \tag{15}$$

The resultant signal, $y(t)$ is then passed through a low-pass filter to obtain a scaled version of the message signal:

$$y(t) \xrightarrow{\text{Low-pass}} \frac{1}{2} m(t). \tag{16}$$

D. FM

For FM demodulation, see [3, pp. 175-178]. In FM,

$$s(t) = \cos\left[2\pi f_c t + \Phi(t)\right] \tag{17}$$

where $\Phi(t)$ is the phase deviation given by

$$\Phi(t) = 2\pi k_f \int_0^t m(\tau)d\tau \tag{18}$$

If $s(t)$ is passed through a differentiator, the output is

$$\begin{aligned} e(t) &= \frac{d}{dt}\cos\left[2\pi f_c t + \Phi(t)\right] \\ &= -\left[2\pi f_c + \frac{d\Phi}{dt}\right]\sin\left[2\pi f_c t + \Phi(t)\right] \end{aligned} \tag{19}$$

This is in the same form as an DSB-C modulated signal, except for the phase deviation $\Phi(t)$. Therefore, an envelope detector can be used:

- Squarer

$$\begin{aligned} y(t) &= [e(t)]^2 \\ &= \left[2\pi f_c + \frac{d\Phi}{dt}\right]^2 \sin^2\left[2\pi f_c t + \Phi(t)\right] \\ &= \frac{1}{2}\left[2\pi f_c + \frac{d\Phi}{dt}\right]^2 \left[1 - \cos(4\pi f_c t + 2\Phi(t))\right] \end{aligned} \tag{20}$$

- Low-pass filter

$$y(t) \xrightarrow{\text{Low-pass}} \frac{1}{2}\left[2\pi f_c + \frac{d\Phi}{dt}\right]^2 \tag{21}$$

- Square-rooter

$$w(t) = \sqrt{\frac{1}{2}\left[2\pi f_c + \frac{d\Phi}{dt}\right]^2} = \frac{1}{\sqrt{2}}\left[2\pi f_c + \frac{d\Phi}{dt}\right] \tag{22}$$

  if

$$f_c > -\frac{1}{2\pi}\frac{d\Phi}{dt} \text{ for all } t \tag{23}$$

  which usually holds since the carrier frequency is typically a lot greater than the bandwidth of the message signal
- DC Blocker

$$\begin{aligned} w(t) &\xrightarrow{\text{DC Blocker}} \frac{1}{\sqrt{2}}\frac{d\Phi(t)}{dt} \\ &= \frac{2}{\sqrt{2}}\pi k_f m(t) \end{aligned} \tag{24}$$

We thus obtain a scaled version of the message signal.

IV. Implementation

A. Identifying Which Modulation Scheme Was Used

To determine which modulation scheme was used for each file, we can first plot its waveform with the following code:

```
% from t = 0 to t = 32, spacing = 0.001
t = transpose(0:0.001:32-0.001);
filename = '%d.txt';
for i = 1:5
    fileID = fopen(sprintf(filename, i), 'r');
    % read samples into vector x
    x = fscanf(fileID, '%f');
    fclose(fileID);
    subplot(5, 1, i);
    % plot the first two seconds
    plot(t(1:2000), x(1:2000));
end
```

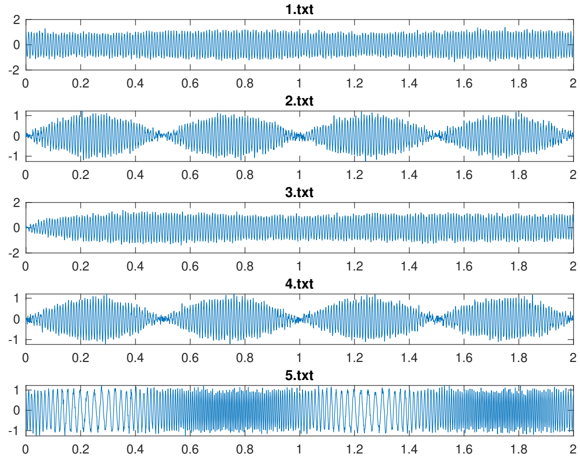

Fig. 1: Waveform of each $s_i(t)$ from t = 0 to t = 2 (sec)

Let $s_i(t)$ denote the i-th modulated signal. As shown in Fig.1, we can see that

- $s_2(t)$ and $s_4(t)$ have 'phase-reversal'
- The instantaneous frequency of $s_5(t)$ varies over time

