**7.2 Phương pháp cửa sổ**

**7.2.1 Cửa sổ chữ nhật**

Trong miền cửa sổ chữ nhật được định nghĩa như sau:

Hình 7.2.1.1 dưới đây cho ta đồ thị của cửa sổ chữ nhật.

Hình 7.2.1.1 Đồ thị cửa sổ hình chữ nhật

Từ trên ta có thể thấy rằng cửa sổ chữ nhật cũng chính là dãy chữ nhật .

Cửa sổ này được sử dụng để hạn chế chiều dài của lý tưởng có pha tuyến tính. Cửa sổ là đối xứng, tâm đối xứng nằm tại ( lẻ) vậy trong miền tần số sẽ có pha tuyến tính là (FIR loại 1) thì trong miền , và sẽ có cùng tâm đối xứng tại ( lẻ). Như vậy phiên bản chiều dài hữu hạn của chiều dài vô hạn sẽ cho bởi biểu thức sau đây:

Tương đương:

**7.2.2 Cửa sổ tam giác (cửa sổ Bartlett)**

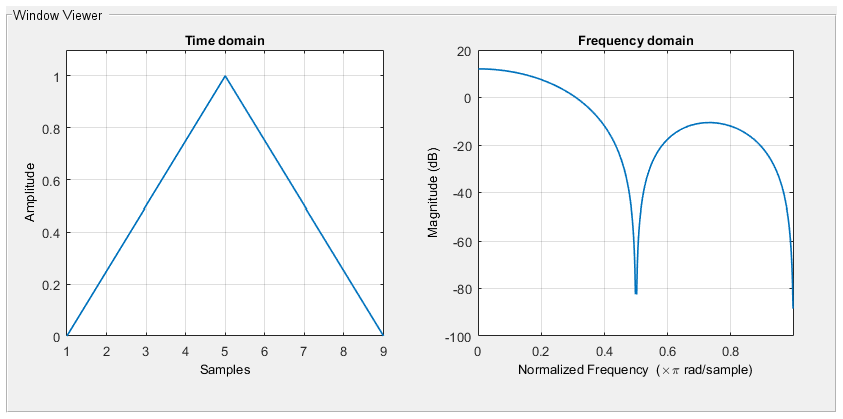
Với mục đích giảm biên độ của các đỉnh thứ cấp của cửa sổ chữ nhật , chung sta chọn một cửa sổ khác có dạng đồ thị tam giác cân, gọi là cửa sổ tam giác (cửa sổ Bartlett).

Trong miền cửa sổ tam giác được định nghĩa như sau:

*Ví dụ 7.2.2.1:* Biểu diễn đồ thị cửa sổ tam giác với .

Giải

Cửa sổ tam giác :



Hình 7.2.2.1 Đồ thị cửa sổ tam giác trên miền thời gian và đồ thị tần số

*Ví dụ 7.2.2.2:* Hãy tổng hợp bộ lọc thông thấp FIR pha tuyến tính dùng cửa sổ tam giác với , .

L = 9;

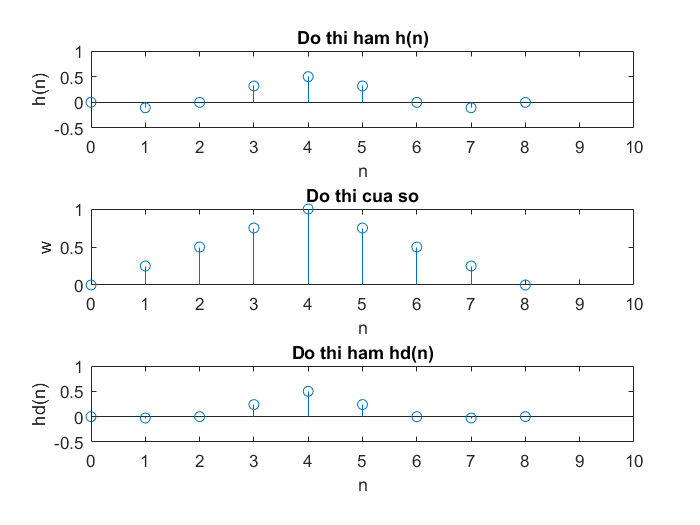
bw = bartlett(L);

wvtool(bw)

