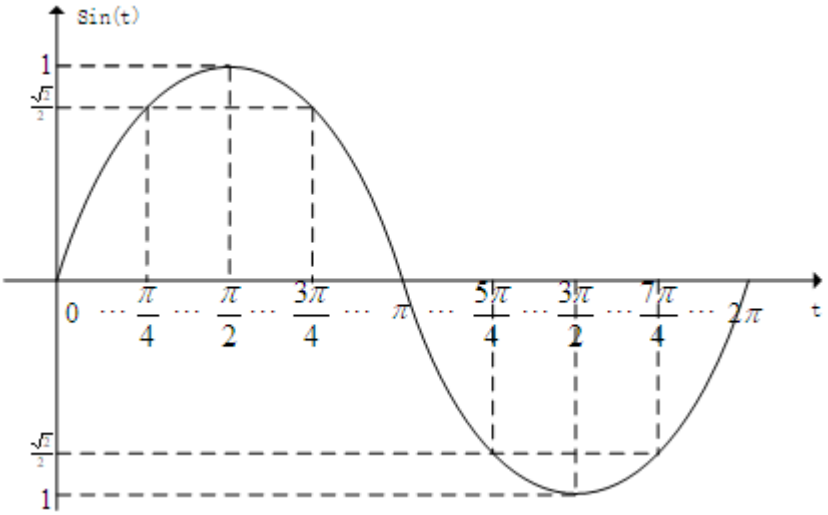
Singnal Creating and Plotting（信号的创建及绘制）

Numeric Method（数值方法）

Create a numeric signal



t	0	$\frac{\pi}{4}$	$\frac{\pi}{2}$	$\frac{3\pi}{4}$	π	$\frac{5\pi}{4}$	$\frac{3\pi}{2}$	$\frac{7\pi}{4}$	2π
sin (t)	$\sin(0) = 0$	$\sin\left(\frac{\pi}{4}\right) = \frac{\sqrt{2}}{2}$	$\sin\left(\frac{\pi}{2}\right) = 1$	$\sin\left(\frac{3\pi}{4}\right) = \frac{\sqrt{2}}{2}$	$\sin(\pi) = 0$	$\sin\left(\frac{5\pi}{4}\right) = -\frac{\sqrt{2}}{2}$	$\sin\left(\frac{3\pi}{2}\right) = -1$	$\sin\left(\frac{7\pi}{4}\right) = -\frac{\sqrt{2}}{2}$	$\sin(2\pi) = 0$

```
clear; clf;
t = 0:pi/4:2*pi          % 先定义时间轴（横轴）
sigt = sin(t)            % 根据时间轴算出对应函数值。时间轴、函数值一一对应
```

Plot a numeric signal

- **plot** 用于绘制数值法创建的连续信号

```
clear; clf;
t = 0:0.1:2*pi-0.01;    % 先定义时间轴（横轴）
sigx = sin(t);          % 根据时间轴算出函数值
sigy = cos(t);
```

```
plot(t,sigx);           % 绘制连续信号图形，横轴坐标为时间
plot(t,sigx, '-r');          % 改变绘图的颜色（'r'）和线型（'-'）
plot(t,sigx,'--r',t,sigy);  % 将 2 个信号绘制在同一幅图中
```

Symbolic Method（符号方法）

Create a symbolic signal

```
clf;
syms x              % 定义符号变量
y = sin(2*pi*x)     % 写出符号表达式
```

Plot a symbolic signal

- **fplot** 用于绘制符号法创建的信号

```
clf;
syms x              % 定义符号变量
y = sin(2*pi*x)
```

```
fplot(x,y);
fplot(y,[0 5]);      % 第二个参数[0 5]限制了 y 的横轴范围，如省略，默认范围为[-5 5]
fplot(x,y,[0 5],"Color",'m',"LineStyle","--")    % 修改图片颜色、线型等
```

About Plotting（关于绘图）

Create subplot

- **subplot** is used to create a subgraph, its format is like:subplot(number of subgraph rows, number of subgraph columns, current active subgraph)

```
clf;
subplot(2,2,1); plot(t,sigx);    % 创建 2 行 2 列的绘图区域，并在区域 1 上绘制(t,sigx)
subplot(2,2,2); plot(sigx);
subplot(2,2,3); plot(t,sigx,'-r');
subplot(2,2,4); plot(t,sigx,'-r',t,sigy);
```

Labeling

- When using **legend**, name the dignals in the drawing order.

```
clf;
plot(t,sigx,t,sigy);
title('Sine');       % 图片标题
xlabel('time');      % 横轴性质，单位
ylabel('amplitude');     % 纵轴性质，单位
legend('sin(x)','sin(y)');       % 图片中有多个信号时，每个信号的说明
```

Display range

- **axis** is used to limit the range of both X axis and Y axis.
- **xlim** is used to limit the range of X axis
- **ylim** is used to limit the range of Y axis

```
axis([0 6.5 -1.1 1.1]);     % 使用数组的形式组织范围参数:[xmin xmax ymin ymax]
```

```
xlim([0 5]);        % [xmin xmax]       % 只限制 x 轴范围
ylim([-1 1]);       % [ymin ymax]       % 只限制 y 轴范围
```

