Homework 3

Course Title: Digital Signal Processing I (Spring 2020)

Course Number: ECE53800

Instructor: Dr. Li Tan

Author: **Zhankun Luo**

Problems 4.2, 4.5, 4.6, 4.9, 4.14, 4.19 (a), (b), 4.20, 4.21, 4.22, 4.25, 4.26

Advanced problems

4.37, 4.38, 4.41

MATLAB problem

4.31, 4.32 (optional)

Problems

Problem 4.2

Given a sequence x(n) for $0 \le n \le 3$, where $x(0)=4, x(1)=3, x(2)=2, \text{ and } x(3)=1$, evaluate its DFT X(k).

solution

From the formula: $X(k)=\sum_{n=0}^{N-1} x(n)e^{-j2\pi\frac{kn}{N}}$, N=4

$$\begin{aligned}
X(0) &= x(0)e^{-j2\pi\frac{0\times0}{4}} + x(1)e^{-j2\pi\frac{0\times1}{4}} + x(2)e^{-j2\pi\frac{0\times2}{4}} + x(3)e^{-j2\pi\frac{0\times3}{4}} \\
&= 4\times1+3\times1+2\times1+1\times1=10 \\
X(1) &= x(0)e^{-j2\pi\frac{1\times0}{4}} + x(1)e^{-j2\pi\frac{1\times1}{4}} + x(2)e^{-j2\pi\frac{1\times2}{4}} + x(3)e^{-j2\pi\frac{1\times3}{4}} \\
&= 4\times1+3\times(-j)+2\times(-1)+1\times j=2-j2 \\
X(2) &= x(0)e^{-j2\pi\frac{2\times0}{4}} + x(1)e^{-j2\pi\frac{2\times1}{4}} + x(2)e^{-j2\pi\frac{2\times2}{4}} + x(3)e^{-j2\pi\frac{2\times3}{4}} \\
&= 4\times1+3\times(-1)+2\times1+1\times(-1)=2 \\
X(3) &= x(0)e^{-j2\pi\frac{3\times0}{4}} + x(1)e^{-j2\pi\frac{3\times1}{4}} + x(2)e^{-j2\pi\frac{3\times2}{4}} + x(3)e^{-j2\pi\frac{3\times3}{4}} \\
&= 4\times1+3\times j+2\times(-1)+1\times(-j)=2+j2
\end{aligned}$$

So, its DFT sequence $X(k)=[10, 2-j2, 2, 2+j2]$

Problem 4.5

Given the DFT sequence X(k) for $0 \le k \le 3$ obtained in Problem 4.2, evaluate its inverse of DFT x(n).

solution

Since, $\overline{X}(k)$ is conjugate of $X(k)$, we have

$$x(n)=\frac{1}{N}\sum_{k=0}^{N-1}X(k)e^{j2\pi\frac{kn}{N}}=\frac{1}{N}\overline{\sum_{k=0}^{N-1}\overline{X}(k)e^{-j2\pi\frac{kn}{N}}}=\frac{1}{N}\overline{\mathrm{DFT}[\overline{X}(k)]}$$

So, we can calculate the inverse of DFT by applying DFT to $\overline{X}(k)$

$$x(0)=\frac{1}{4}\overline{[10\times1+(2+j2)\times1+2\times1+(2-j2)\times1]}=4$$
$$x(1)=\frac{1}{4}\overline{[10\times1+(2+j2)\times(-j)+2\times(-1)+(2-j2)\times j]}=3$$
$$x(1)=\frac{1}{4}\overline{[10\times1+(2+j2)\times(-1)+2\times1+(2-j2)\times(-1)]}=2$$
$$x(1)=\frac{1}{4}\overline{[10\times1+(2+j2)\times j+2\times(-1)+(2-j2)\times(-j)]}=1$$

Then we have:

Its inverse of DFT $x(n)=[4,3,2,1]$

Problem 4.6

Given a sequence x(n), where $x(0)=4, x(1)=3, x(2)=2,\ \text{and}\ x(3)=1$ with the last two data zero padded as $x(4)=0,\ \text{and}\ x(5)=0$, evaluate its DFT X(k).

solution

Its DFT $X(k)=[10,3.5-j4.330127,2.5-j0.866025,2,2.5+j0.866025,3.5+j4.330127]$

Problem 4.9

Using the DFT sequence X(k) for $0\leq k\leq 5$ computed in Problem 4.6, evaluate the inverse of DFT for x(0) and x(4).

solution

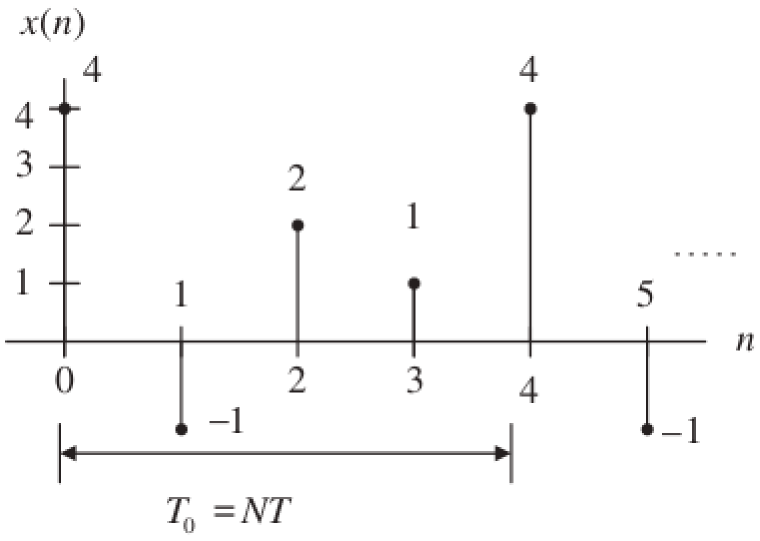
By applying IDFT to X(k):

We have the inverse of DFT for $x(n)=[4,3,2,1,0,0]$

So, x(0)=4, x(4)=0

Problem 4.14

Given the sequence in Fig. 4.46 and assuming $f_s=100Hz$, compute the amplitude spectrum, phase spectrum, and power spectrum.



solution

We define the amplitude spectrum as

$$A_k = \frac{1}{N}|X(k)| = \frac{1}{N}\sqrt{(\mathrm{Real}[X(k)])^2 + (\mathrm{Imag}[X(k)])^2}, k = 0, 1, 2, \cdots, N-1$$

Correspondingly, the phase spectrum is given by, in $(-\pi, \pi]$

$$\varphi_k = \tan^{-1}\left(\frac{\mathrm{Imag}[X(k)]}{\mathrm{Real}[X(k)]}\right), k = 0, 1, 2, \cdots, N-1$$

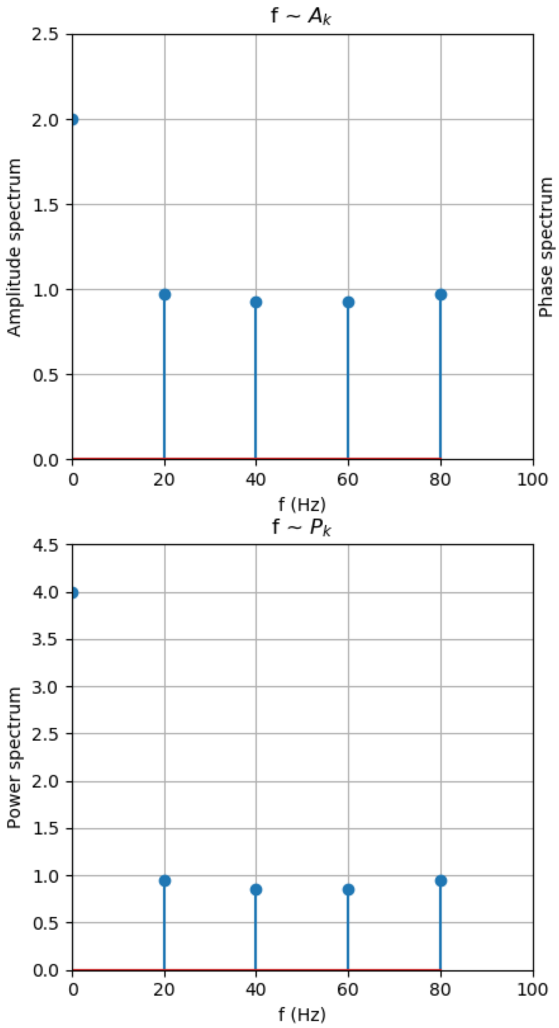
The DFT power spectrum is defined as

$$P_k = \frac{1}{N^2}|X(k)|^2 = \frac{1}{N^2}\left\{(\mathrm{Real}[X(k)])^2 + (\mathrm{Imag}[X(k)])^2\right\}, k = 0, 1, 2, \cdots, N-1$$

and map the frequency bin k to its corresponding frequency as

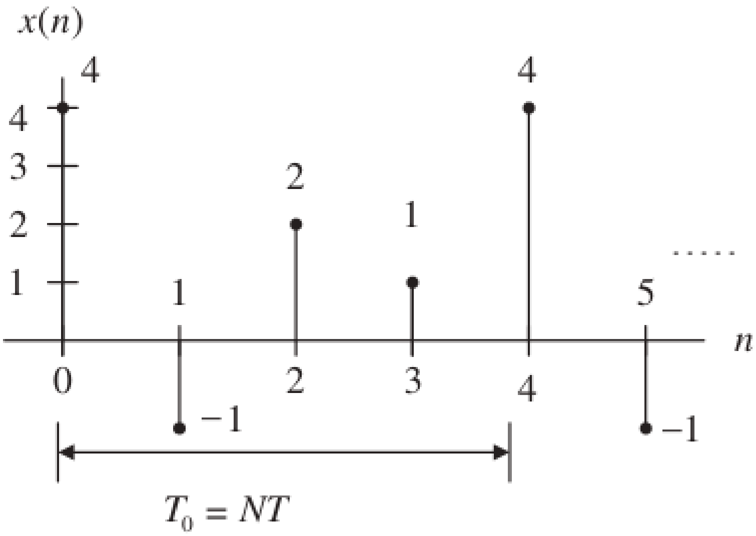
$$f = \frac{kf_s}{N}$$

1. Amplitude spectrum
2. phase spectrum
3. power spectrum



Problem 4.19

Given a sequence in Fig. 4.47 where $f_s = 100Hz$ and $T = 0.01s$, compute the amplitude spectrum, phase spectrum, and power spectrum using

