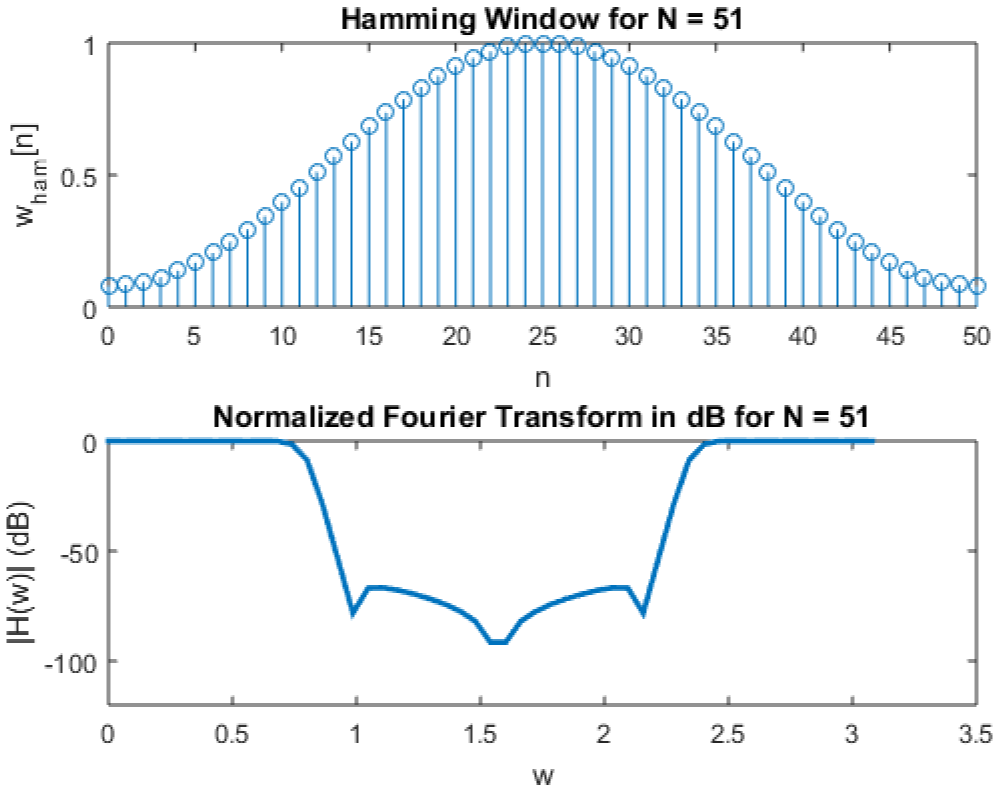
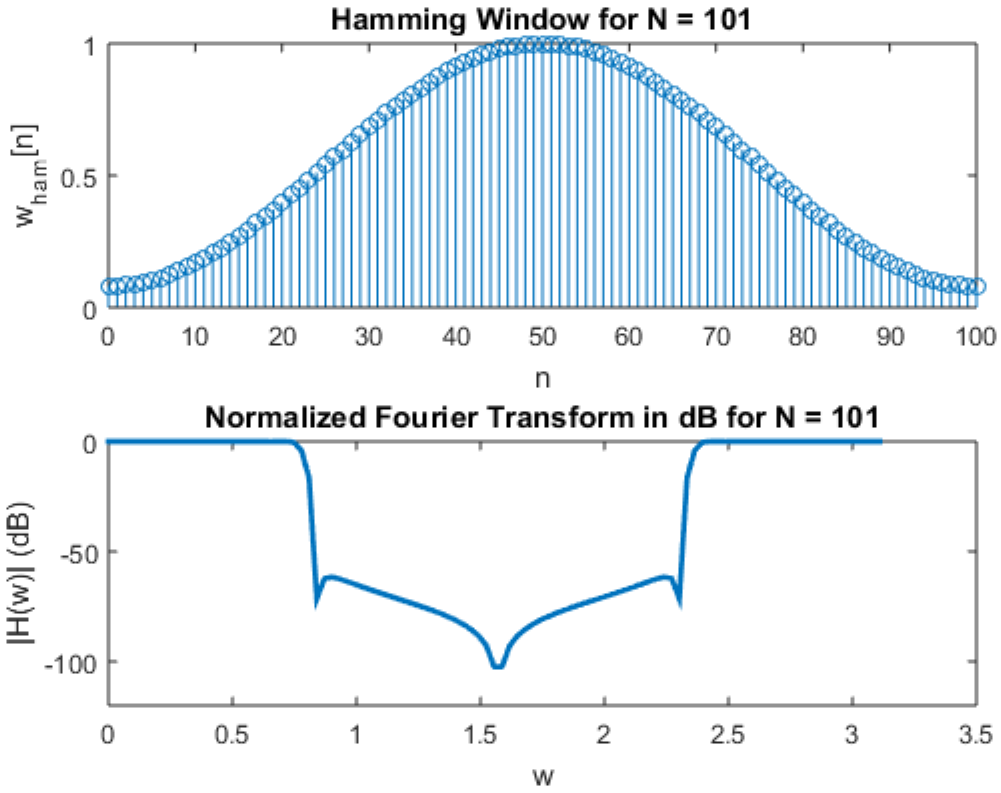
Hamming Window