Giải

Bộ lọc thông thấp hữu hạn được tổng hợp theo biểu thức:

Thu được kết quả:



Hình 7.2.2.2 Đồ thị tín hiệu , và

wc = 0.5\*pi;

N = 9;

phi=(N - 1)/2;

n = 0:N-1;

w\_bart = bartlett(N);

h = (wc/pi)\*sin(wc\*(n-phi))./(wc\*(n-phi));

h(1,5) = 0.5;

hd = h.\*(w\_bart.');

subplot(3,1,1);

stem(n,h); title('Do thi ham h(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('h(n)');

subplot(3,1,2);

stem(n,w\_bart); title('Do thi cua so');

axis([0 10 0 1]); xlabel('n'); ylabel('w');

subplot(3,1,3);

stem(n,hd); title('Do thi ham hd(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('hd(n)');

**7.2.3 Cửa sổ Hanning và Hamming**

Trong miền cửa sổ Hanning và Hamming được định nghĩa như sau:

Nếu , ta có cửa sổ Hanning như sau:

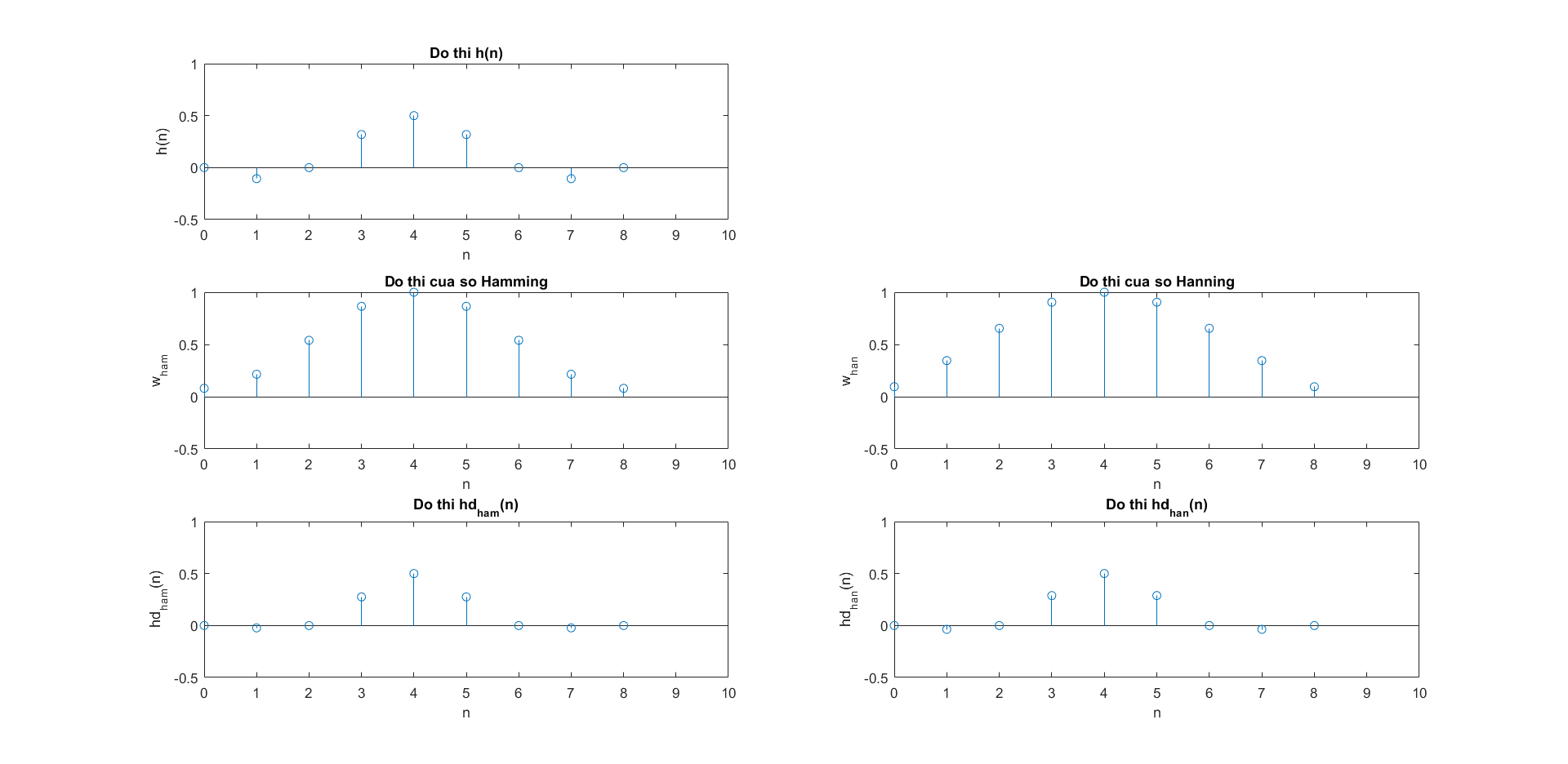
Nếu , ta có cửa sổ Hamming như sau:

*Ví dụ 7.2.3.1:* Hãy tổng hợp bộ lọc thông thấp FIR pha tuyến tính dùng cửa sổ Hanning và Hamming với , .

Giải

Bộ lọc thông thấp hữu hạn được tổng hợp theo biểu thức:

Thu được kết quả:



Hình 7.2.3.1 Đồ thị tín hiệu, , và ,

wc = 0.5\*pi;

N = 9;

phi=(N - 1)/2;

n = 0:N-1;

w\_ham = hamming(N);

w\_han = hanning(N);

h = (wc/pi)\*sin(wc\*(n-phi))./(wc\*(n-phi));

h(1,5) = 0.5;

hd\_ham = h.\*(w\_ham.');

hd\_han = h.\*(w\_han.');

subplot(3,2,1);

stem(n,h); title('Do thi h(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('h(n)');

subplot(3,2,3);

stem(n,w\_ham); title('Do thi cua so Hamming');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('w\_h\_a\_m');

subplot(3,2,4);

stem(n,w\_han); title('Do thi cua so Hanning');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('w\_h\_a\_n');

subplot(3,2,5);

stem(n,hd\_ham); title('Do thi hd\_h\_a\_m(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('hd\_h\_a\_m(n)');

subplot(3,2,6);

stem(n,hd\_han); title('Do thi hd\_h\_a\_n(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('hd\_h\_a\_n(n)');

**7.2.4 Cửa sổ Blackman**

Trong miền cửa sổ Blackman được định nghĩa như sau:

Điều kiện:

Cửa sổ có dạng trong biểu thức trên được gọi là cửa sổ Blackman tổng quát hóa. Ta thấy rằng các cửa sổ Hanning và Hamming chính là cửa sổ Blackman với hai hệ số và khác không:

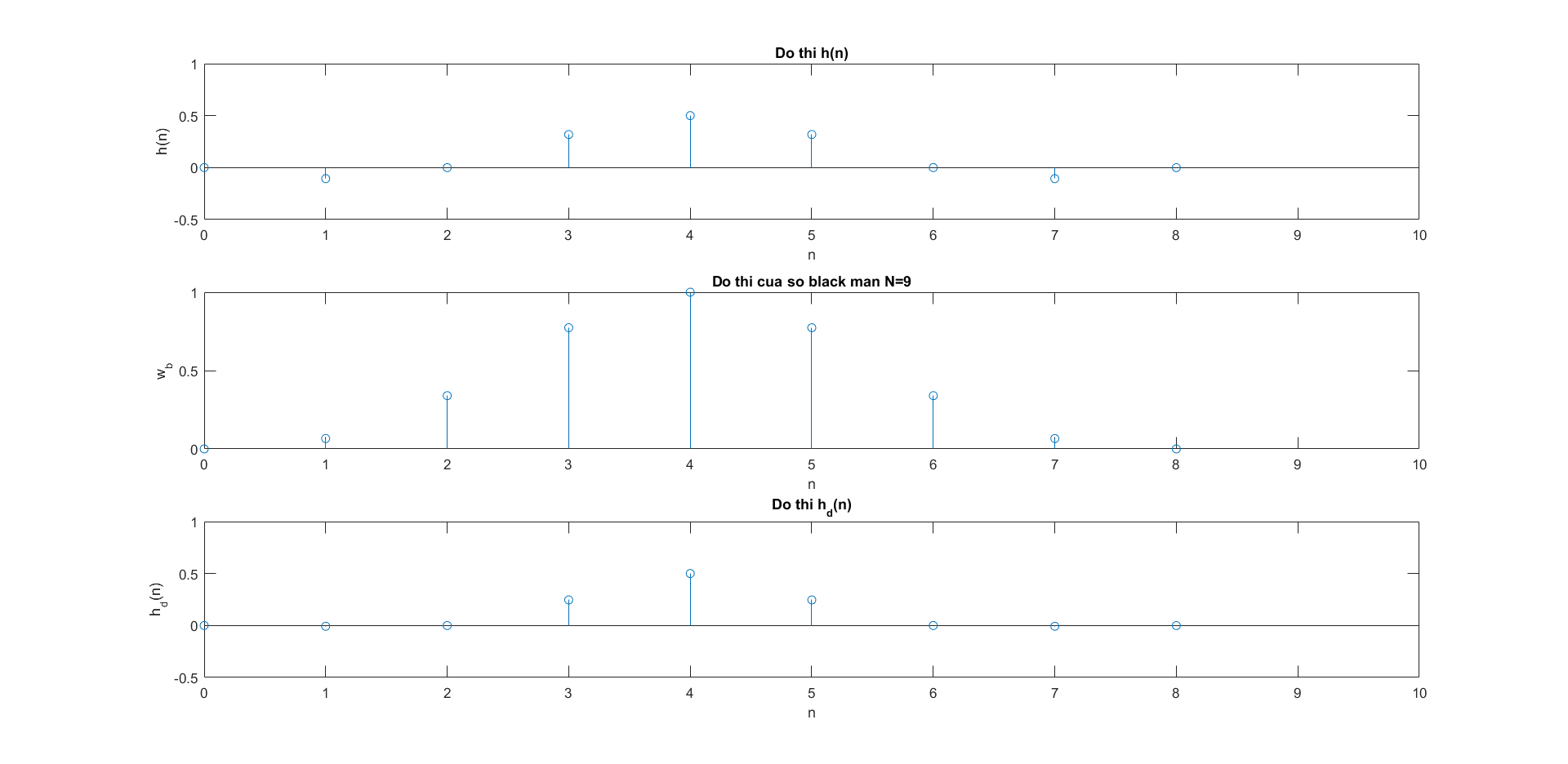
với

Trong thực tế Blackman thường dung ba hệ số khác không là , và . Việc xác định các hệ số này với mục đich làm giảm bề rộng của đỉnh trung tâm ta tìm được:

Xuất phát từ các giá trị này , Blackman đã định nghĩa một cửa sổ, sau này gọi là cửa sổ Blackman như sau:

*Ví dụ 7.2.4.1:* Hãy tổng hợp bộ lọc thông thấp FIR pha tuyến tính dùng cửa sổ Blackman với , .