Let $S_i(f)$ denote the spectrum of the i-th modulated signal. If we also plot the spectrum of each modulated signal with:

```
for i = 1:5
    fileID = fopen(sprintf(filename, i), 'r');
    x = fscanf(fileID, '%f');
    fclose(fileID);
    % F.T. of x
    y = fft(x);
    % take magnitude
    p2 = abs(y/L);
    % single-sided spectrum
    p1 = p2(1:L/2+1);
    p1(2:end-1) = 2*p1(2:end-1);
    f = fs * (0:(L/2))/L;
    subplot(5, 1, i);
    plot(f, p1);
end
```

then from Fig.2, we can further find that

- $S_2(f)$ and $S_4(f)$ exhibits symmetry about $f = 100$ Hz
- For $S_3(f)$, only the upper sideband was kept
- $S_5(f)$ spans across a range of frequencies ($\Delta f \approx 70$ Hz)

Therefore, we can conclude that

- $s_1(t)$ was modulated with DSB-C
- $s_2(t)$ and $s_4(t)$ was modulated with DSB-SC
- $s_3(t)$ was modulated with SSB
- $s_4(t)$ was modulated with FM

B. *Demodulation process*

1) Reading the samples from each file:

```
fc = 100; % carrier frequency
fs = 1000; % sampling rate
% Read samples of x(t) into x
filename = '*.txt';
fileID = fopen(filename, 'r');
x = fscanf(fileID, '%f');
fclose(fileID);
```

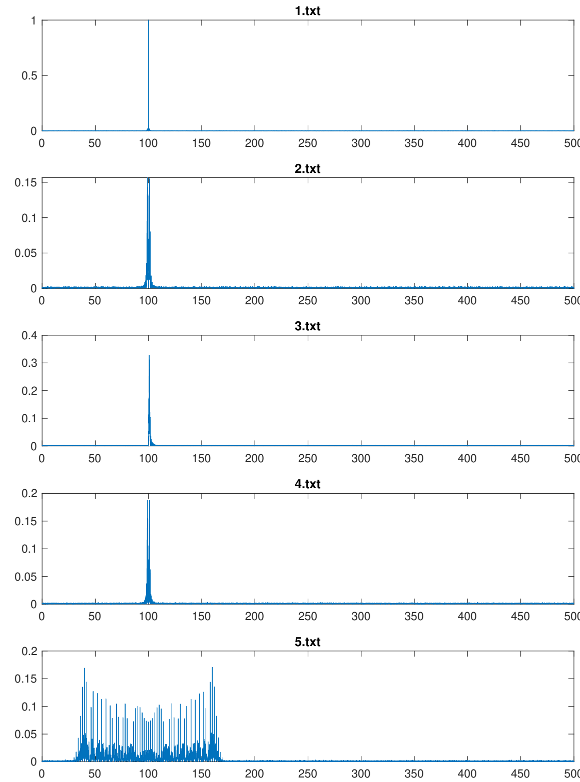

Fig. 2: Single-sided magnitude spectrum of each $s_i(t)$

2) Demodulation: the only step different in each program.

```
% For DSB-C: Envelope Detector
y = x.^2; % Squarer
z = lowpass(y, 20, fs); % Low-pass, cutoff @ 20Hz
w = sqrt(z); % Square rooter
% DC Blocker: assuming m(t), noise are zero-mean
m = w - mean(w);
% Recover its magnitude (k_a?)
m = m * sqrt(2);
```

```
% For DSB-SC and SSB: Coherent Demodulation
% Local oscillator at Rx
localOsc = cos(2 * pi * fc * t);
y = x .* localOsc;
m = lowpass(y, 20, fs); % Low-pass, cutoff @ 20Hz
% Recover its magnitude
m = m .* 2;
```

```
% For FM: Differentiator + Envelope Detector
% Differentiator
dt = 0.001;
dx = diff(x); % returns [x2-x1, x3-x2, ... ]
dxdt = dx/dt;
e(1) = 0; % Padding, since dxdt is of length 31999
e(2:32000) = dxdt;
y = e.^2; % Squarer
z = lowpass(y, 20, fs); % Low-pass, cutoff @ 20Hz
w = sqrt(z); % Square rooter
% DC Blocker: assuming m(t), noise are zero-mean
m = w - mean(w);
% Recover its magnitude
m = m * sqrt(2);
m = m / (2 * pi * 70); % in 5, df ~ 70
```