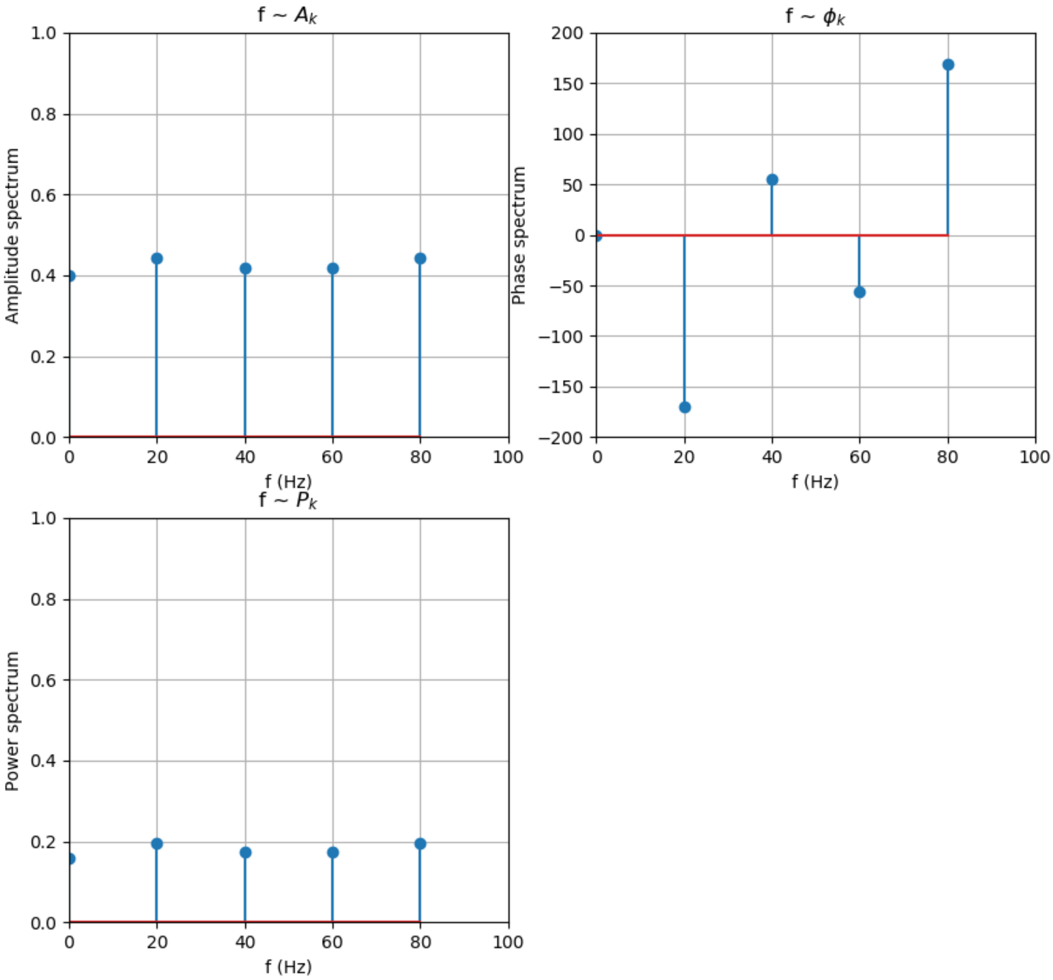


1. Triangular window.
2. Hamming window.

solution

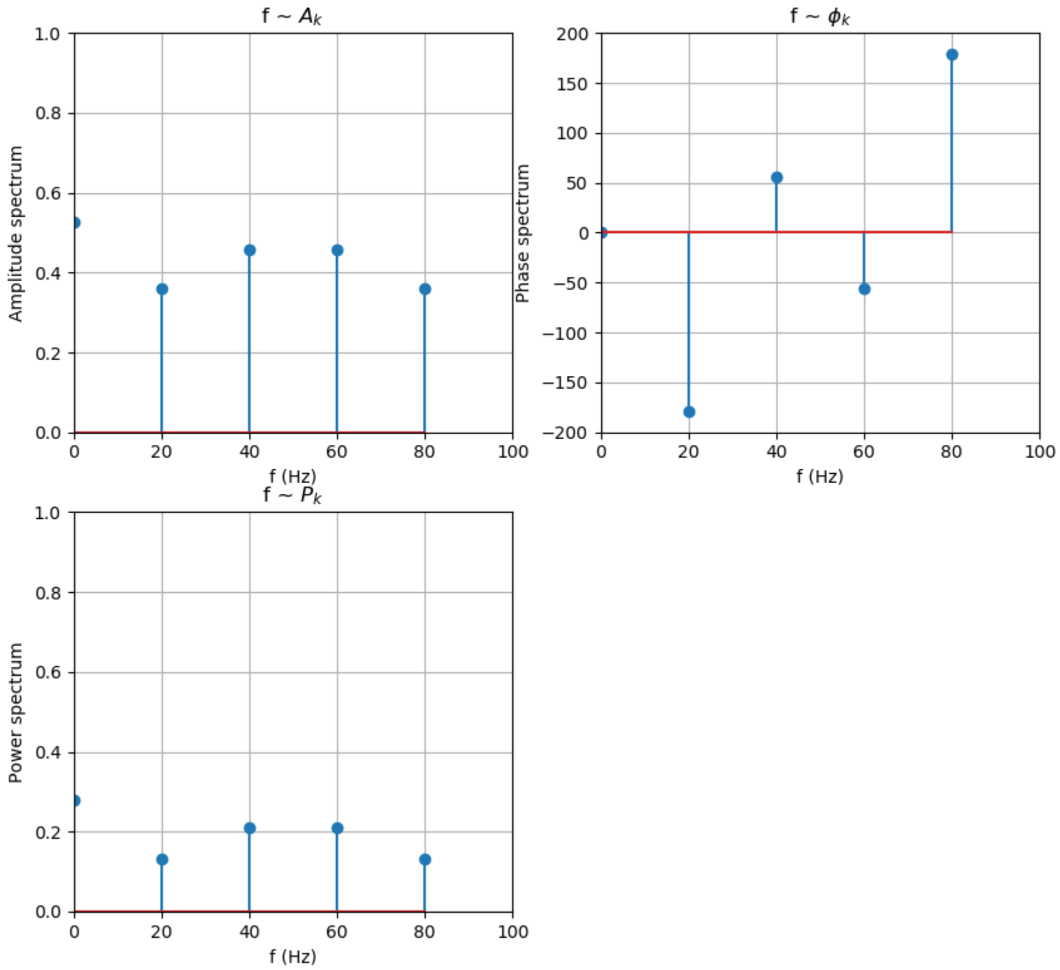
1. Triangular window.

$$w_{tri}(n) = 1 - \frac{|2n - N + 1|}{N - 1}, 0 \leq n \leq N - 1$$



2. Hamming window

$$w_{hm}(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1$$



Problem 4.20

Given a sinusoid

$$x(n) = 2 \times \sin\left(2000 \times 2\pi \times \frac{n}{8000}\right)$$

obtained by using the sampling rate of $f_s = 8000Hz$, we apply the DFT to compute the amplitude spectrum.

1. Determine the frequency resolution when the data length is 100 samples.

   Without using the window function, is there any spectral leakage in the computed spectrum? Explain.
2. Determine the frequency resolution when the data length is 73 samples.

   Without using the window function, is there any spectral leakage in the computed spectrum? Explain.

solution

N_T is the minimal positive period for $x(n)$

$$2000 \times 2\pi \times \frac{N_T}{8000} = 2\pi \Longrightarrow N_T = 4$$

1. frequency resolution: $\Delta f = \frac{f_s}{N} = \frac{8000}{100} = 80Hz$

   No spectra leakage, explanation:

   $N = 100 = 25N_T$, is integer multiple of minimum positive period

   So N is period for $x(n)$, no leakage

2. frequency resolution: $\Delta f = \frac{f_s}{N} = \frac{8000}{73} = 109.589Hz$

   There is spectra leakage, explanation:

   No spectra leakage, explanation:

   $N = 73 = 18N_T + 1$, is NOT integer multiple of minimum positive period

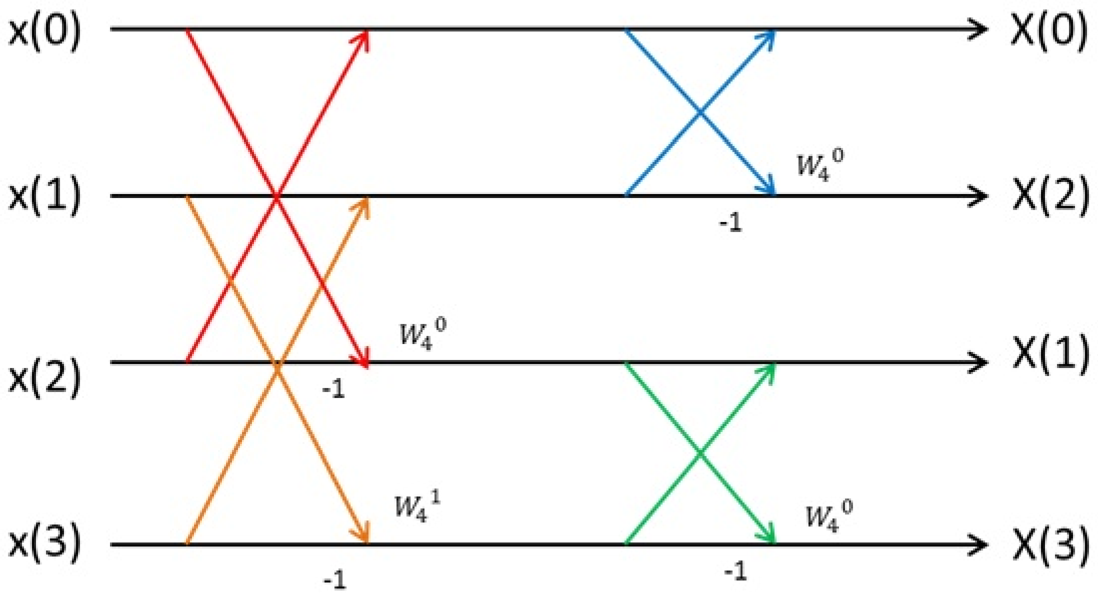
   So N is NOT period for $x(n)$, there is spectra leakage

Problem 4.21

Given a sequence x(n) for $0 \le n \le 3$, where $x(0) = 4, x(1) = 3, x(2) = 2, \text{ and } x(3) = 1$, evaluate its DFT X(k) using the decimation-in-frequency FFT method, and determine the number of complex multiplications.

solution

1. decimation-in-frequency FFT method



   After 1st level of butterfly operations:

$$x(0) + x(2) = 4 + 2 = 6$$
$$x(1) + x(3) = 3 + 1 = 4$$
$$[x(0) - x(2)] \cdot W_4^0 = [4 - 2] \times 1 = 2$$
$$[x(1) - x(3)] \cdot W_4^1 = [3 - 1] \times (-j) = -j2$$

After 2nd level:

$$
\begin{aligned}
6+4 &= 10 \\
[6-4]\cdot W_4^0 = [6-4]\times 1 &= 2 \\
2+(-j2) &= 2-j2 \\
[2-(-j2)]\cdot W_4^0 = [2-(-j2)]\times 1 &= 2+j2
\end{aligned}
$$

So we have $X(0)=10, X(2)=2, X(1)=2-j2, X(3)=2+j2$

Finally, we evaluate its DFT $X(k)=[10, 2-j2, 2, 2+j2]$

2. For $N=2^n$ points FFT:

   It has n-level butterfly operations, each level has $\frac{N}{2}=2^{n-1}$ butterfly operations,

   and each butterfly operation has 1 complex multiplication.