Giải



Hình 7.2.4.1 Đồ thị tín hiệu , và

vwc = 0.5\*pi;

N = 9;

phi=(N - 1)/2;

n = 0:N-1;

w\_blackman = blackman(N);

h = (wc/pi)\*sin(wc\*(n-phi))./(wc\*(n-phi));

h(1,5) = 0.5;

hd = h.\*(w\_blackman.');

subplot(3,1,1);

stem(n,h); title('Do thi h(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('h(n)');

subplot(3,1,2);

stem(n,w\_blackman); title('Do thi cua so');

axis([0 10 0 1]); xlabel('n'); ylabel('w\_b');

subplot(3,1,3);

stem(n,hd); title('Do thi h\_d(n)');

axis([0 10 -0.5 1]); xlabel('n'); ylabel('h\_d(n)');

**7.2.5 Cửa sổ Kaiser**

Kaiser đã đưa ra một học các hàm cửa sổ. Trong miền tần số nó cho phép chúng ta định rõ mâu thuẫn giữa bề rộng của đỉnh trung tâm và biên độ của các đỉnh thứ cấp, mà chủ yếu là của đỉnh thứ cấp thứ nhất để xác định bằng cách thay đổi giá trị của tham số .

Một đặc trưng rất quan trọng của họ cửa sổ này là tạo ra khả năng giảm mạnh biên độ của các đỉnh thứu cấp mà vẫn đảm bảo bề rộng tối thiểu của đỉnh trung tâm.

Trong miền cửa sổ Kaiser được định nghĩa dạng tổng quát như sau:

Nếu càng lớn thì bề rộng của đỉnh trung tâm sẽ càng lớn.

Ở đây là hàm Bessel biến dạng loại một bậc không.

Bởi vì hàm Bessel rất phức tạp , vì vậy việc tính biểu thức giải tích của biến đổi Fourier là rất phức tạp và người ta không thể tìm một cách chính xác dưới dạng giải tích, nhưng người ta có thể tìm một cách gần đúng.

**7.3. Phương pháp lấy mẫu tần số**

- Xét đáp ứng xung của bộ lọc số thực tế FIR có chiều dài hữu hạn N:

* Biến đổi Fourier rời rạc với N điểm:
* Biến đổi Fourier rời rạc ngược:
* Xác định theo hàm của :
* chính là được đánh giá dựa trên vòng tròn đơn vị tại những điểm rời rạc với:

* Đối với đáp ứng tần số ta có:
* (3.3.3.1)

Với:

Biểu thức (3.3.3.1) chỉ đúng với dãy có chiều dài hữu hạn là N. Với bộ lọc số lý tưởng, đáp ứng xung có chiều dài vô hạn. Khi đó hàm truyền và đáp ứng tần số của bộ lọc số lý tưởng được xác định:

Xét gần đúng bằng hàm của bộ lọc thực tế. nhận được qua việc nội suy giữa các mẫu lấy trên tại các tần số

Sai số của phép gần đúng trên bằng không tại tần số và là hữu hạn đối với các tần số khác.

Tức là:

⬄

⬄

Qua phép gần đúng ta thu được bộ lọc số lí tưởng:

Bộ lọc số sẽ không lý tưởng nếu:

Với hàm nội suy:

Như vậy, ta có thể lấy từ N mẫu đáp ứng tần số của bộ lọc số lý tưởng để thu được đáp ứng tần số của bộ lọc số thực tế .

Thu được hai loại lấy mẫu tần số:

Loại 1:

Loại 2:

**7.3.1. Loại 1**

Xác định theo hàm của , trong tường hợp này , có thể viết:

Với bộ lọc số FIR pha tuyến tính thì:

Xét các bộ lọc cụ thể:

**7.3.1.1. Bộ lọc FIR N lẻ, đối xứng**

Ví dụ với N=9 ta thu được :

**7.3.1.2. Bộ lọc FIR N chẵn, đối xứng**

Ví dụ với N=8 ta thu được:

**7.3.2. Loại 2**

**7.3.2.1. Bộ lọc FIR N lẻ, đối xứng**

Ví dụ với N=9 ta thu được :

**7.3.2.2. Bộ lọc FIR N chẵn, đối xứng**

Ví dụ với N=8 ta thu được:

*Ví dụ 7.3.1.* Hãy tổng hợp bộ lọc số thông thấp FIR pha tuyến tính với , , dung phương pháp lấy mẫu tần số loại 1.

Giải

Ta có bộ lọc số thông thấp lý tưởng pha tuyến tính như sau:

Thay và ta có:

Chúng ta xét trong khoảng tần số với sau đó lấy mẫu để được trong khoảng với ..