3) Recovering the sequence and converting to decimal:

```
% A represents how a sequence of 1's is sent
A = sin(2*pi*t);
% sequence with all elements set to zero
bin_seq = zeros(32, 1);
% Check the entire signal duration of 32 seconds
for i = 0:31
    pos = 0;
    neg = 0;
    % For each second, check if this part of m(t)
    % is in-phase with sin(2*pi*t)
    for j = 1:fs
        % For every sample of m(t),
        % if it has the same sign as the
        % corresponding point in A, then pos += 1;
        eval = m(i*fs+j) / A(i*fs+j);
        if eval > 0
            pos = pos + 1;
        else
            neg = neg + 1;
        end
    end
    % If pos > neg, then this part of m(t)
    % is "probably" in-phase with sin(2*pi*t)
    % meaning that this bit should be set to 1
    % Otherwise, it should be left unchanged (0)
    if pos > neg
        bin_seq(i+1) = 1;
    end
end

% Join the binary sequence into a string
bitString = strjoin(string(int8(bin_seq)));
% Convert to decimal
dec = bin2dec(bitString);
% Display results
disp(dec);
```

V. Results & Conclusion

A. Demodulation Results

Fig.3 - Fig.7 show each waveform of the demodulated signals (estimates of $m_i(t)$).After recovering the 32-bit sequences, we can then convert them back to decimals as shown in Table I.

File Name	Modulation	Decimal
1.txt	DSB-C	10101309
2.txt	DSB-SC	10101355
3.txt	SSB	10101351
4.txt	DSB-SC	10101302
5.txt	FM	10101327

TABLE I: Modulation and recovered decimals in each file

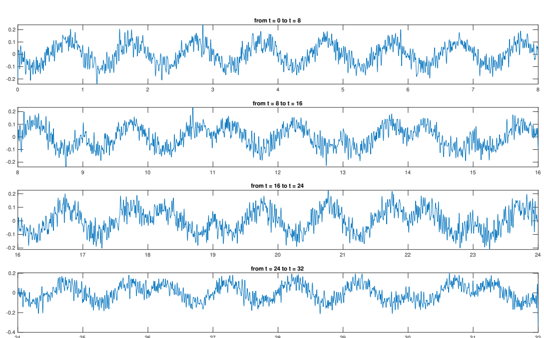

Fig. 3: Estimate of $k_a m_1(t)$

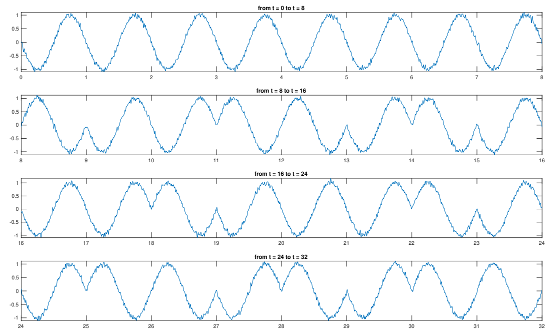

Fig. 4: Estimate of $m_2(t)$

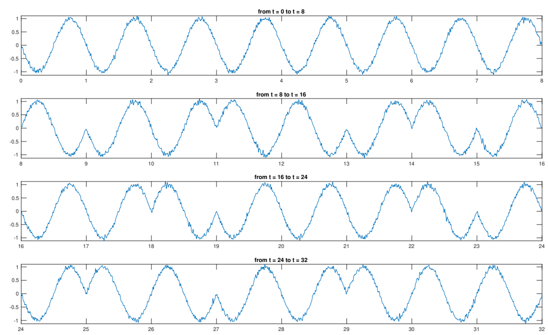

Fig. 5: Estimate of $m_3(t)$

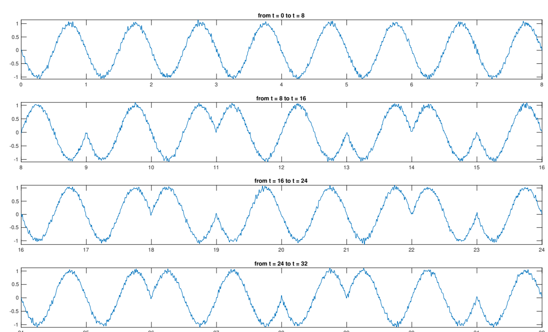

Fig. 6: Estimate of $m_4(t)$

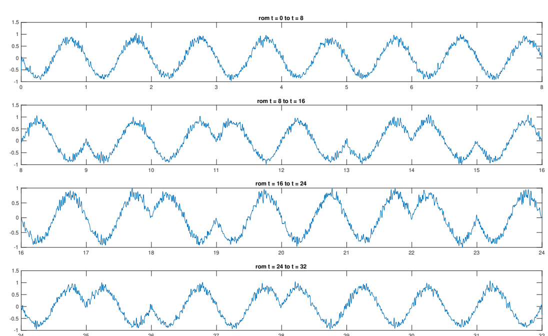

Fig. 7: Estimate of $m_5(t)$

B. Effects of Phase Distortion

Fig.8 - Fig.12 show each waveform of the demodulated signals (estimates of $m_i(t)$) with phase distortion. Altough

the waveforms are slightly different from those without phase distortion, I was still able to obtain the same results for the decimals as shown in Table II.