   So, the number of complex multiplication is $n\times 2^{n-1}=\frac{N}{2}\log(N)$

   Here $N=4=2^2$

   The number of complex multiplication is $\frac{N}{2}\log(N)=2\times 2=4$

Problem 4.22

Given the DFT sequence X(k) for $0\le k\le 3$ obtained in Problem 4.21, evaluate its inverse DFT x(n) using the decimation-in-frequency FFT method.

solution

Since, $\overline{X}(k)$ is conjugate of $X(k)$, we have

$$
x(n)=\frac{1}{N}\sum_{k=0}^{N-1}X(k)e^{j2\pi\frac{kn}{N}}=\frac{1}{N}\overline{\sum_{k=0}^{N-1}\overline{X}(k)e^{-j2\pi\frac{kn}{N}}}=\frac{1}{N}\overline{\mathrm{DFT}[\overline{X}(k)]}
$$

We can first calculate conjugate for DFT of $\overline{X}(k)$, then divide by N

In the other way, we can replace W_N^k with $\overline{W_N^k}$ in FFT calculation diagram, then divide by N

After 1st level of butterfly operations:

$$
\begin{aligned}
X(0)+X(2)=10+2 &= 12 \\
X(1)+X(3)=(2-j2)+(2+j2) &= 4 \\
[X(0)-X(2)]\cdot \overline{W}_4^0 = [10-2]\times 1 &= 8 \\
[X(1)-X(3)]\cdot \overline{W}_4^1 = [(2-j2)-(2+j2)]\times j &= 4
\end{aligned}
$$

After 2nd level:

$$
\begin{aligned}
12+4 &= 16 \\
[12-4]\cdot \overline{W}_4^0 = [12-4]\times 1 &= 8 \\
8+4 &= 12 \\
[8-4]\cdot \overline{W}_4^0 = [8-4]\times 1 &= 4
\end{aligned}
$$

Then divide by N=4

$$x(0) = 16/4 = 4, x(2) = 8/4 = 2, x(1) = 12/4 = 3, x(3) = 4/4 = 1$$

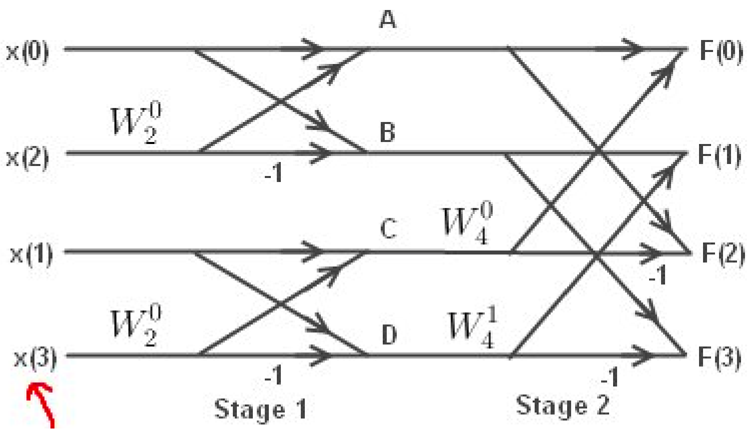
Finally, we evaluate its DFT $x(n) = [4, 3, 2, 1]$

Problem 4.25

Given a sequence x(n) for $0 \le n \le 3$, where $x(0) = 4, x(1) = 3, x(2) = 2, \text{ and } x(3) = 1$, evaluate its DFT X(k) using the decimation-in-time FFT method, and determine the number of complex multiplications.

solution

1. decimation-in-frequency FFT method

   After 1st level of butterfly operations:

   $$x(0) + W_4^0 x(2) = 4 + 1 \times 2 = 6$$
   $$x(0) - W_4^0 x(2) = 4 - 1 \times 2 = 2$$
   $$x(1) + W_4^0 x(3) = 3 + 1 \times 1 = 4$$
   $$x(1) - W_4^0 x(3) = 3 - 1 \times 1 = 2$$

   After 2nd level:

   $$6 + W_4^0 4 = 6 + 1 \times 4 = 10$$
   $$2 + W_4^1 2 = 2 + (-j) \times 2 = 2 - j2$$
   $$6 - W_4^0 4 = 6 - 1 \times 4 = 2$$
   $$2 - W_4^1 2 = 2 - (-j) \times 2 = 2 + j2$$

   So we have $X(0) = 10, X(1) = 2 - j2, X(2) = 2, X(3) = 2 + j2$

   Finally, we evaluate its DFT $X(k) = [10, 2 - j2, 2, 2 + j2]$

2. For $N = 2^n$ points FFT:

   It has n-level butterfly operations, each level has $\frac{N}{2} = 2^{n-1}$ butterfly operations,

   and each butterfly operation has 1 complex multiplication.

   So, the number of complex multiplication is $n \times 2^{n-1} = \frac{N}{2}\log(N)$

   Here $N = 4 = 2^2$

   The number of complex multiplication is $\frac{N}{2}\log(N) = 2 \times 2 = 4$

Problem 4.26

Given the DFT sequence X(k) for $0 \leq k \leq 3$ computed in Problem 4.25, evaluate its inverse DFT x(n) using the decimation-in-time FFT method.

solution

Since, $\overline{X}(k)$ is conjugate of $X(k)$, we have

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi\frac{kn}{N}} = \overline{\frac{1}{N}\sum_{k=0}^{N-1} \overline{X}(k)e^{-j2\pi\frac{kn}{N}}} = \frac{1}{N}\overline{\mathbf{DFT}[\overline{X}(k)]}$$

We can first calculate conjugate for DFT of $\overline{X}(k)$, then divide by N

In the other way, we can replace W_N^k with $\overline{W_N^k}$ in FFT calculation diagram, then divide by N

After 1st level of butterfly operations:

$$\begin{aligned} X(0) + \overline{W}_4^0 X(2) = 10 + 1 \times 2 = 12 \\ X(0) - \overline{W}_4^0 X(2) = 10 - 1 \times 2 = 8 \\ X(1) + \overline{W}_4^0 X(3) = (2 - j2) + 1 \times (2 + j2) = 4 \\ X(1) - \overline{W}_4^0 X(3) = (2 - j2) - 1 \times (2 + j2) = -j4 \end{aligned}$$

After 2nd level:

$$\begin{aligned} 12 + \overline{W}_4^0 4 = 12 + 1 \times 4 = 16 \\ 8 + \overline{W}_4^1 (-j4) = 8 + j \times (-j4) = 12 \\ 12 - \overline{W}_4^0 4 = 12 - 1 \times 4 = 8 \\ 8 - \overline{W}_4^1 (-j4) = 8 - j \times (-j4) = 4 \end{aligned}$$

Then divide by N=4

$$x(0) = 16/4 = 4, x(1) = 12/4 = 3, x(2) = 8/4 = 2, x(3) = 4/4 = 1$$

Finally, we evaluate its DFT $x(n) = [4, 3, 2, 1]$

Advanced Problems

Problem 4.37

Consider a sequence of x(n) defined for $0 \leq n < N$, where N = even

$$x(n) = \begin{cases} 1 & n = \text{ even} \\ 0 & n = \text{ odd} \end{cases}$$

show that

$$X(k) = \begin{cases} N/2 & k = 0, k = N/2 \\ 0 & \text{elsewhere} \end{cases}$$

solution

We know when N is even:

$$\begin{aligned}
X(k) &= \sum_{n=0}^{N-1} x(n) e^{-j2\pi\frac{kn}{N}} \\
&= \sum_{n'=0}^{N/2-1} x(2n') e^{-j2\pi\frac{k2n'}{N}} + \sum_{n'=0}^{N/2-1} x(2n'+1) e^{-j2\pi\frac{k(2n'+1)}{N}} \\
&= \sum_{n'=0}^{N/2-1} 1\cdot e^{-j2\pi\frac{k2n'}{N}} + \sum_{n'=0}^{N/2-1} 0\cdot e^{-j2\pi\frac{k(2n'+1)}{N}} \\
&= \sum_{n'=0}^{N/2-1} e^{-j2\pi\frac{kn'}{N/2}} \\
&= \begin{cases} \sum_{n'=0}^{N/2-1} 1 = N/2 & k=0, k=N/2 \\ \frac{1-(e^{-j2\pi\frac{k}{N/2}})^{N/2}}{1-e^{-j2\pi\frac{k}{N/2}}} = \frac{1-1^k}{1-e^{-j2\pi\frac{k}{N/2}}} = 0 & \text{elsewhere} \end{cases}
\end{aligned}$$

Problem 4.38

A sequence is shown below:

$$x(n) = \frac{1}{2}\left(1 - \cos\frac{2\pi n}{N}\right) \text{ for } 0 \le n < N$$

show that the DFT of x(n) is given by

$$X(k) = \begin{cases} N/2 & k=0 \\ -N/4 & k=1, k=N-1 \\ 0 & \text{elsewhere} \end{cases}$$

solution

We can calculate $X(k)$ as below:

$$\begin{aligned}
X(k) &= \sum_{n=0}^{N-1} x(n) e^{-j2\pi\frac{kn}{N}} \\
&= \sum_{n=0}^{N-1} \frac{1}{2}\left(1 - \cos\frac{2\pi n}{N}\right) e^{-j2\pi\frac{kn}{N}} \\
&= \sum_{n=0}^{N-1} \frac{1}{2}\left(1 - \frac{1}{2}e^{j\frac{2\pi n}{N}} - \frac{1}{2}e^{-j\frac{2\pi n}{N}}\right) e^{-j2\pi\frac{kn}{N}} \\
&= \frac{1}{2}\sum_{n=0}^{N-1} e^{-j2\pi\frac{kn}{N}} - \frac{1}{4}\sum_{n=0}^{N-1} e^{-j2\pi\frac{(k-1)n}{N}} - \frac{1}{4}\sum_{n=0}^{N-1} e^{-j2\pi\frac{(k+1)n}{N}}
\end{aligned}$$

Here we know:

$$\sum_{n=0}^{N-1} e^{-j2\pi \frac{kn}{N}} = \begin{cases} \sum_{n'=0}^{N-1} 1 = N & k = k'N, k' \in Z \\ \frac{1-(e^{-j2\pi \frac{k}{N}})^N}{1-e^{-j2\pi \frac{k}{N}}} = \frac{1-1^k}{1-e^{-j2\pi \frac{k}{N}}} = 0 & \text{elsewhere} \end{cases}$$
$$= N \sum_{k' \in Z} \delta(k - k'N)$$

So, we have (* represent convolution)

$$\begin{aligned} X(k) &= \frac{1}{2} N \sum_{n=0}^{N-1} e^{-j2\pi \frac{kn}{N}} - \frac{1}{4} N \sum_{n=0}^{N-1} e^{-j2\pi \frac{(k-1)n}{N}} - \frac{1}{4} N \sum_{n=0}^{N-1} e^{-j2\pi \frac{(k+1)n}{N}} \\ &= \frac{1}{2} N \sum_{k' \in Z} \delta(k - k'N) - N \frac{1}{4} \sum_{k' \in Z} \delta(k - 1 - k'N) - \frac{1}{4} N \sum_{k' \in Z} \delta(k + 1 - k'N) \\ &= \frac{1}{2} N \sum_{k' \in Z} \delta(k - k'N) - \frac{1}{4} N \sum_{k' \in Z} \delta(k - 1 - k'N) - \frac{1}{4} N \sum_{k' \in Z} \delta(k - (N-1) - k'N) \\ &= \sum_{k' \in Z} N [\frac{1}{2} \delta(k - k'N) - \frac{1}{4} \delta(k - 1 - k'N) - \frac{1}{4} \delta(k - (N-1) - k'N)] \\ &= N [\frac{1}{2} \delta(k) - \frac{1}{4} \delta(k-1) - \frac{1}{4} \delta(k - (N-1))] * \sum_{k' \in Z} \delta(k - k'N) \end{aligned}$$