Để thử xem có thỏa mãn chỉ tiêu kỹ thuật của bộ lọc số hay không , chúng ta thay và công thức trong trường hợp bộ lọc FIR N lẻ, đối xứng, ta có: ,

Vậy:

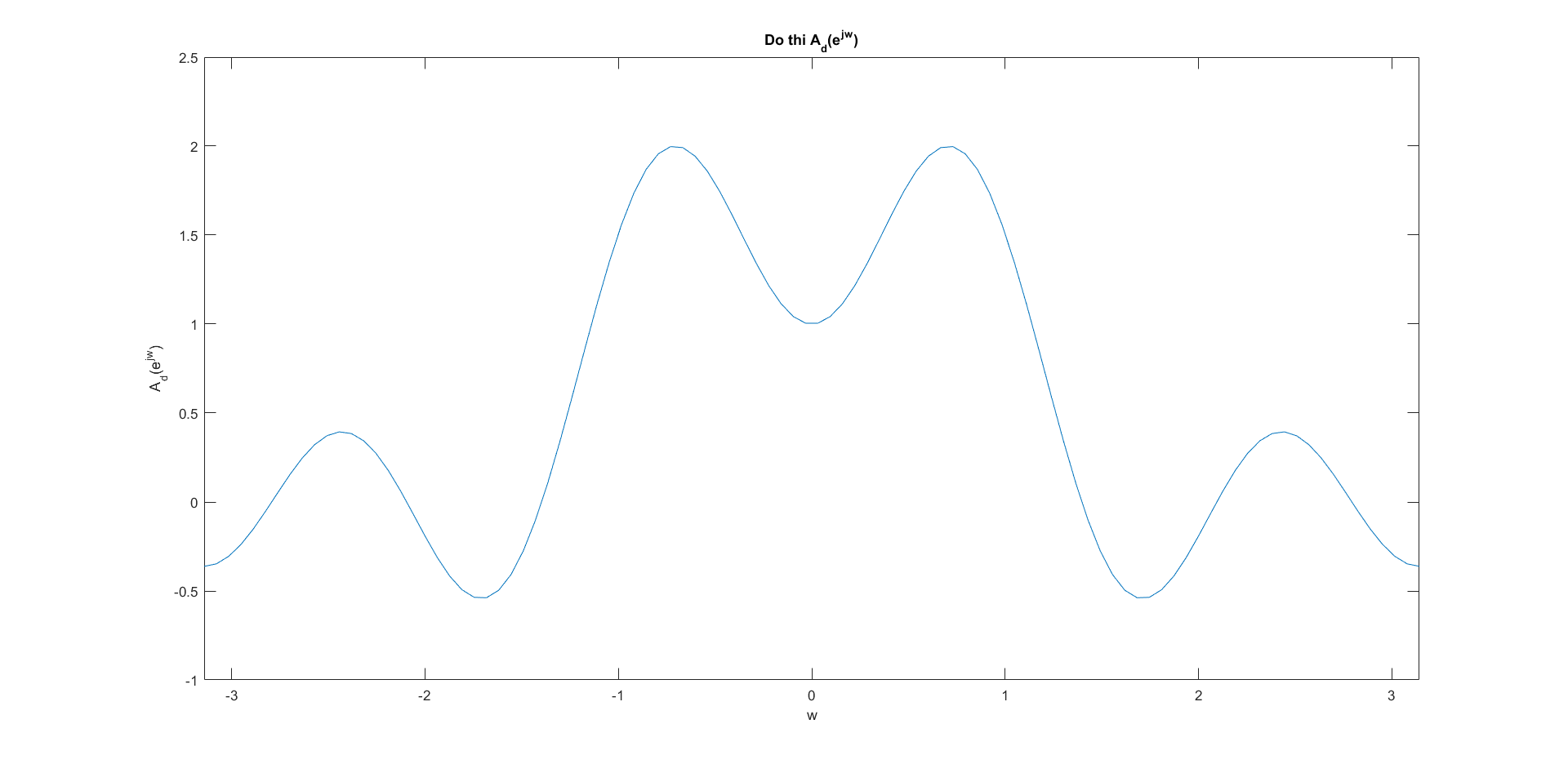
Hình 7.3.1 sẽ minh họa bằng đồ thị dạng của

Sau khi thử trong miền tần số, nếu đã thỏa mãn các chỉ tiêu kỹ thuật của bộ lọc só rồi thì ta tiến hành tìm các giá trị của đáp ứng xung với .