Table of Contents

For N = 51, 101, 201

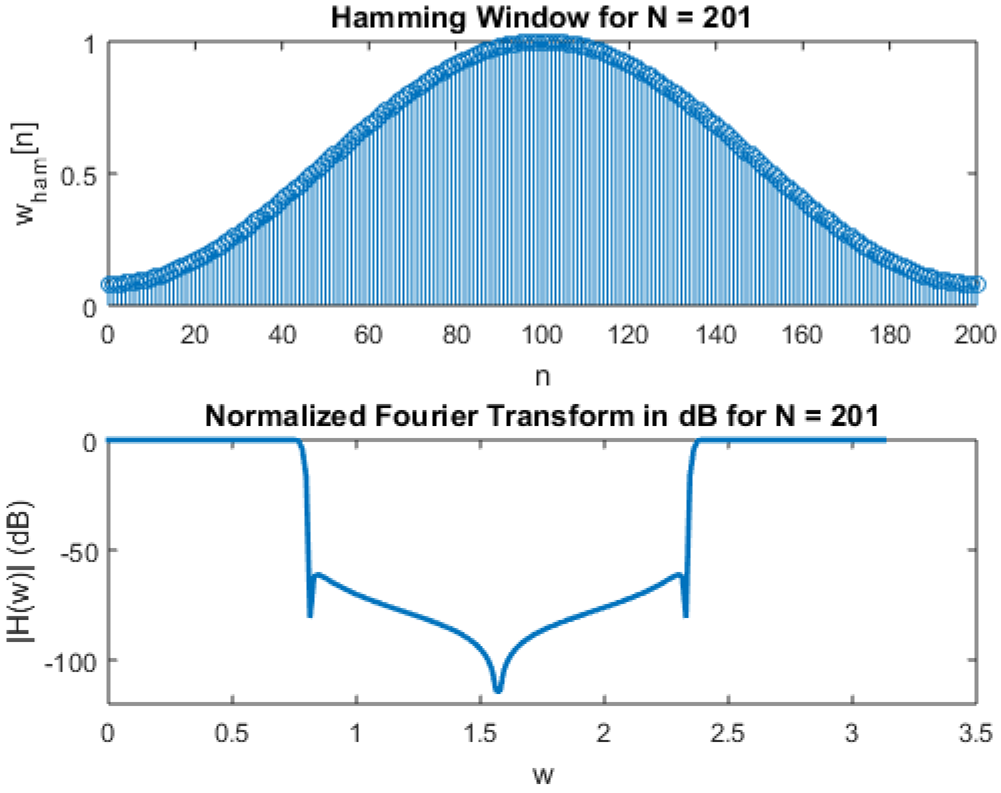
```
for i = 1:3

    switch(i)
        case 1
            N = 51;
        case 2
            N = 101;
        case 3
            N = 201;
    end

    m = 0:N-1;
    w = pi*(0:N-1)/N;

    wc = pi/2;
    a = (N-1)/2;

    w_ham = zeros(1,N);
    n = 0;
    for k = 1:N
        if (n >= 0 && n <= (N-1))
            w_ham(k) = 0.54 - 0.46*cos(2*pi*n/(N-1));
        else
            w_ham(k) = 0;
        end
        n = n + 1;
    end

    h_d = zeros(1,N);
    n = 0;
    for k = 1:N
        if (n == (N-1)/2)
            h_d(k) = 1/2;
        else
            h_d(k) = sin(wc*(n-a))/(pi*(n-a));
        end
        n = n + 1;
    end
    h = h_d.*w_ham;

    H = four_tran(h,N);
    H_norm = abs(H)/abs(max(H));
    H_dB = 20*log10(abs(H_norm));

    figure,
    subplot(211), stem(m,w_ham), xlabel('n'), ylabel('w_h_a_m[n]'),
 title(['Hamming Window for N = ' num2str(N)]);
    subplot(212), plot(w,H_dB, 'linewidth', 2), ylim([-120 0]),
 xlabel('w'), ylabel('|H(w)| (dB)'), title(['Normalized Fourier
 Transform in dB for N = ' num2str(N)]);

end
```





Hamming Window for N = 101

Normalized Fourier Transform in dB for N = 101



Hamming Window for N = 201

Normalized Fourier Transform in dB for N = 201

Function for Fourier Transform

```
function [y] = four_tran(x,N)

    y = zeros(1,N);
    L = length(x);

    for m = 1:N
            y(m) = 0;
        for n = 1:L
            y(m) = y(m) + x(n).*exp(-1j.*2.*pi.*(n-1).*(m-1)./L);
        end
    end

end
```

Credits

```
Made by:
Muhammad Abdullah
(2015-EE-166)
```

Published with MATLAB® R2015a