When $0 \le k < N$, we have:

$$\begin{aligned} X(k) &= N [\frac{1}{2} \delta(k) - \frac{1}{4} \delta(k-1) - \frac{1}{4} \delta(k - (N-1))] \\ &= \begin{cases} N/2 & k = 0 \\ -N/4 & k = 1, k = N-1 \\ 0 & \text{elsewhere} \end{cases} \end{aligned}$$

Problem 4.41

For a real sequence x(n) defined for $0 \le n < N$, its discrete-cosine transform is given by

$$X_{\text{DCT}}(k) = \sum_{n=0}^{N-1} 2x(n) \cos\left[\frac{\pi k(2n+1)}{2N}\right], 0 \le k < N$$

Show that

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} \alpha(k) X_{\text{DCT}}(k) \cos\left[\frac{\pi k(2n+1)}{2N}\right], 0 \le n < N$$

where

$$\alpha(k) = \begin{cases} 1/2 & k = 0 \\ 1 & 0 < k < N \end{cases}$$

solution

Consider the new sequence:

$$x_{new}(n) = [x(0) \cdots x(N-1), x(N-1), \cdots x(0)]$$

Then, we find the relationship between 2N point DFT and N point DCT:

$$\begin{aligned}
X_{new}(k) = \mathrm{DFT}[x_{new}(n)] &= \sum_{n=0}^{N-1} x(n)e^{-j2\pi\frac{kn}{2N}} + \sum_{n=N}^{2N-1} x(-n+2N-1)e^{-j2\pi\frac{kn}{2N}} \\
&= \sum_{n=0}^{N-1} x(n)e^{-j2\pi\frac{kn}{2N}} + \sum_{n=0}^{N-1} x(n)e^{-j2\pi\frac{k(-n-1+2N)}{2N}} \\
&= \sum_{n=0}^{N-1} x(n)[e^{-j2\pi\frac{kn}{2N}} + e^{j2\pi\frac{k(n+1)}{2N}}] \\
&= e^{j2\pi\frac{k}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi k(2n+1)}{2N}\right] \\
&= \mathrm{Phase} \times \mathrm{Amplitude} \\
&= e^{j2\pi\frac{k}{4N}} X_{DCT}(k) \\
X_{new}(2N-k) &= e^{j2\pi\frac{(2N-k)}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi(2N-k)(2n+1)}{2N}\right] \\
&= (-1)e^{-j2\pi\frac{k}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi(-2N+k)(2n+1)}{2N}\right] \\
&= (-1)e^{-j2\pi\frac{k}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi k(2n+1)}{2N} - \pi(2n+1)\right] \\
&= (-1)^{2n+2}e^{-j2\pi\frac{k}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi k(2n+1)}{2N}\right] \\
&= e^{-j2\pi\frac{k}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi k(2n+1)}{2N}\right] \\
&= e^{-j2\pi\frac{k}{4N}} X_{DCT}(k) \\
X_{new}(N) &= e^{j2\pi\frac{N}{4N}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi N(2n+1)}{2N}\right] \\
&= e^{j2\pi\frac{1}{4}} \sum_{n=0}^{N-1} 2x(n)\cos\left[\frac{\pi}{2} + n\pi\right] = 0
\end{aligned}$$

Notice that:

$$\begin{aligned}
[e^{-j2\pi\frac{kn}{2N}} + e^{j2\pi\frac{k(n+1)}{2N}}] &= e^{j2\pi\frac{k}{4N}}[e^{-j2\pi\frac{k(n+\frac{1}{2})}{2N}} + e^{j2\pi\frac{k(n+\frac{1}{2})}{2N}}] \\
&= e^{j2\pi\frac{k}{4N}} 2\cos[2\pi\frac{k(n+\frac{1}{2})}{2N}] \\
&= e^{j2\pi\frac{k}{4N}} 2\cos\left[\frac{\pi k(2n+1)}{2N}\right]
\end{aligned}$$

So, we know that from he inverse of DFT:

$$\begin{aligned}
2Nx_{new}(n) = 2N \times \text{IDFT}[X_{new}(k)] &= \sum_{k=0}^{N-1} X_{new}(k)e^{j2\pi\frac{kn}{2N}} + 0 + \sum_{k=N+1}^{2N-1} X_{new}(k)e^{j2\pi\frac{kn}{2N}} \\
&= \sum_{k=0}^{N-1} X_{new}(k)e^{j2\pi\frac{kn}{2N}} + \sum_{k=1}^{N-1} X_{new}(2N-k)e^{j2\pi\frac{(2N-k)n}{2N}} \\
&= X_{new}(0) + \sum_{k=1}^{N-1}[X_{new}(k)e^{j2\pi\frac{kn}{2N}} + X_{new}(2N-k)e^{j2\pi\frac{(2N-k)n}{2N}}] \\
&= X_{DCT}(0) + \sum_{k=1}^{N-1}[e^{j2\pi\frac{k}{4N}} X_{DCT}(k)e^{j2\pi\frac{kn}{2N}} + e^{-j2\pi\frac{k}{4N}} X_{DCT}(k)e^{-j2\pi\frac{kn}{2N}}] \\
&= X_{DCT}(0) + \sum_{k=1}^{N-1} X_{DCT}(k)[e^{j2\pi(\frac{k}{4N}+\frac{kn}{2N})} + e^{-j2\pi(\frac{k}{4N}+\frac{kn}{2N})}] \\
&= X_{DCT}(0) + \sum_{k=1}^{N-1} X_{DCT}(k) 2\cos\left[\frac{\pi k(2n+1)}{2N}\right]
\end{aligned}$$

Thus, we conclude:

$$\begin{aligned}
x(n) = x_{new}(n) &= \frac{1}{N}[\frac{1}{2}X_{DCT}(0) + \sum_{k=1}^{N-1} X_{DCT}(k)\cos\left[\frac{\pi k(2n+1)}{2N}\right]] \\
&= \frac{1}{N}\sum_{k=0}^{N-1} a(k)X_{\text{DCT}}(k)\cos\left[\frac{\pi k(2n+1)}{2N}\right]
\end{aligned}$$

where

$$\alpha(k) = \begin{cases} 1/2 & k = 0 \\ 1 & 0 < k < N \end{cases}$$

MATLAB Projects

Problem 4.31

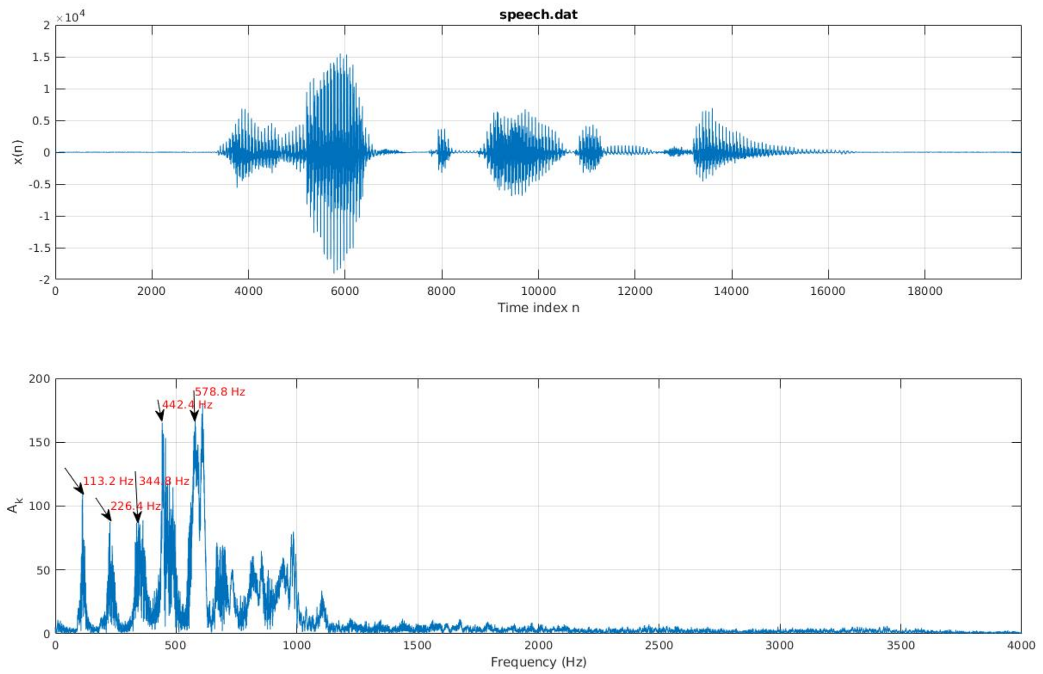
Signal spectral analysis: Given below are four practical signals, compute their one-sided spectra and create their time domain plots and spectral plots, respectively:

1. Speech signal ("speech.dat"), sampling rate = 8000 Hz.

   From the spectral plot, identify the first five formants.
2. ECG signal ("ecg.dat"), sampling rate = 500 Hz. From the spectral plot, identify the 60 Hz-interference component.
3. Seismic data ("seismic.dat"), sampling rate = 15 Hz. From the spectral plot, determine the dominant frequency component.
4. Vibration signal of the acceleration response from a simple supported beam ("vbrdata.dat"), sampling rate = 1000 Hz. From the spectral plot, determine the four dominant frequencies (modes).