Theo biểu thức xác định ta có:

Thay các giá trị của vào ta có:

Đồ thị của cho trên hình 7.3.2



Hình 7.3.1 Đồ thị

N=9;

A1=1; A2=1; A0=1; A3=0; A4=0;

w=linspace(-pi,pi);

for i=1:length(w)

Ad0=A0/N\*sin(w\*N/2)./sin(w/2);

Ad1=(A1/N)\*(sin(N\*(w/2-pi\*i/N))./sin(w/2-pi\*i/N)+sin(N\*(w/2+pi\*i/N))./sin(w/2+pi\*i/N));

Ad2=(A2/N)\*(sin(N\*(w/2-pi\*i/N))./sin(w/2-pi\*i/N)+sin(N\*(w/2+pi\*i/N))./sin(w/2+pi\*i/N));

Ad3=(A3/N)\*(sin(N\*(w/2-pi\*i/N))./sin(w/2-pi\*i/N)+sin(N\*(w/2+pi\*i/N))./sin(w/2+pi\*i/N));

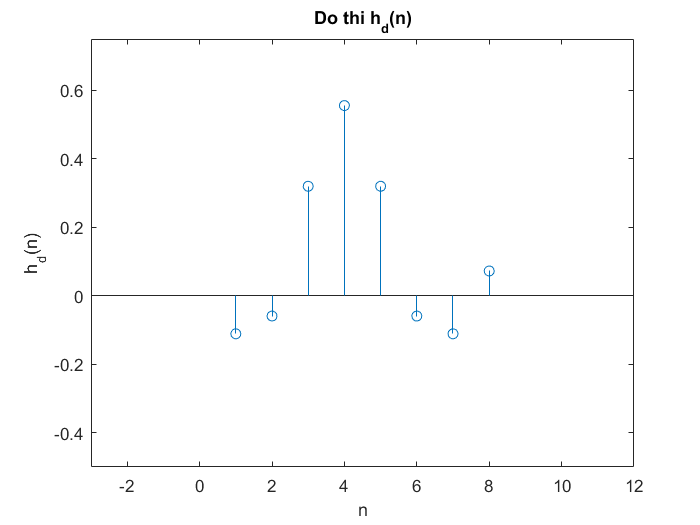
Ad4=(A4/N)\*(sin(N\*(w/2-pi\*i/N))./sin(w/2-pi\*i/N)+sin(N\*(w/2+pi\*i/N))./sin(w/2+pi\*i/N));

end

Ad=Ad0+Ad1+Ad2+Ad3+Ad4;

plot(w,Ad); title('Do thi A\_d(e^j^w)');

axis([-pi pi -1 2.5]); xlabel('w'); ylabel('A\_d(e^j^w)');



Hình 7.3.2 Đồ thị

N=9;

A1=1; A2=1; A0=1; A3=0; A4=0;

n=[1:N-1];

for i=1:length(n)

hd0=A0/N;

hd1=(A1/N)\*2\*(-1)^1\*cos(pi/N\*(2\*i+1));

hd2=(A2/N)\*2\*(-1)^2\*cos(2\*pi/N\*(2\*i+1));

hd3=(A3/N)\*2\*(-1)^3\*cos(3\*pi/N\*(2\*i+1));

hd4=(A4/N)\*2\*(-1)^4\*cos(4\*pi/N\*(2\*i+1));

hd(i)=hd0+hd1+hd2+hd3+hd4;

end

stem(n,hd); title('Do thi h\_d(n)');

axis([-3 12 -0.5 0.75]); xlabel('n'); ylabel('h\_d(n)');