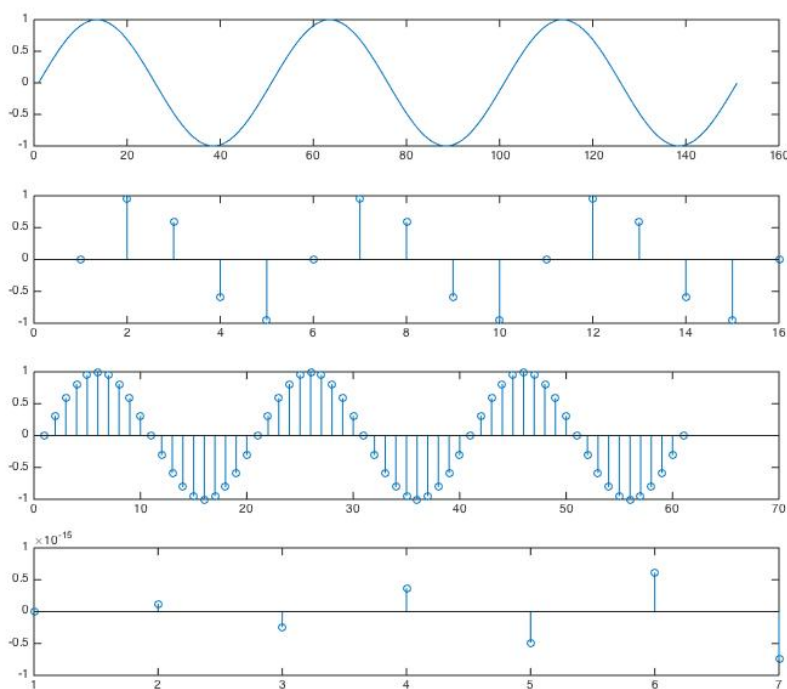


Chương 3

Lấy mẫu tín hiệu

Mục đích của bài thực hành này là giới thiệu cho sinh viên về quá trình lấy mẫu tín hiệu, phổ của tín hiệu và các hiện tượng xảy ra khi lấy mẫu với các tần số khác nhau (ví dụ như hiện tượng chồng phổ). Phần thực hành sẽ thực hiện trên cả tín hiệu mô phỏng và cả tín hiệu âm thanh.



Hình 3.1: Tín hiệu sin được lấy mẫu với các tần số khác nhau.

3.1 Lý thuyết lấy mẫu tín hiệu

Giả sử chúng ta có một tín hiệu sin liên tục theo thời gian $x = \sin(2\pi ft)$ trong đó f (Hz) là tần số vốn có của tín hiệu, t là biến thời gian, nhận mọi giá trị thực. Đồ thị của nó được biểu diễn trên hình 4.1a. Quá trình lấy mẫu và giữ mẫu tín hiệu được thực hiện bằng cách cứ sau một khoảng thời gian cố định T_s ta lấy một giá trị,

T_s lớn: số mẫu thu được nhỏ, các mẫu cách xa nhau.

T_s nhỏ: số mẫu thu được lớn, các mẫu gần nhau.

Rõ ràng, tùy theo T_s , tín hiệu lấy mẫu rời rạc thu được có mức độ giống với $x(t)$ khác nhau (hình 4.2(b, c, d)).

Bài tập 1: Mô phỏng tín hiệu dạng sin có tần số $f = 1000Hz$:

```
f=1000;
t=0:1/(50*f):3/f;
x = sin(2*pi*f*t);
figure;
subplot(4,1,1);
plot(t,x,'r'); % show continued signal
%hold on;
%stem(t,x); % show sampled signal
%grid on;
```

Bây giờ để mô tả quá trình lấy mẫu với các tần số khác nhau, ta thay đổi độ chia trong biến thời gian. Độ chia càng nhỏ, thì tần số lấy mẫu càng lớn, và ngược lại. Lưu ý, để hiện thị nhiều đồ thị trên cùng hình vẽ, ta sử dụng lệnh *subplot*; hiện thị tín hiệu dạng rời rạc ta dùng lệnh *stem* để vẽ hình.

```
t1=0:1/(10*f):3/f;
x1 = sin(2*pi*f*t1);
subplot(4,1,2);
stem(t1,x1);
grid on; % show sampled signal 1
```

```
t2=0:1/(1.5*f):3/f;
x2 = sin(2*pi*f*t2);
subplot(4,1,3);
stem(t2,x2);
grid on; % show sampled signal 2
```

```
t3=0:1/(100*f):3/f;
```

```
x3 = sin(2*pi*f*t3);
subplot(4,1,4);
stem(t3,x3); % show sampled signal 3

grid on;
```