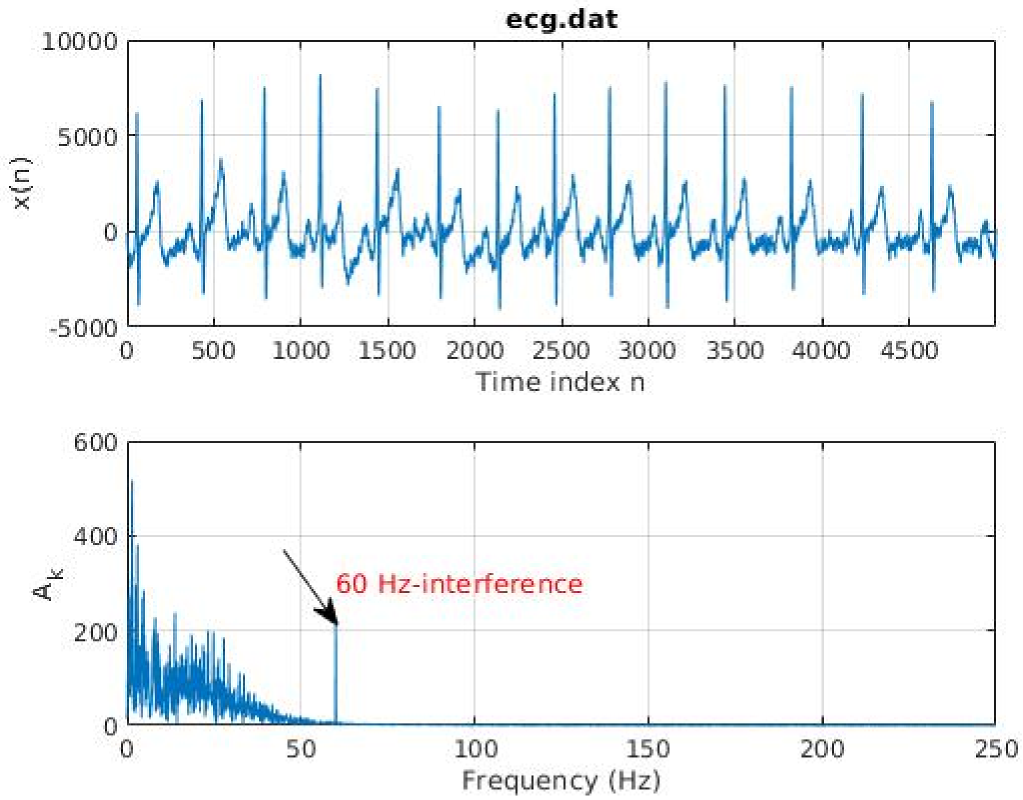
solution

1. Speech signal:



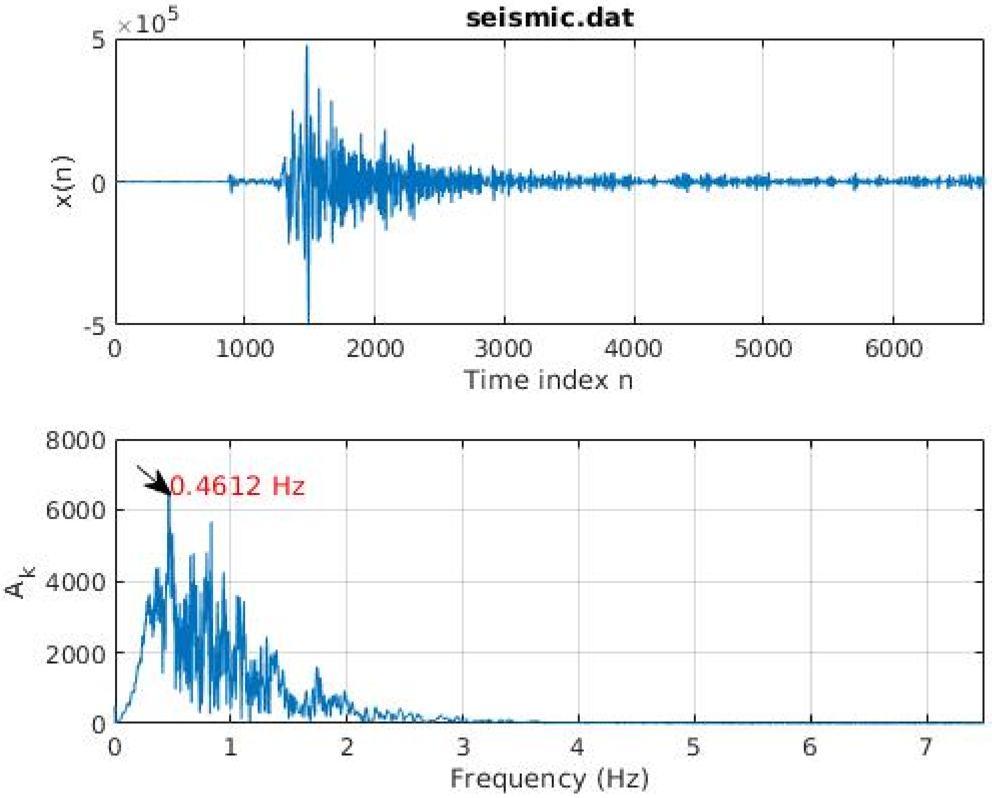
The first five formants are $113.2, 226.4, 344.8, 442.4, 578.8 Hz$.

2. ECG signal:



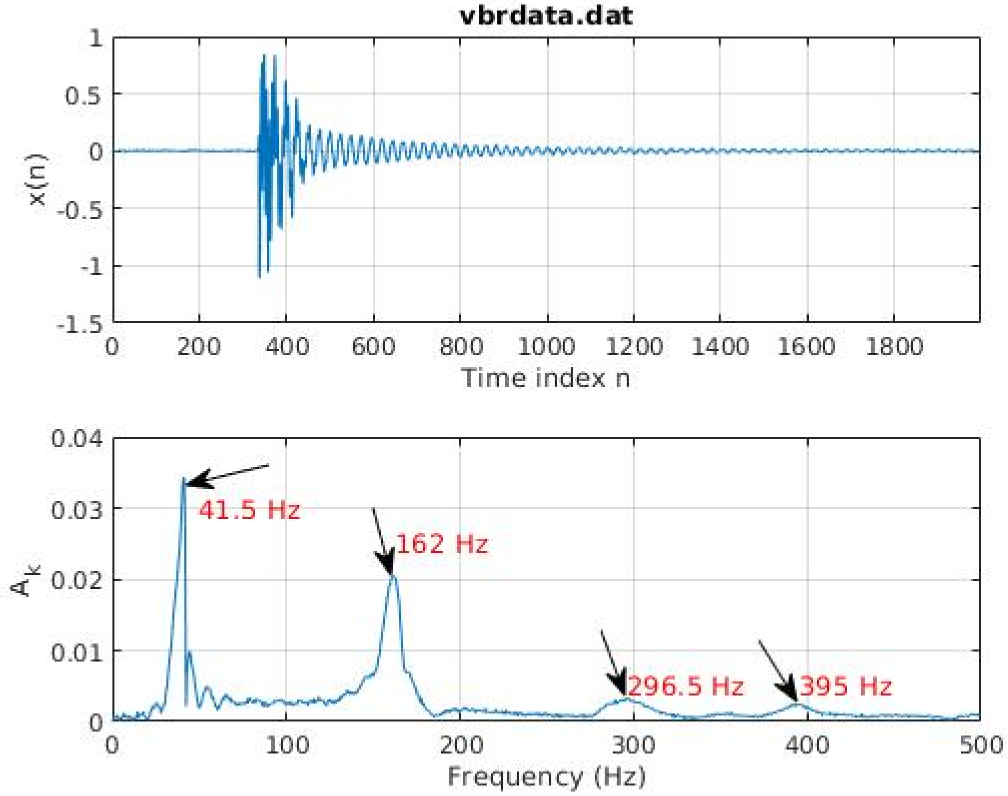
The 60 Hz-interference component is identified by arrow.

3. Seismic signal:



The dominant frequency component is $0.4612Hz$.

4. Vibration signal:



The four dominant frequencies (modes) are $41.5, 162, 296.5, 395Hz$.

MATLAB code as follow:

```
clear; clc; close all
load('speech.dat'); fs_1 = 8000;
load('ecg.dat'); fs_2 = 500;
load('seismic.dat'); fs_3 = 15;
load('vbrdata.dat'); fs_4 = 1000;
%% speech
len = size(speech, 2);
spectrum = fft(speech); list_freq = fs_1 * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(2, 1, 1); plot(0:len-1, speech); grid; xlim([0, len-1])
xlabel('Time index n'); ylabel('x(n)'); title('speech.dat')
subplot(2, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
clear speech len list_freq spectrum amplitude_spectrum
text(113.2, 120, '113.2 Hz', 'Color', 'r'); text(226.4, 100, '226.4 Hz', 'Color', 'r');
text(344.8, 120, '344.8 Hz', 'Color', 'r'); text(442.4, 180, '442.4 Hz', 'Color', 'r');
text(578.8, 190, '578.8 Hz', 'Color', 'r');
%% ecg
len = size(ecg, 2);
spectrum = fft(ecg); list_freq = fs_2 * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len;
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(2, 1, 1); plot(0:len-1, ecg); grid; xlim([0, len-1])
xlabel('Time index n'); ylabel('x(n)'); title('ecg.dat')
subplot(2, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
clear ecg len list_freq spectrum amplitude_spectrum
text(60, 300, '60 Hz-interference', 'Color', 'r');
%% seismic
len = size(seismic, 2);
spectrum = fft(seismic); list_freq = fs_3 * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len;
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(2, 1, 1); plot(0:len-1, seismic); grid; xlim([0, len-1])
xlabel('Time index n'); ylabel('x(n)'); title('seismic.dat')
subplot(2, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
clear seismic len list_freq spectrum amplitude_spectrum
text(0.4612, 6700, '0.4612 Hz', 'Color', 'r');
%% vbrdata
len = size(vbrdata, 2);
spectrum = fft(vbrdata); list_freq = fs_4 * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len;
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(2, 1, 1); plot(0:len-1, vbrdata); grid; xlim([0, len-1])
xlabel('Time index n'); ylabel('x(n)'); title('vbrdata.dat')
subplot(2, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);

xlabel('Frequency (Hz)'); ylabel('A_k');
```

```
clear vbrdata len list_freq spectrum amplitude_spectrum
text(50, 0.03, '41.5 Hz', 'Color', 'r'); text(162, 0.025, '162 Hz', 'Color', 'r');
text(296.5, 0.005, '296.5 Hz', 'Color', 'r'); text(395, 0.005, '395 Hz', 'Color', 'r');
```

Problem 4.32

Vibration signature analysis: The acceleration signals measured from the gearbox can be used for monitoring the condition of the gears inside the gearbox. The early diagnosis of the gear condition can prevent the future catastrophic failure of the system. Given the following measurements and specifications (cour- tesy of SpectraQuest, Inc.):

1. The input shaft has a speed of 1000 rpm and meshing frequency is approximately 300 Hz.
2. Data specifications:

   Sampling rate = 12.8 kHz v0.dat: healthy condition v1.dat: damage severity level 1 (lightly chipped gear) v2.dat: damage severity level 2 (moderately chipped gear) v3.dat: damage severity level 3 (chipped gear) v4.dat: damage severity level 4 (heavily chipped gear) v5.dat: damage severity level 5 (missing tooth) Investigate the spectrum for each measurement and identify sidebands. For each measurement, determine the ratio of the largest sideband amplitude over the amplitude of meshing frequency and investigate the ratio effect related to the damage severity.

solution

The meshing frequency is approximately $f_m \approx 300Hz$.

The input shaft frequency is $f_i = 1000/60 = 16.67Hz$.

