Indefinite integral

Last Tut_1, at the last part, there is an introduction to **Integral** using symbolic math.I think the power of the symbolic math is integral operations. See and learn how to code it

```
clear all;
LW = 'LineWidth';
syms x
f = sin(x);
g = sin(x)^2;
h = sin(x^2);
int(f)
```

ans $= -\cos(x)$

```
fplot(int(f),LW,2); hold on; grid on
int(g)
```

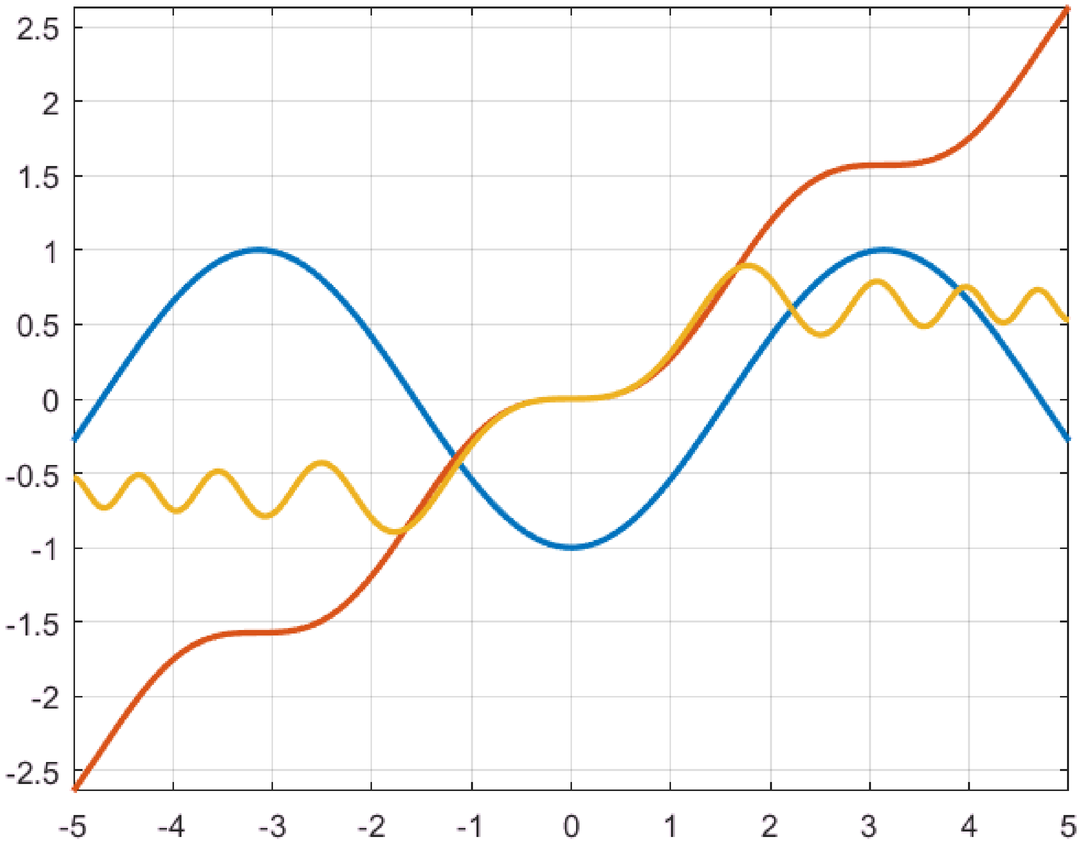
ans $= \frac{x}{2} - \frac{\sin(2\,x)}{4}$

```
fplot(int(g),LW,2);
int(h)
```

ans =

$$\frac{\sqrt{2}\,\sqrt{\pi}\,S\left(\frac{\sqrt{2}\,x}{\sqrt{\pi}}\right)}{2}$$

```
fplot(int(h),LW,2);
hold off
```



There are three results, two of them you may be familiar, the last one is not as $S\left(\sqrt{\frac{2}{\pi}}x\right)$. What is this?

Special Functions

Here a special function which is not solved analytically such as

$$\int sin(x^2)\,dx$$

which is called as a **Fresnel integral**.

