Digital Signal Processing Lab

Experiment 4

By Hardik Tibrewal (18EC10020)

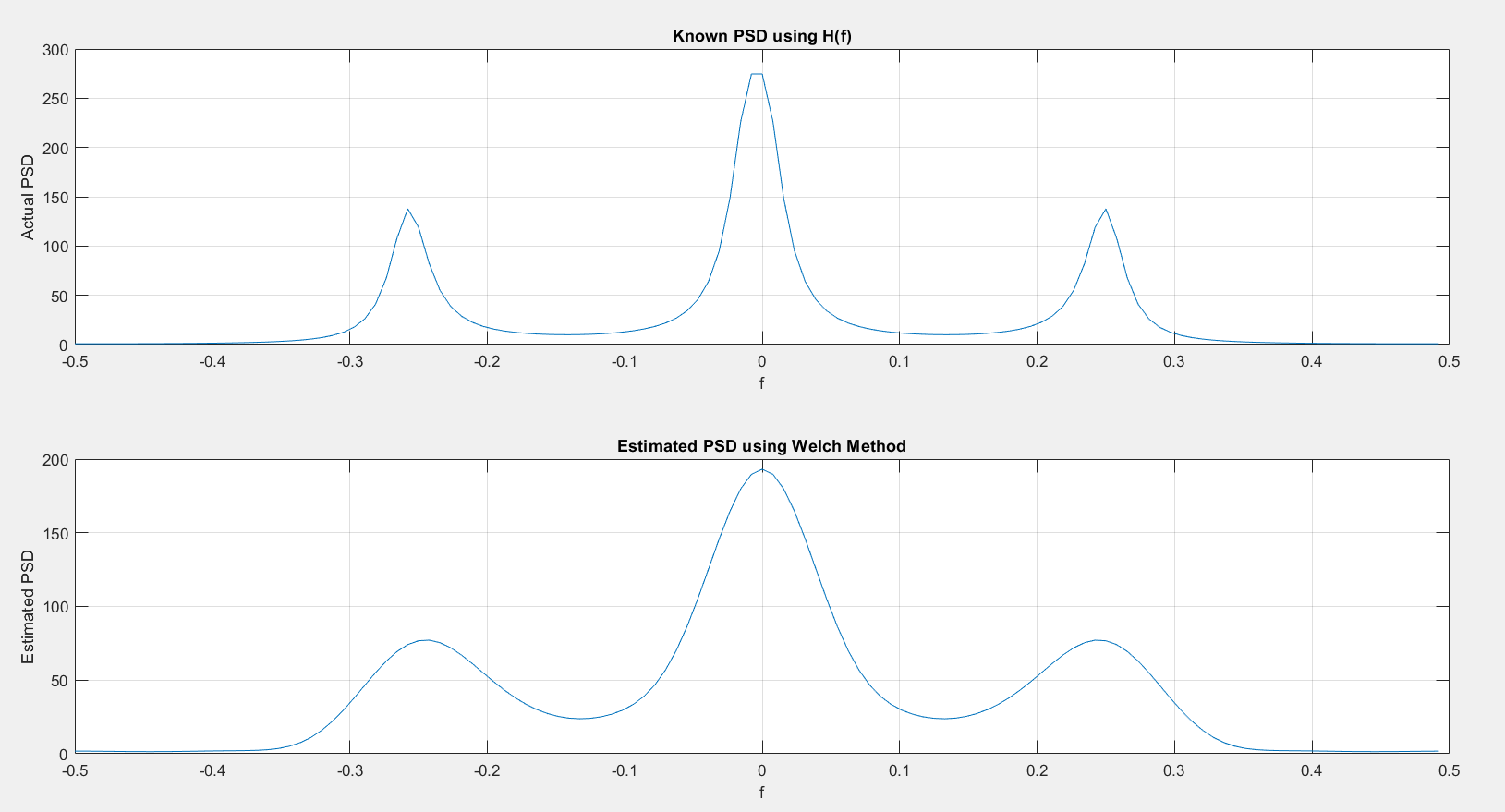