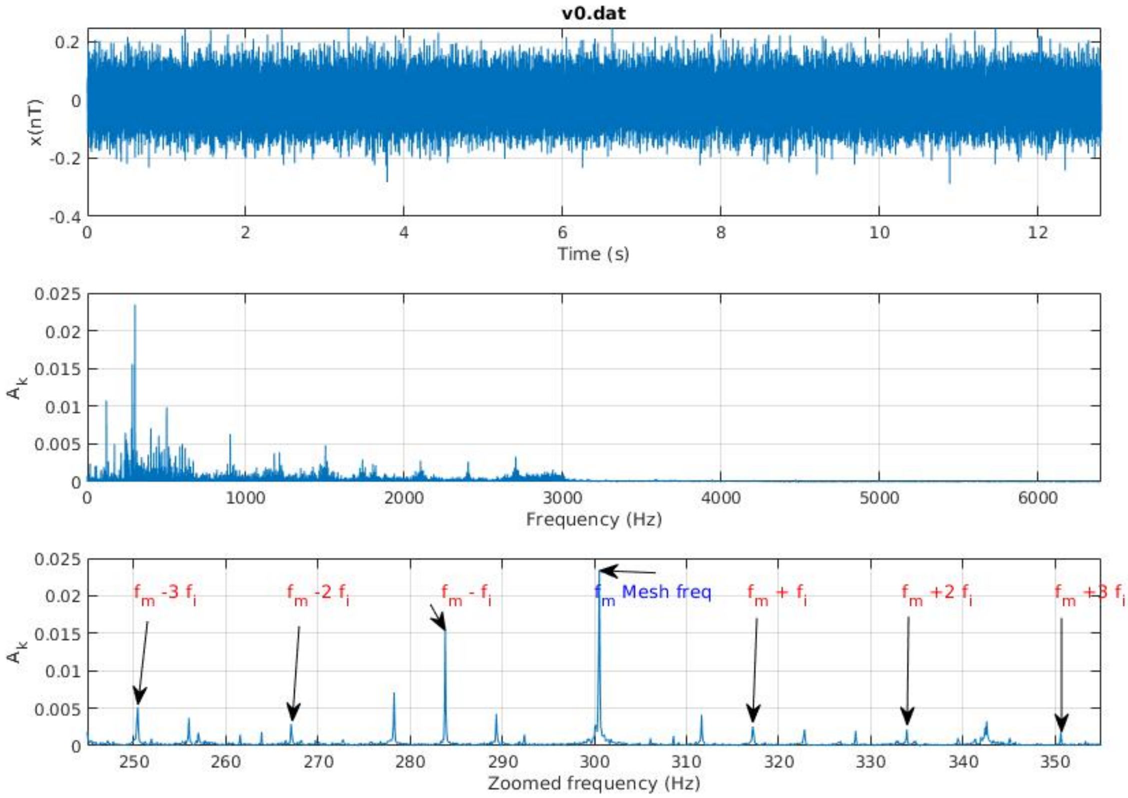
The sidebands are $(f_m \pm f_i, f_m \pm 2f_i \ldots)$.

Now we only consider sidebands $(f_m \pm f_i, f_m \pm 2f_i, f_m \pm 3f_i)$, are approximately:

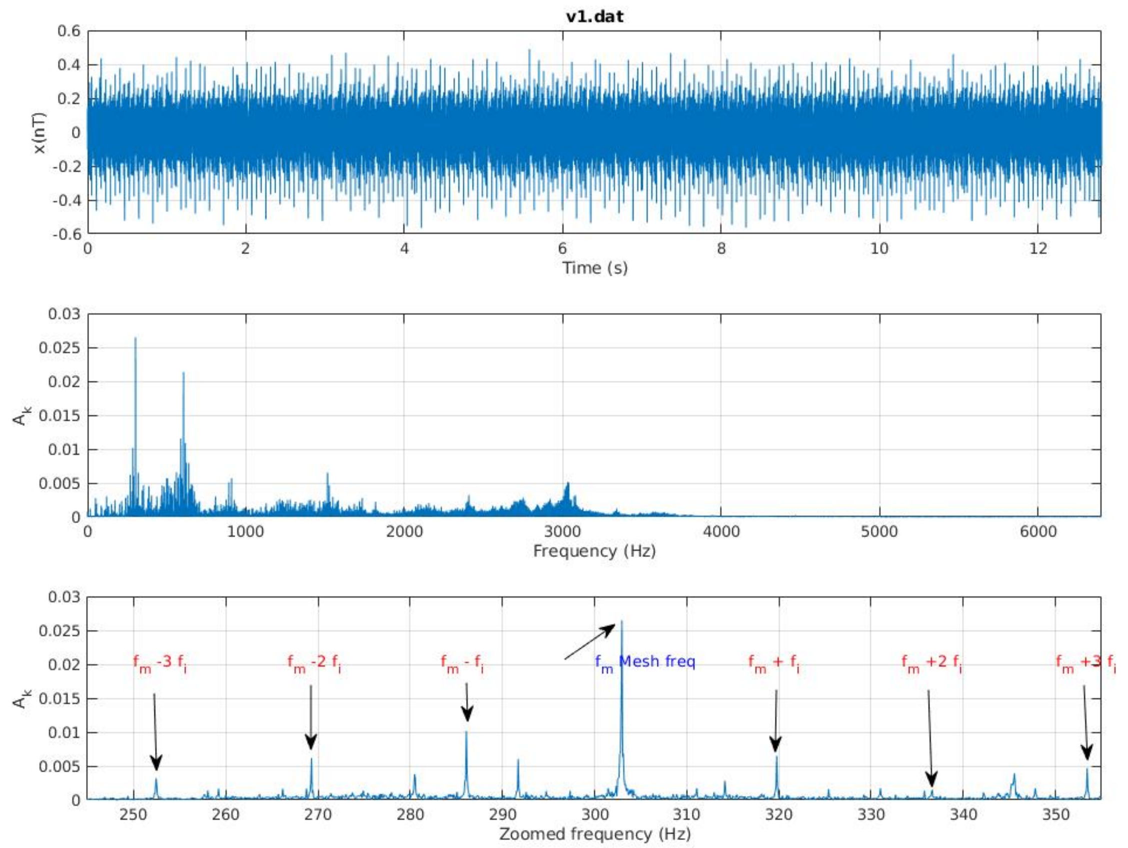
250, 266.67, 283.33, 316.67, 333.33, 350 Hz, so the zoomed frequency range is set to be [245, 355] Hz.

The spectrum for each measurement are demonstrated and sidebands are identified as follows:

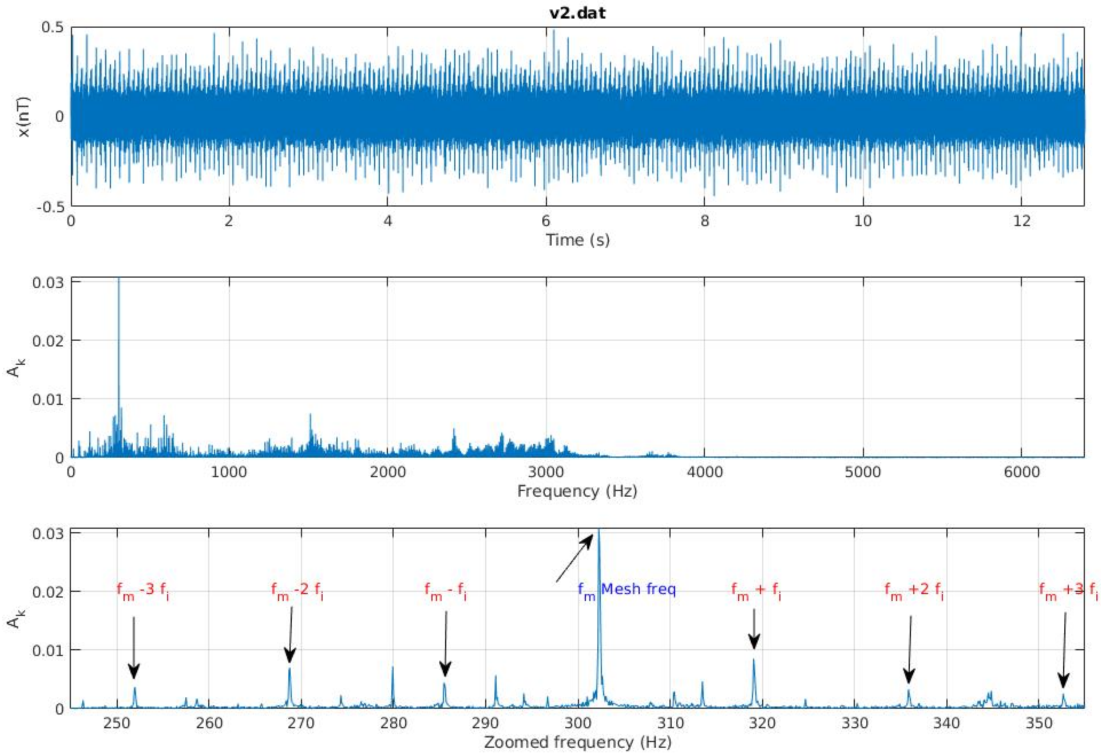
v0.dat: healthy condition



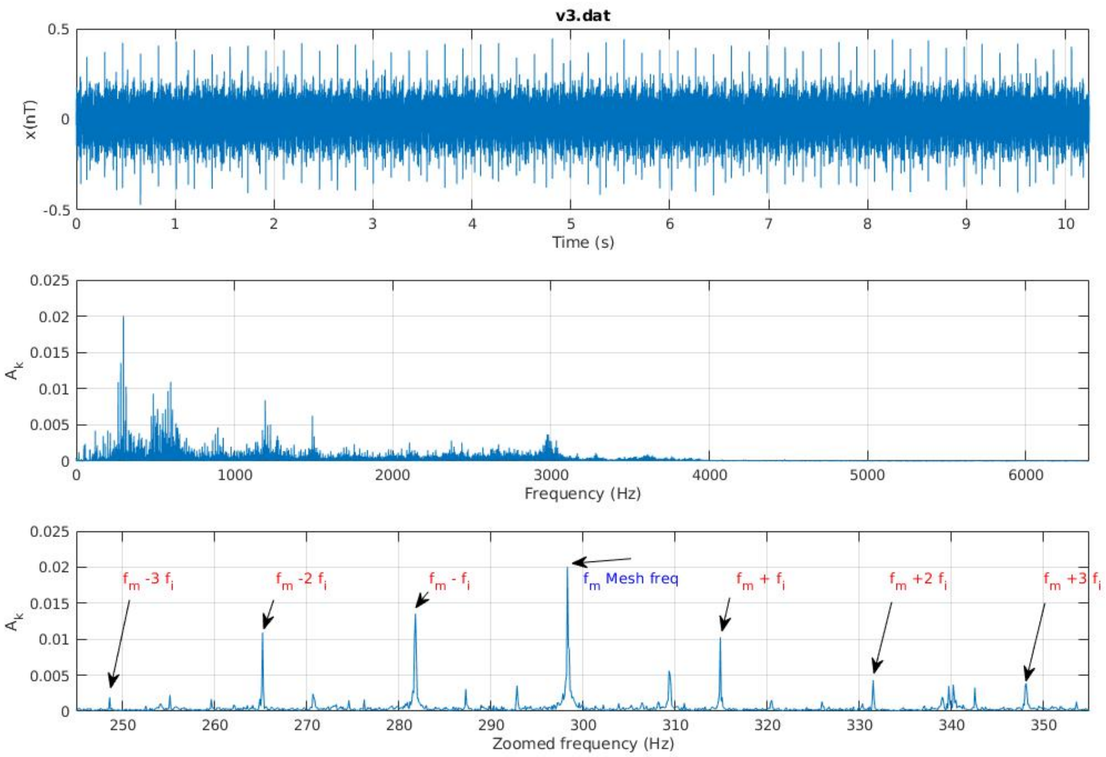
v1.dat: damage severity level 1 (lightly chipped gear)