There are another special function, **error function,** as

$$erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt,\ erf(\infty) = 1$$

which is related to the gaussian probability density whic may be familiar to you.

definite integral

Let us study after **definite integrals.** Consider the following definite integral

$$\int_0^{\pi} sin(x)dx = -cos(\pi) + cos(0) = 2$$

which is coded as

```
clear all;clc
syms x
f = sin(x);
int(f,0 ,pi)
```

ans $= 2$

Now how the fresnel integral? For example

$$S(1) = \int_0^1 sin(t^2)dt$$

which can be evaluated as

```
clear all; clc;
syms x
h(x) = sin(x^2);
y=int(h,0,1)
```

y =

$$\frac{\sqrt{2}\,\sqrt{\pi}\,S\left(\frac{\sqrt{2}}{\sqrt{\pi}}\right)}{2}$$

Hence, in conclusion,

```
sqrt(2*pi)*fresnels(sqrt(2/pi))/2
```

ans =   0.310268301723381

which is also seen in the previous plot.

definite integral in data type

To plot a function, in data type(what I mean there should be step size) we may use the the command "plot ". To evaluate the definite integral, there is a command as 'integral", as

```
clear all; clc
```

```
h = @(x) sin(x.^2);
integral(h,0,1)
```

```
ans =
   0.310268301723381
```

Which is similar to the value using symbolic math. If you need more precisely

```
format long
```

which is same to the value of the symbolic math case. Why we may use the symbolic math instead of data type (what i mean as an anonymous function)?

First of all, In the case of indefinite integral case, we may not use the data type, since it needs a specific interval. Second, let us consider even if definite integral,

$$\int_1^c f(x)\ dx$$

where "c" is a variable. In this case it is better to use "symbolic math".

Solve algebraic equation(page 1-26)

Consider a one variable equation as

$$x^3 - 6x^2 + 11x - 6 = 0$$

which is solved as

```
syms x
solve(x^3 - 6*x^2 + 11*x -6)
```

ans =

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

```
solve(x^3 - 6*x^2 == 6-11*x)
```

ans =

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

So that the solutions are 1,2 or 3.

Solve a quadratic equation (p3-3)

Another example of a quadratic equation as

$$ax^2 + bx + c = 0$$

You may remember the solution formular. In matlab

```
clear all
syms a b c x
eqn = a*x^2 + b*x +c ==0;
solx  = solve(eqn,x)
```

solx =

$$\begin{pmatrix} -\frac{b+\sqrt{b^2-4\,a\,c}}{2\,a} \\ -\frac{b-\sqrt{b^2-4\,a\,c}}{2\,a} \end{pmatrix}$$

Solve a system equation(p.3-8)

a system of equations are multi equations with multivariables. In this case it is better to define a system to solve the system equation. Consider a system equation with two unknown variables

$$x + 2y = -1$$
$$2x + y = 1$$

to get the solution to satisfy the system equations , in matlab

```
clear all;
syms x y
eqn1 = x+2*y ==-1;
eqn2 = 2*x+y ==1;
S = solve(eqn1,eqn2);
S.x
```

ans $=1$

```
S.y
```

ans $=-1$

Solve differential equations

In control society, differential equations can not be emphasized **enough**! If something is changed, then you are better to think of difference or differential equation in time domain or in space domain. Let us a one dimensional linear differential equation,

$$\frac{dx}{dt} = -2t$$

in matlab, the solution is

```
clear all;
syms x(t)
ode = diff(x,t)+2*x ==0;
y = dsolve(ode)
```

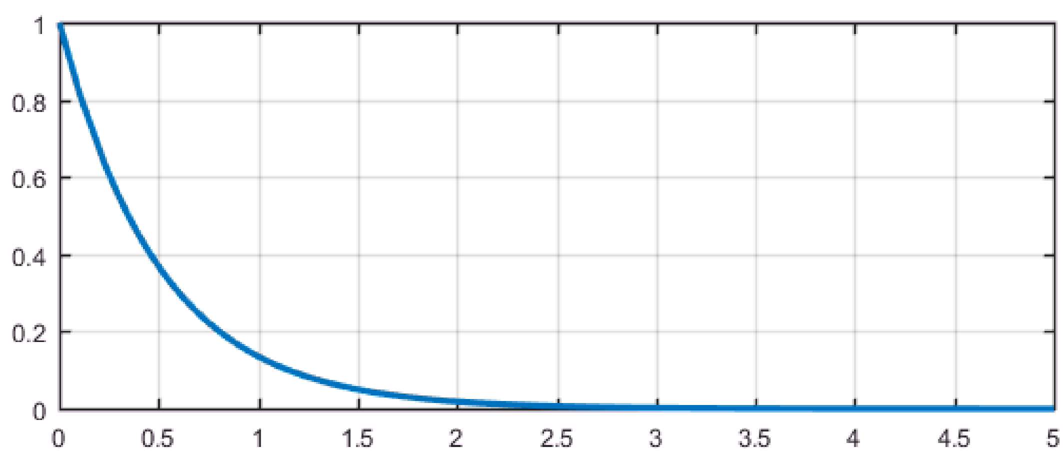
$y = C_1 e^{-2t}$

Here "ode" is a just function name as you assign, but the new commnad is "dsolve" , which stands for "differential solve".The solution is included with "one" unknown variable c_2, which should be idnetified by a known condition. Let us assume $x(0) = 1$, then

```
clear all;clf
syms x(t)
LW ='LineWidth';
ode = diff(x,t) +2*x ==0;
con = x(0)==1;
y = dsolve(ode,con)
```