**Aim:**

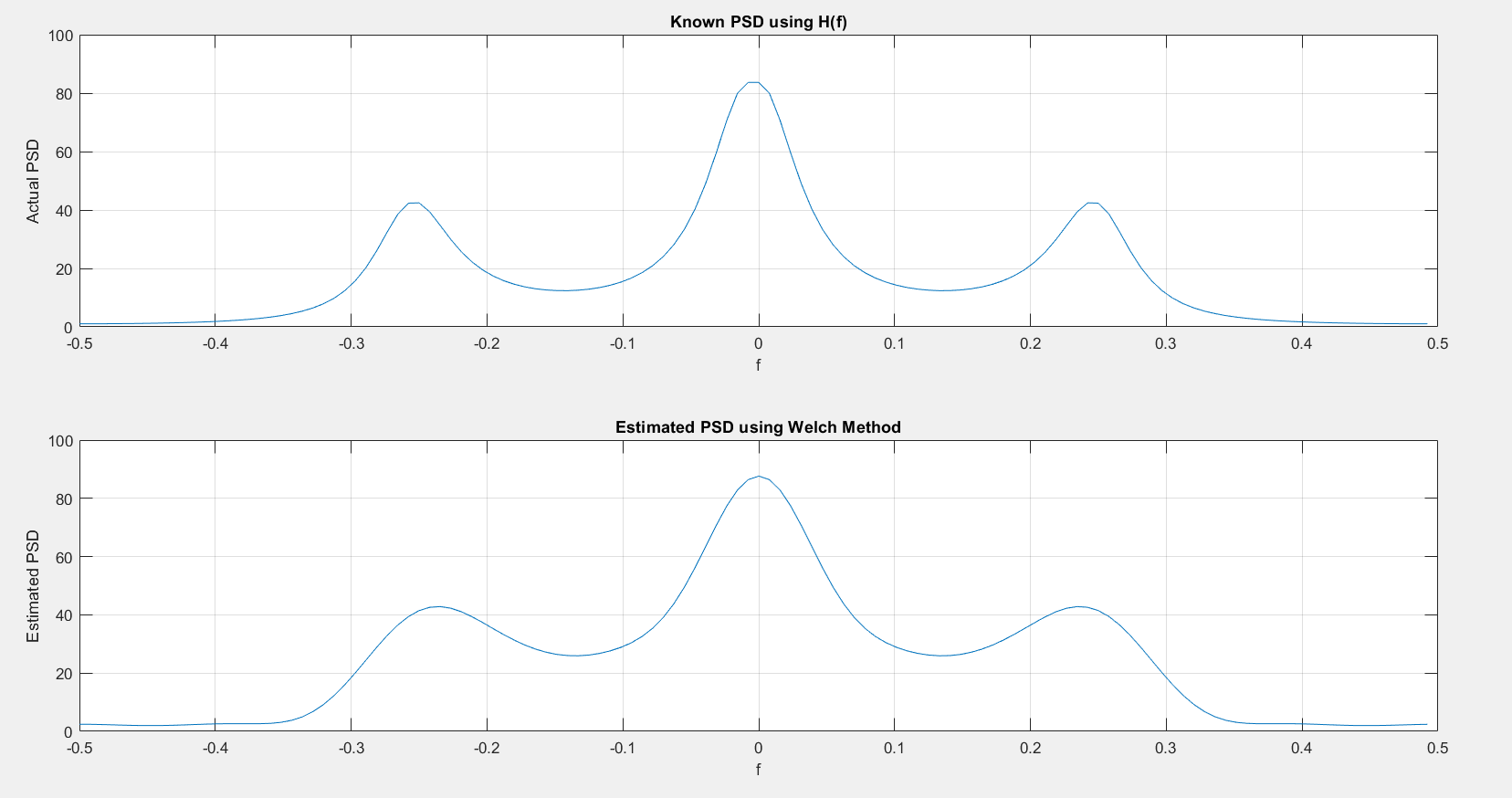
Power Spectrum Estimation using Welch Method