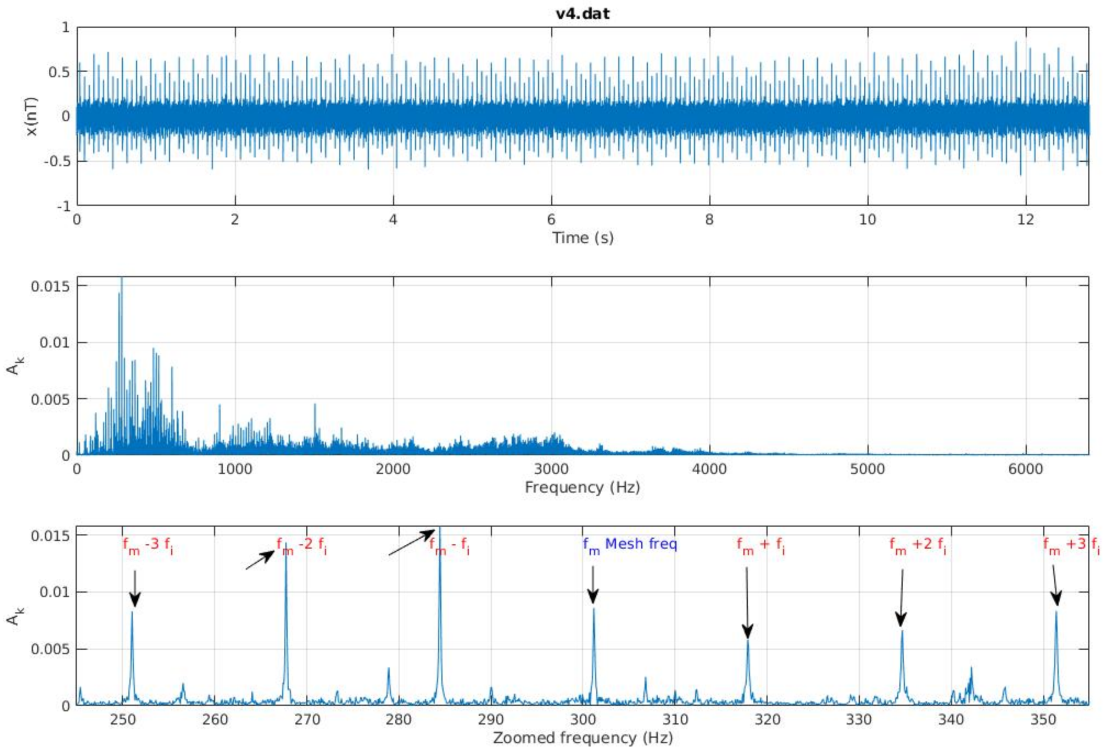
v2.dat: damage severity level 2 (moderately chipped gear)



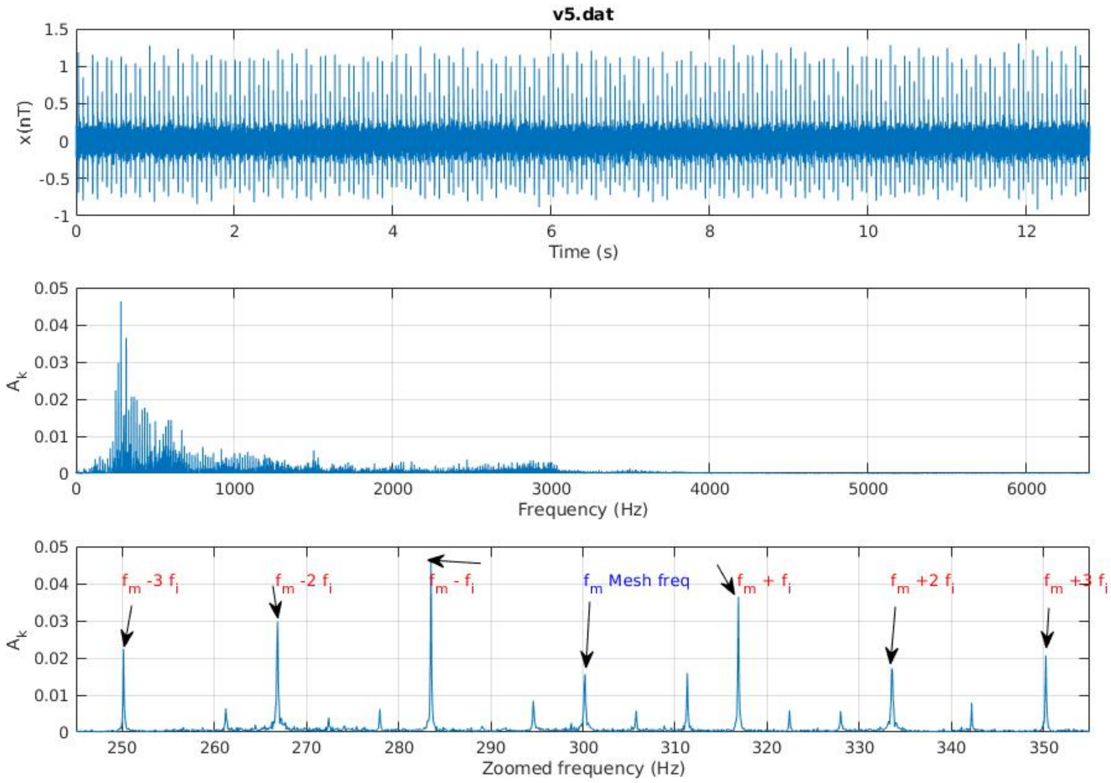
v3.dat: damage severity level 3 (chipped gear)



v4.dat: damage severity level 4 (heavily chipped gear)

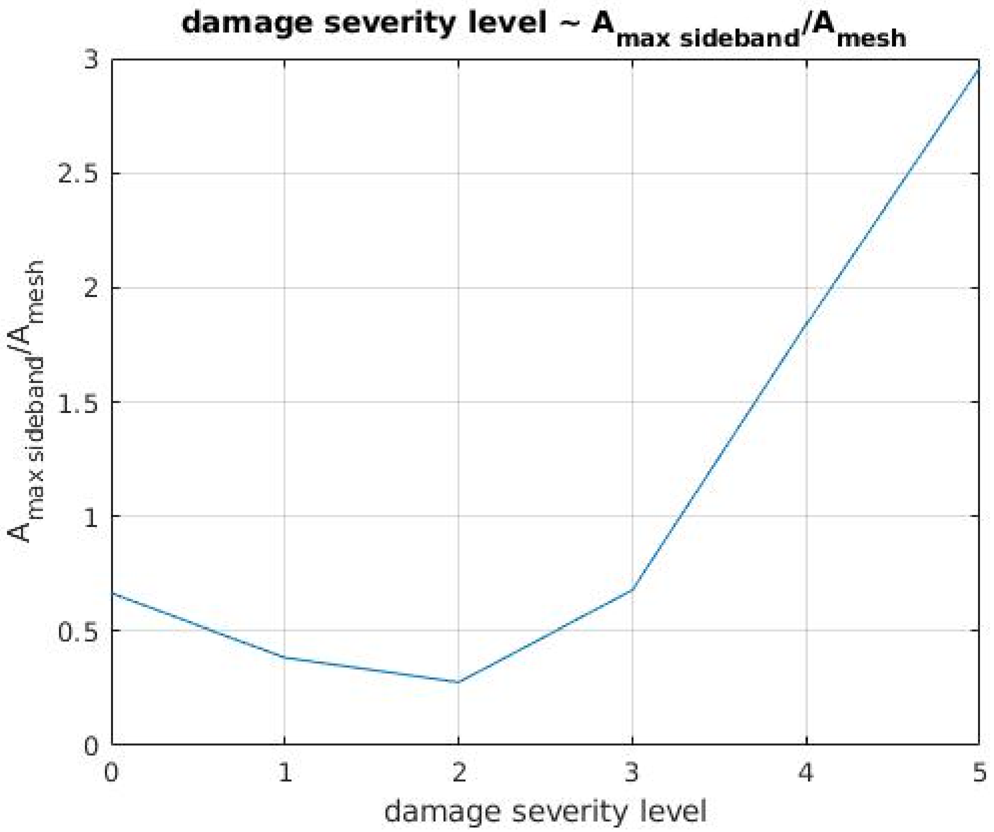


v5.dat: damage severity level 5 (missing tooth)



The ratio of the largest sideband amplitude among $(f_m \pm f_i, f_m \pm 2f_i, f_m \pm 3f_i)$ over the amplitude of meshing frequency:

damage severity level	description	$A_{\text{max sideband}}/A_{\text{mesh}}$
0	healthy condition	0.6648
1	lightly chipped gear	0.3840
2	moderately chipped gear	0.2766
3	chipped gear	0.6767
4	heavily chipped gear	1.8390
5	missing tooth	2.9617



conclusion:

When the damage severity level is very high(=4, 5), the ratio increases sharply, much larger than the value of other damage severity levels(=0,1,2,3).