3.2 Kiểm tra phổ của tín hiệu

Chúng ta đã biết phổ của tín hiệu \sin / \cos có dạng vạch. Phổ của $\sin(2\pi ft)$ có dạng là hai vạch có biên độ $1/2$ tại tần số $\pm 2\pi f$. Giả sử ta có tín hiệu x là tổng của 2 tín hiệu \sin có tần số lần lượt là 50 Hz và 120 Hz. Gọi F_s là tần số lấy mẫu. Khi $F_s > 2f_{\max}$ sẽ không có chồng phổ xảy ra; tín hiệu thu được sẽ có tần số như tín hiệu ban đầu. Ngược lại khi $F_s < 2f_{\max}$; hiện tượng chồng phổ xảy ra, tần số "lạ" sẽ xuất hiện. Để hiển thị tín hiệu trên miền tần số, ta dùng lệnh *fft* để tính biểu diễn tần số $X(w)$ của tín hiệu đầu vào x và *abs(X)* và *angle(X)* để xác định phổ biên độ và phổ pha. Đoạn lệnh sau sẽ thực hiện vẽ tín hiệu trong miền thời gian và tần số:

```
Fs = 1000; % Sampling frequency
T = 1/Fs; % Sample time
L = 1000; % Length of signal ; số mẫu sử dụng để vẽ tín hiệu
t = (0:L-1)*T; % Time vector

x = 0.7*sin(2*pi*50*t) + sin(2*pi*120*t); %Tín hiệu đầu vào, là tổng của 2 sin
y = x + 2*randn(size(t)); % Tín hiệu có thêm nhiễu
figure;
%%% Plot signal in time domain:
subplot(2,1,1)
plot(Fs*t(1:50),x(1:50))
title(' Original signal in time domain')
subplot(2,1,2);
plot(Fs*t(1:50),y(1:50))
title(' Noised signal in time domain')
xlabel('time (milliseconds)')

%%% Plot signal in frequency domain when Fsampling = 1000Hz:
figure;

NFFT = 2^nextpow2(L); % Next power of 2 from length of y
subplot(2,1,1);
X = fft(x,NFFT)/L;
f = Fs/2*linspace(0,1,NFFT/2+1);
```

```
plot(f,2*abs(X(1:NFFT/2+1)))
title('Amplitude Spectrum of original x(t)')
xlabel('Frequency (Hz)')
ylabel('|X(f)|');

subplot(2,1,2);
Y = fft(y,NFFT)/L;

plot(f,2*abs(Y(1:NFFT/2+1)))
title('Amplitude Spectrum of noised signal y(t)')
xlabel('Frequency (Hz)')
ylabel('|Y(f)|');
```

Bài tập 2:

Bây giờ lấy mẫu lần lượt với $F_s = 80$ Hz, $F_s = 100$, $F_s = 120$, $F_s = 150$, $F_s = 200$, $F_s = 300$. Hiện tượng chồng phổ sẽ như thế nào. Chạy lại đoạn lệnh trên với các giá trị F_s khác nhau.

Bài tập 3:

Kiểm tra bài tập 2 với tín hiệu $x = \sin(2\pi 1000t) + 2\cos(2\pi 750t) - \sin(2\pi 400t)$ với các F_s khác nhau để nhìn chồng phổ và không có chồng phổ.

3.3 Lấy mẫu tín hiệu âm thanh

Phần này các bạn sẽ kiểm tra ảnh hưởng của lấy mẫu đối với một tín hiệu âm thanh. Tín hiệu âm thanh ở đây là tín hiệu rời rạc đã được lấy mẫu từ âm thanh tương tự với tần số f_{0s} và *save* dưới dạng file *audioclip1.wav*. Sử dụng lệnh `[y, fs] = wavread('audioclip1.wav');` các bạn sẽ có được dữ liệu âm thanh lấy mẫu và thông tin về tần số lấy mẫu $f_{0s} = 16kHz$. Sử dụng lệnh `sound(y, fs)` để nghe thử âm thanh.

Bài tập 4:

1. Vẽ tín hiệu âm thanh và phổ biên độ của tín hiệu âm thanh y .
2. Tín hiệu âm thanh này đã được lấy mẫu từ một tín hiệu tương tự. Theo tiêu chuẩn Nyquist, tần số cực đại của tín hiệu tương tự phải nhỏ hơn $2f_{0s}$. Vậy tín hiệu âm thanh tương tự có tần số là bao nhiêu?
3. Giả sử tín hiệu âm thanh tương tự này được lấy mẫu với tần số $F_s = f_{0s}/4$ bằng cách thực hiện *downsampling*; có nghĩa là cứ 4 mẫu ta chỉ giữ lại 1, bỏ đi 3. Điều này tương đương với lấy mẫu tín hiệu với $F_s = 16/4 = 4kHz$. Tín hiệu ra có bị chồng phổ không? Vẽ phổ của tín hiệu ra trong trường hợp này? Để không có chồng phổ thì phải làm thế nào? Nghe thử âm thanh mới: `sound(ysampling, fs/4)`. Nhận xét
4. Giả sử vẫn lấy mẫu với tần số $4kHz$, thì tín hiệu đầu vào phải có tần số cực đại bằng bao nhiêu?
5. Giả sử cho tín hiệu âm thanh qua bộ lọc thông thấp tần số cắt $2k$, trước khi lấy mẫu. Vẽ phổ

của tín hiệu ra? Chạy và kiểm tra âm thanh nghe được. *sound (yrecover , fs/4)*