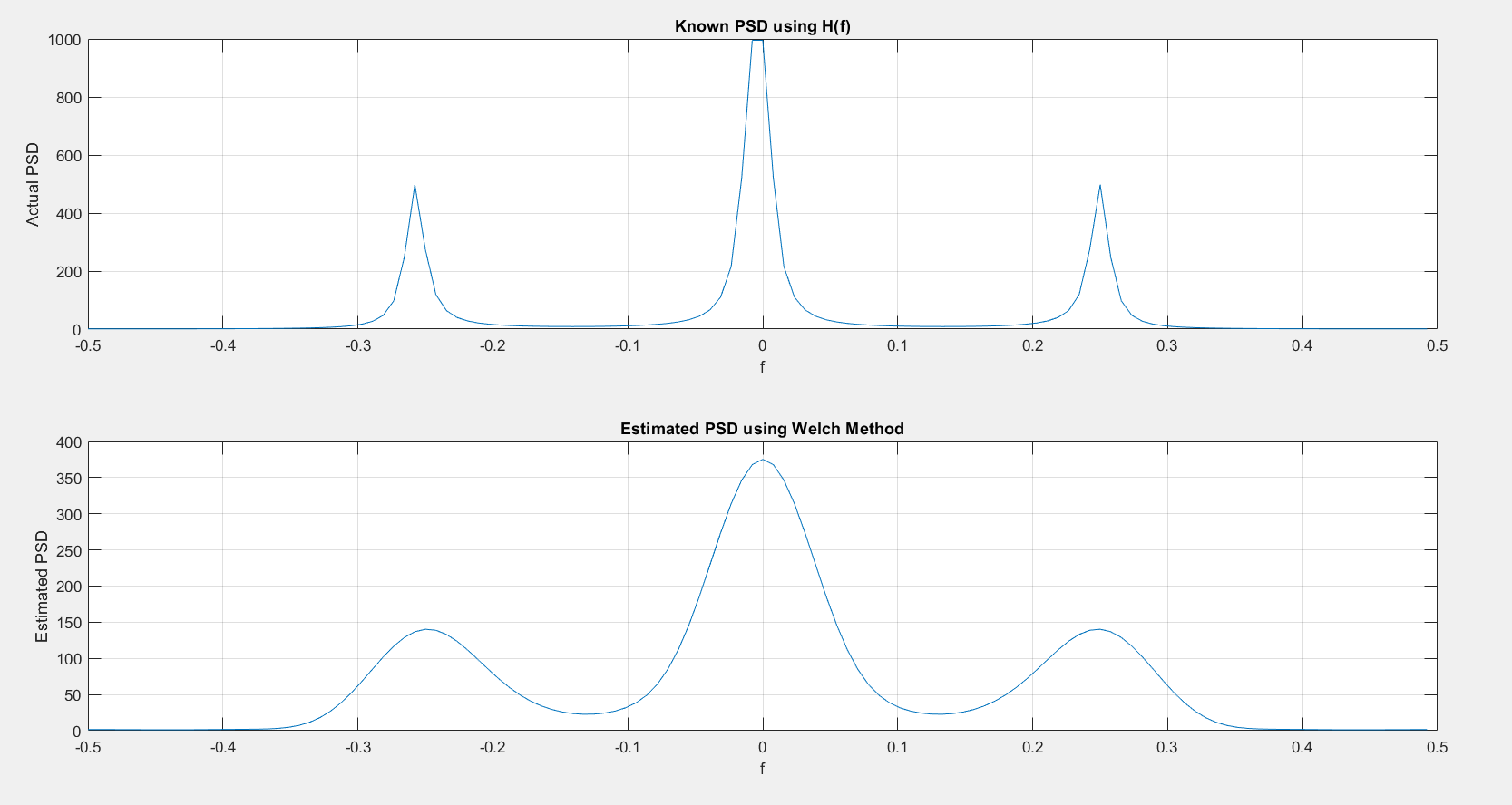
**Plots:**

Denominator coefficients corresponding to [z0, z-1, z-2, z-3] are mentioned above each figure. Numerator is always 1.

Non-overlapping blocks here.

A = [1, -0.9, 0.81, -0.729]

A = [1, -0.8, 0.64, -0.512]

A = [1, -0.95, 0.9025, -0.8574]

(Due to network issues, overlapping part could not be done today.)

**Code:**

clc

clear all

close all

mean = 0;

std\_dev = 3;

N = 128;

rng('default');

noise = std\_dev.\*randn(N,1) + mean;

denom = 0.9;

A = [1, -denom, denom^2, -denom^3];

X = filter(1, A, noise);

L = 8;

M = 16;

D = 0;

X\_divs = zeros(L, M);

for ii = 1:L

X\_divs(ii,:) = X((1+(ii-1)\*M):(M+(ii-1)\*M));

end

n = 0:1:(M-1);

hamming = 0.54 - 0.46\*cos(2\*pi\*n/(M-1));

U = (1/M)\*sum(hamming.\*hamming);

P\_n = zeros(L, M);

for ii = 1:L

P\_n(ii,:) = X\_divs(ii,:).\*hamming;

end

f = -0.5:1/N:(0.5-(1/N));

cosine = 0; sine = 0;

P\_f = zeros(L, N);

for ii = 1:L

for F = 1:length(f)

cosine = 0; sine = 0;

for jj = 1:M

cosine = cosine + cos(2\*pi\*f(F)\*jj)\*P\_n(ii,jj);

sine = sine + sin(2\*pi\*f(F)\*jj)\*P\_n(ii,jj);

end

idx = floor((N - length(f))/2)+F;

P\_f(ii, idx) = (cosine^2 + sine^2)/(M\*U);

end

end

Pw\_f = zeros(1, N);

for ii = 1:L

Pw\_f = Pw\_f + P\_f(ii,:);

end

Pw\_f = Pw\_f./L;

[H, W] = freqz(1,A,N/2);

figure();

subplot(211);

l1 = (abs(H).^2).\*std\_dev^2;

l2 = flip(l1);

l = [l2' l1'];

plot(f, l);

grid on;

xlabel('f');ylabel("Actual PSD");title("Known PSD using H(f)");

subplot(212);

plot(f, Pw\_f);

grid on;

xlabel('f');ylabel("Estimated PSD");title("Estimated PSD using Welch Method");