code:

```
clear; clc; close all
fs = 12.8e3; zoomed_freq_range = [245, 355];
load('v0.dat'); load('v1.dat'); load('v2.dat');
load('v3.dat'); load('v4.dat'); load('v5.dat');
f_mesh_range = [300-4, 300+4]; % find f_mesh in the range
f_i = 16.67; % input shaft freq
offset_sideband = [-3, -2, -1, 1, 2, 3] * f_i;
list_ratio_A = [];
%% v0.dat: healthy condition
len = size(v0, 2);
spectrum = fft(v0); list_freq = fs * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(3, 1, 1); plot((0:len-1) / fs, v0); grid; xlim([0, (len-1)/fs])
xlabel('Time (s)'); ylabel('x(nT)'); title('v0.dat')
subplot(3, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
subplot(3, 1, 3); plot(list_freq, amplitude_spectrum); grid; xlim(zoomed_freq_range);
xlabel('Zoomed frequency (Hz)'); ylabel('A_k');
text(250, 0.02, 'f_m -3 f_i', 'Color', 'r'); text(266.67, 0.02, 'f_m -2 f_i', 'Color', 'r');
text(283.33, 0.02, 'f_m - f_i', 'Color', 'r'); text(316.67, 0.02, 'f_m + f_i', 'Color', 'r');
text(333.33, 0.02, 'f_m +2 f_i', 'Color', 'r'); text(350, 0.02, 'f_m +3 f_i', 'Color', 'r');
text(300, 0.02, 'f_m Mesh freq', 'Color', 'b');
f_mesh_range_ind = int32(f_mesh_range * len / fs); % find f_mesh
[A_mesh, ind_mesh] = max(amplitude_spectrum( f_mesh_range_ind(1):f_mesh_range_ind(end) ) );
f_mesh = double( ind_mesh + f_mesh_range_ind(1) ) * fs / len; f_sideband = offset_sideband +
f_mesh;
list_A_sideband = zeros(1, size(f_sideband, 2) );
for ii=1:size(f_sideband, 2)
    f_sideband_range = [f_sideband(ii)-0.3, f_sideband(ii)+0.3];
    f_sideband_range_ind = int32( f_sideband_range * len / fs );
    [list_A_sideband(ii), ~] = max(amplitude_spectrum(
f_sideband_range_ind(1):f_sideband_range_ind(end) ) );
end
A_sideband_max = max( list_A_sideband );
list_ratio_A = [ list_ratio_A, A_sideband_max / A_mesh];
clear v0 len list_freq spectrum amplitude_spectrum
%% v1.dat: damage severity level 1 (lightly chipped gear)
len = size(v1, 2);
spectrum = fft(v1); list_freq = fs * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(3, 1, 1); plot((0:len-1) / fs, v1); grid; xlim([0, (len-1)/fs])
xlabel('Time (s)'); ylabel('x(nT)'); title('v1.dat')
subplot(3, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
subplot(3, 1, 3); plot(list_freq, amplitude_spectrum); grid; xlim(zoomed_freq_range);
xlabel('Zoomed frequency (Hz)'); ylabel('A_k');
text(250, 0.02, 'f_m -3 f_i', 'Color', 'r'); text(266.67, 0.02, 'f_m -2 f_i', 'Color', 'r');
text(283.33, 0.02, 'f_m - f_i', 'Color', 'r'); text(316.67, 0.02, 'f_m + f_i', 'Color', 'r');

text(333.33, 0.02, 'f_m +2 f_i', 'Color', 'r'); text(350, 0.02, 'f_m +3 f_i', 'Color', 'r');
```

```
text(300, 0.02, 'f_m Mesh freq', 'Color', 'b');
f_mesh_range_ind = int32(f_mesh_range * len / fs); % find f_mesh
[A_mesh, ind_mesh] = max(amplitude_spectrum( f_mesh_range_ind(1):f_mesh_range_ind(end) ) );
f_mesh = double( ind_mesh + f_mesh_range_ind(1) ) * fs / len; f_sideband = offset_sideband +
f_mesh;
list_A_sideband = zeros(1, size(f_sideband, 2) );
for ii=1:size(f_sideband, 2)
    f_sideband_range = [f_sideband(ii)-0.3, f_sideband(ii)+0.3];
    f_sideband_range_ind = int32( f_sideband_range * len / fs );
    [list_A_sideband(ii), ~] = max(amplitude_spectrum(
f_sideband_range_ind(1):f_sideband_range_ind(end) ) );
end
A_sideband_max = max( list_A_sideband );
list_ratio_A = [ list_ratio_A, A_sideband_max / A_mesh];
clear v1 len list_freq spectrum amplitude_spectrum
%% v2.dat: damage severity level 2 (moderately chipped gear)
len = size(v2, 2);
spectrum = fft(v2); list_freq = fs * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure(); % plot
subplot(3, 1, 1); plot((0:len-1) / fs, v2); grid; xlim([0, (len-1)/fs])
xlabel('Time (s)'); ylabel('x(nT)'); title('v2.dat')
subplot(3, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
subplot(3, 1, 3); plot(list_freq, amplitude_spectrum); grid; xlim(zoomed_freq_range);
xlabel('Zoomed frequency (Hz)'); ylabel('A_k');
text(250, 0.02, 'f_m -3 f_i', 'Color', 'r'); text(266.67, 0.02, 'f_m -2 f_i', 'Color', 'r');
text(283.33, 0.02, 'f_m - f_i', 'Color', 'r'); text(316.67, 0.02, 'f_m + f_i', 'Color', 'r');
text(333.33, 0.02, 'f_m +2 f_i', 'Color', 'r'); text(350, 0.02, 'f_m +3 f_i', 'Color', 'r');
text(300, 0.02, 'f_m Mesh freq', 'Color', 'b');
f_mesh_range_ind = int32(f_mesh_range * len / fs); % find f_mesh
[A_mesh, ind_mesh] = max(amplitude_spectrum( f_mesh_range_ind(1):f_mesh_range_ind(end) ) );
f_mesh = double( ind_mesh + f_mesh_range_ind(1) ) * fs / len; f_sideband = offset_sideband +
f_mesh;
list_A_sideband = zeros(1, size(f_sideband, 2) );
for ii=1:size(f_sideband, 2)
    f_sideband_range = [f_sideband(ii)-0.3, f_sideband(ii)+0.3];
    f_sideband_range_ind = int32( f_sideband_range * len / fs );
    [list_A_sideband(ii), ~] = max(amplitude_spectrum(
f_sideband_range_ind(1):f_sideband_range_ind(end) ) );
end
A_sideband_max = max( list_A_sideband );
list_ratio_A = [ list_ratio_A, A_sideband_max / A_mesh];
clear v2 len list_freq spectrum amplitude_spectrum
%% v3.dat: damage severity level 3 (chipped gear)
len = size(v3, 2);
spectrum = fft(v3); list_freq = fs * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure(); % plot
subplot(3, 1, 1); plot((0:len-1) / fs, v3); grid; xlim([0, (len-1)/fs])

xlabel('Time (s)'); ylabel('x(nT)'); title('v3.dat')
```

```
subplot(3, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
subplot(3, 1, 3); plot(list_freq, amplitude_spectrum); grid; xlim(zoomed_freq_range);
xlabel('Zoomed frequency (Hz)'); ylabel('A_k');
text(250, 0.018, 'f_m -3 f_i', 'Color', 'r'); text(266.67, 0.018, 'f_m -2 f_i', 'Color',
'r');
text(283.33, 0.018, 'f_m - f_i', 'Color', 'r'); text(316.67, 0.018, 'f_m + f_i', 'Color',
'r');
text(333.33, 0.018, 'f_m +2 f_i', 'Color', 'r'); text(350, 0.018, 'f_m +3 f_i', 'Color',
'r');
text(300, 0.018, 'f_m Mesh freq', 'Color', 'b');
f_mesh_range_ind = int32(f_mesh_range * len / fs); % find f_mesh
[A_mesh, ind_mesh] = max(amplitude_spectrum( f_mesh_range_ind(1):f_mesh_range_ind(end) ) );
f_mesh = double( ind_mesh + f_mesh_range_ind(1) ) * fs / len; f_sideband = offset_sideband +
f_mesh;
list_A_sideband = zeros(1, size(f_sideband, 2) );
for ii=1:size(f_sideband, 2)
    f_sideband_range = [f_sideband(ii)-0.3, f_sideband(ii)+0.3];
    f_sideband_range_ind = int32( f_sideband_range * len / fs );
    [list_A_sideband(ii), ~] = max(amplitude_spectrum(
f_sideband_range_ind(1):f_sideband_range_ind(end) ) );
end
A_sideband_max = max( list_A_sideband );
list_ratio_A = [ list_ratio_A, A_sideband_max / A_mesh];
clear v3 len list_freq spectrum amplitude_spectrum
%% v4.dat: damage severity level 4 (heavily chipped gear)
len = size(v4, 2);
spectrum = fft(v4); list_freq = fs * (0:len/2) / len;
amplitude_spectrum = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum(1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(3, 1, 1); plot((0:len-1) / fs, v4); grid; xlim([0, (len-1)/fs])
xlabel('Time (s)'); ylabel('x(nT)'); title('v4.dat')
subplot(3, 1, 2); plot(list_freq, amplitude_spectrum); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)'); ylabel('A_k');
subplot(3, 1, 3); plot(list_freq, amplitude_spectrum); grid; xlim(zoomed_freq_range);
xlabel('Zoomed frequency (Hz)'); ylabel('A_k');
text(250, 0.014, 'f_m -3 f_i', 'Color', 'r'); text(266.67, 0.014, 'f_m -2 f_i', 'Color',
'r');
text(283.33, 0.014, 'f_m - f_i', 'Color', 'r'); text(316.67, 0.014, 'f_m + f_i', 'Color',
'r');
text(333.33, 0.014, 'f_m +2 f_i', 'Color', 'r'); text(350, 0.014, 'f_m +3 f_i', 'Color',
'r');
text(300, 0.014, 'f_m Mesh freq', 'Color', 'b');
f_mesh_range_ind = int32(f_mesh_range * len / fs); % find f_mesh
[A_mesh, ind_mesh] = max(amplitude_spectrum( f_mesh_range_ind(1):f_mesh_range_ind(end) ) );
f_mesh = double( ind_mesh + f_mesh_range_ind(1) ) * fs / len; f_sideband = offset_sideband +
f_mesh;
list_A_sideband = zeros(1, size(f_sideband, 2) );
for ii=1:size(f_sideband, 2)
    f_sideband_range = [f_sideband(ii)-0.3, f_sideband(ii)+0.3];
    f_sideband_range_ind = int32( f_sideband_range * len / fs );

    [list_A_sideband(ii), ~] = max(amplitude_spectrum(
```

```
f_sideband_range_ind (1):f_sideband_range_ind (end) ) );
end
A_sideband_max  = max( list_A_sideband  );
list_ratio_A = [ list_ratio_A, A_sideband_max / A_mesh];
clear v4 len list_freq spectrum amplitude_spectrum
%% v5.dat: damage severity level 5 (missing tooth)
len = size(v5, 2);
spectrum = fft(v5); list_freq = fs * (0:len/2) / len;
amplitude_spectrum  = 2 * abs( spectrum(1:len/2+1) ) / len; % MATLAB index starts from 1
amplitude_spectrum (1) = abs( spectrum(1) ) / len;
figure();  % plot
subplot(3, 1, 1); plot((0:len-1) / fs, v5); grid; xlim([0, (len-1)/fs])
xlabel('Time (s)'); ylabel('x(nT)'); title('v5.dat')
subplot(3, 1, 2); plot(list_freq, amplitude_spectrum ); grid; xlim([0, list_freq(end)]);
xlabel('Frequency (Hz)' ); ylabel('A_k');
subplot(3, 1, 3); plot(list_freq, amplitude_spectrum ); grid; xlim(zoomed_freq_range );
xlabel('Zoomed frequency (Hz)' ); ylabel('A_k');
text(250, 0.04, 'f_m -3 f_i' , 'Color', 'r'); text(266.67, 0.04, 'f_m -2 f_i' , 'Color', 'r');
text(283.33, 0.04, 'f_m - f_i' , 'Color', 'r'); text(316.67, 0.04, 'f_m + f_i' , 'Color', 'r');
text(333.33, 0.04, 'f_m +2 f_i' , 'Color', 'r'); text(350, 0.04, 'f_m +3 f_i' , 'Color', 'r');
text(300, 0.04, 'f_m Mesh freq' , 'Color', 'b');
f_mesh_range_ind  = int32(f_mesh_range * len / fs); % find f_mesh
[A_mesh, ind_mesh] = max(amplitude_spectrum ( f_mesh_range_ind (1):f_mesh_range_ind (end) ) );
f_mesh  = double( ind_mesh + f_mesh_range_ind (1) ) * fs / len; f_sideband  = offset_sideband  +
f_mesh;
list_A_sideband  = zeros(1, size(f_sideband , 2) );
for ii=1:size(f_sideband , 2)
    f_sideband_range  = [f_sideband (ii)-0.3, f_sideband (ii)+0.3];
    f_sideband_range_ind  = int32( f_sideband_range  * len / fs );
    [list_A_sideband (ii), ~] = max(amplitude_spectrum (
f_sideband_range_ind (1):f_sideband_range_ind (end) ) );
end
A_sideband_max  = max( list_A_sideband  );
list_ratio_A = [ list_ratio_A, A_sideband_max / A_mesh];
clear v5 len list_freq spectrum amplitude_spectrum
%% plot the ratio of the largest sideband amplitude over the amplitude of meshing frequency
figure(); plot(0:length(list_ratio_A )-1, list_ratio_A ); grid;
xlabel('damage severity level' ); ylabel('A_{max sideband}/A_{mesh}' );
title('damage severity level ~ A_{max sideband}/A_{mesh}'  );
list_ratio_A
```