File Name	Modulation	Decimal
1_pd.txt	DSB-C	10101309
2_pd.txt	DSB-SC	10101355
3_pd.txt	SSB	10101351
4_pd.txt	DSB-SC	10101302
5_pd.txt	FM	10101327

TABLE II: Modulation and recovered decimals in each file

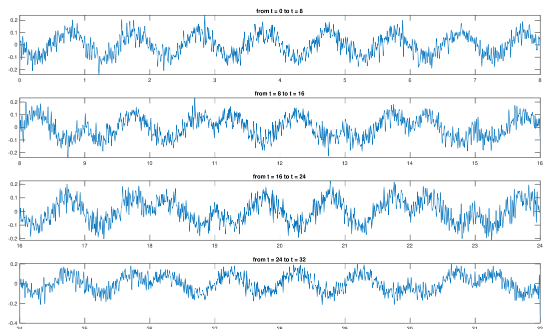

Fig. 8: Estimate of $k_a m_1(t)$ with phase distortion

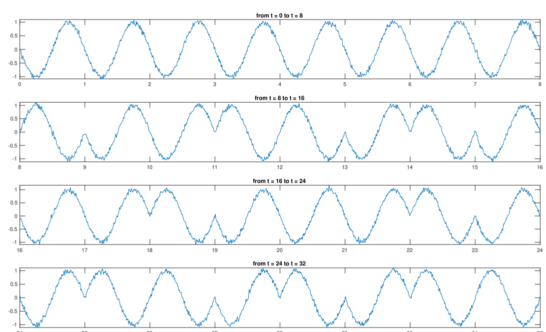

Fig. 9: Estimate of $m_2(t)$ with phase distortion

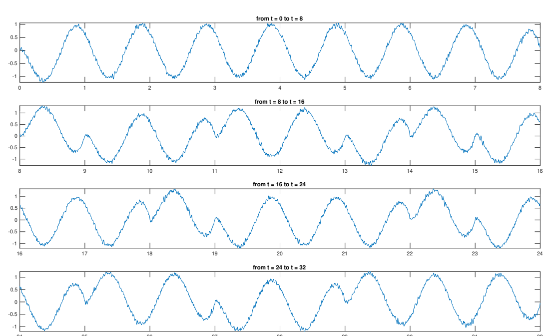

Fig. 10: Estimate of $m_3(t)$ with phase distortion

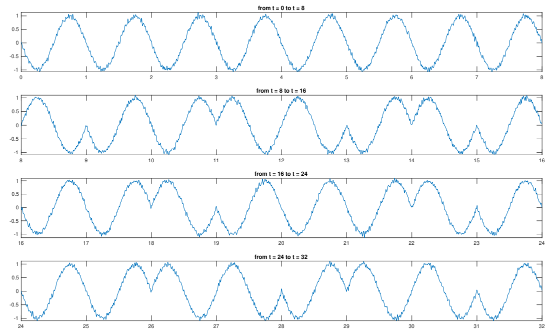

Fig. 11: Estimate of $m_4(t)$ with phase distortion

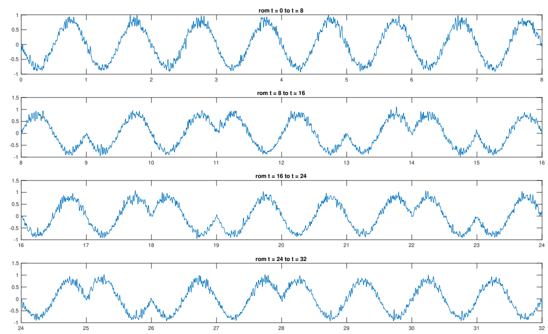

Fig. 12: Estimate of $m_5(t)$ with phase distortion

C. Difficulties Encountered

The modulation and demodulation processes were pretty straight forward and were well-explained in [1], [2], and [3]. Implementation of the demodulators with MATLAB can actually be done in just a few lines of codes. However, since I had no experience with MATLAB, a large portion of the time I spent on this project was actually for looking up basic syntax, functions, etc. Furthermore, since I included quite a lot of equations and figures in this report, using Word for typesetting was a complete disaster. I eventually gave up and switched to LaTeX. Hopefully the 'docx' format converted from pdf would still look fine.

REFERENCES

[1] P.-N. Chen, "Part 4: Noisefree Analog Modulation and Demodulation," http://shannon.cm.nctu.edu.tw/comtheory/p4-2020.pdf, 2022.

[2] S. Haykin, *Communication systems*. John Wiley & Sons, 2008.

[3] R. E. Ziemer and W. H. Tranter, *Principles of communications*. John Wiley & Sons, 2014.