Add grid

```
grid on;        % 大网格
grid minor;     % 更细致的网格
```

Common Functions in Control Systems（控制原理中常用函数）

阶跃函数：heaviside(t)

```
clear;clf;
% 数值法
dt = 0.1;
t = -5:dt:5;
ft = heaviside(t);

% 数值法
syms x
y = heaviside(x);

subplot(2,2,1); plot(t,ft);
title('numeric');xlabel('t(s)');ylabel('f(t)');
subplot(2,2,2); fplot(x,y);
title('symbolic');xlabel('t(s)');ylabel('f(t)');
subplot(2,2,3); plot(t,ft);
axis([-0.5 0.5 -inf inf]); title('numeric');xlabel('t(s)');ylabel('f(t)'); grid minor;
subplot(2,2,4); fplot(x,y,[-0.5 0.5]);
title('symbolic');xlabel('t(s)');ylabel('f(t)');grid minor;
```

传递函数：tf(num,den)

$$a_3y''' + a_2y'' + a_1y' + a_0y = b_3f''' + b_2f'' + b_1f' + b_0f$$

$$G(s) = \frac{Y(s)}{F(s)} = \frac{b_3s^3 + b_2s^2 + b_1s + b_0}{a_3s^3 + a_2s^2 + a_1s + a_0}$$

可使用$\mathrm{tf}(\mathrm{nun}, \mathrm{den})$函数来生成传递函数，其中

$$\mathrm{num} = [b_3, b_2, b_1, b_0]$$

$$\mathrm{den} = [a_3, a_2, a_1, a_0]$$

如：$G(s) = \dfrac{240}{s^3 + 12s^2 + 20s + 240}$

```
clf;clear;
```

```
num = [240];              % 分子
den = [1,12,20,240];      % 分母
G = tf(num,den)           % 生成传递函数
```

单位阶跃响应：step(sys, t)

```
clf;clear;
num = [200];              % 分子
den = [1,12,20,200];      % 分母
sys = tf(num,den)           % 生成传递函数
t = 0:0.01:10;
s = step(sys,t);
plot(t,s); xlabel('t(s)');ylabel('y(t)');title('step response')
```

单位冲击响应：impulse(sys, t)

拉普拉斯变换：laplace(ft)

拉普拉斯反变换：ilaplace(Fs)

$$f(t) = t \cdot e^{-2t} \cdot u(t)$$

```
clear; clf;
syms t
ft = t*exp(-2*t)*heaviside(t)
Fs = laplace(ft)
ft1 = ilaplace(Fs)
```

卷积函数：conv(a,b)

$$G(s) = \frac{5s}{(0.1s+1)\cdot(0.5s+1)}$$

```
clf; clear;
syms s
G1 = 5*s/((0.1*s+1)*(0.5*s+1))
num = [5,0];     % conv([5,0],[1]);
den = conv([0.1 1],[0.5 1])
G2 = tf(num,den)
```

多项式的根：roots(p)

$$G(s) = \frac{5s}{0.05s^2+0.6s+1}$$

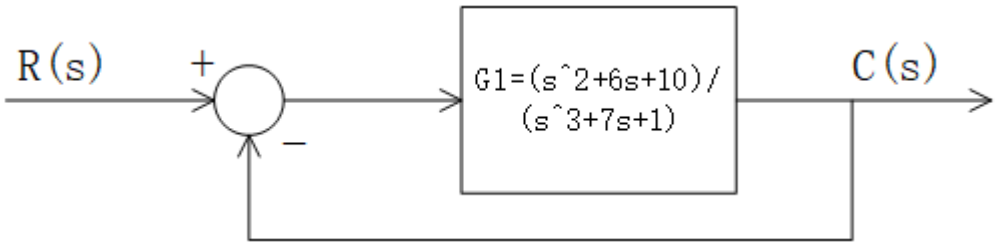
```
clf; clear;
num = [5,0];     % conv([5,0],[1]);
den = [0.05 0.6 1];
roots(num)
```

```
ans = 0
```

```
roots(den)
```

```
ans = 2×1
  -10.0000
   -2.0000
```

负反馈系统：feedback(sys1,sys2,sign)




```
clear; clf;
syms t
h = heaviside(t);
num =  [1 6 10];
den = [1 0 7 1];
G1 = tf(num,den);
G2 = 1;
G = feedback(G1,G2,-1); % -1 for negative feedback, 1 for positive feedback (default -1)
subplot(2,1,1);step(G1);xlim([0 50]);grid on;
subplot(2,1,2);step(G);xlim([0 50]);grid on;
```

零极点图：pzplot(sys)

```
clf; clear;
num = [5,0];    % conv([5,0],[1]);
den = conv([0.1 1],[0.5 1]);
G = tf(num,den);
pzplot(G)
```

伯德图：bode(sys)

```
clf; clear;
num = [5,0];    % conv([5,0],[1]);
den = conv([0.1 1],[0.5 1]);
G = tf(num,den);
bode(G)
```

奈奎斯特图：nyquist(sys)

```
clf; clear;
num = [5,0];    % conv([5,0],[1]);
den = conv([0.1 1],[0.5 1]);
G = tf(num,den);
nyquist(G)
```