$y = e^{-2t}$

```
fplot(y,[0 5],LW,2); hold on; grid on
```

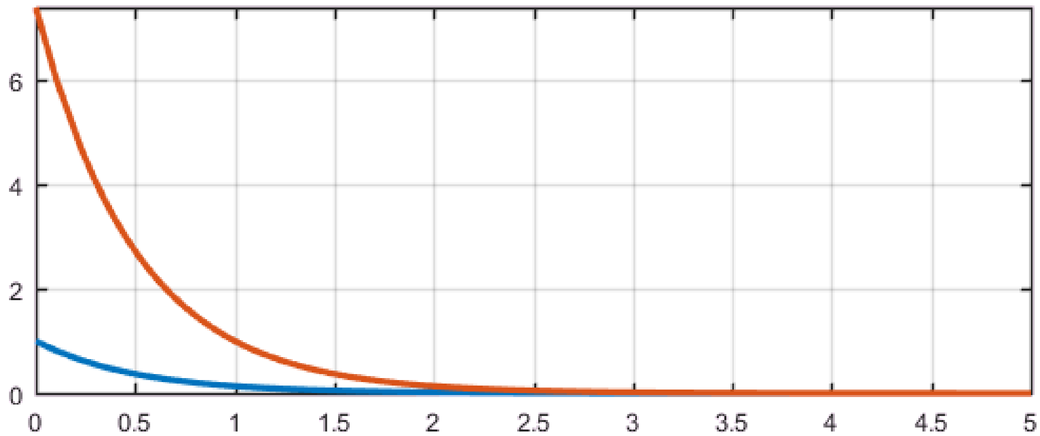


Here we may call the unknown condition $x(0) = 1$ as the initial condition $t = 0$. If a known condition is not initial, i.e., $x(1) = 1$, what is the solutuion? then in matlab

```
con1 = x(1) ==1;
y1 = dsolve(ode,con1)
```

$y1 = e^{-2t} e^{2}$

```
fplot(y1,[0 5],LW,2);
hold off
```



In the figure, you may see the red curve is the solution with the $y(1) = 1$. This is one of the merits to use "symbolic math" . What I mean, without using symbolic math, to get an approximate numerical solution.It is a little bit complicate. First of all, let us code to get a graph for the previous example. Let us see the previous equation once more. Since

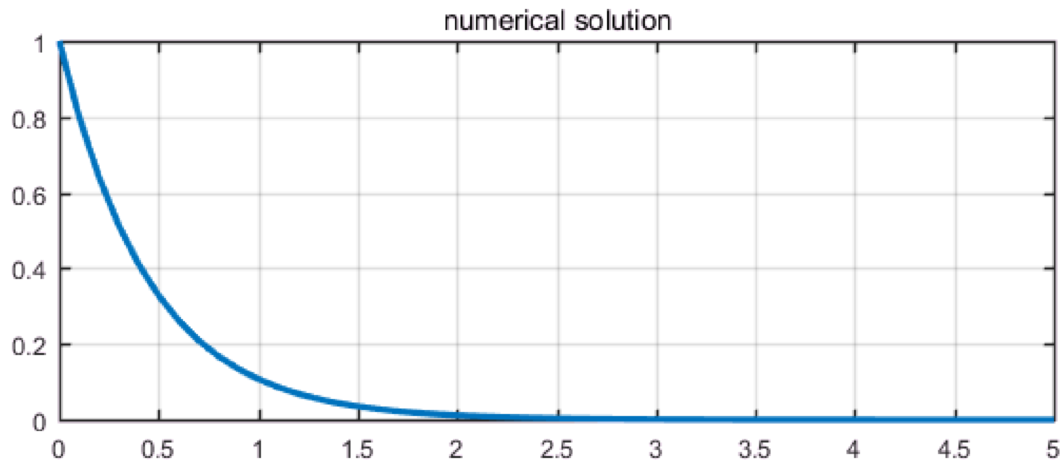
$$\frac{dx}{dt} \sim \frac{x(t+h)-x(t)}{h}$$

where h is a step size as small as possible. then the equation is approximated as

$$x(t+h) = hx(t) - 2x(t)$$

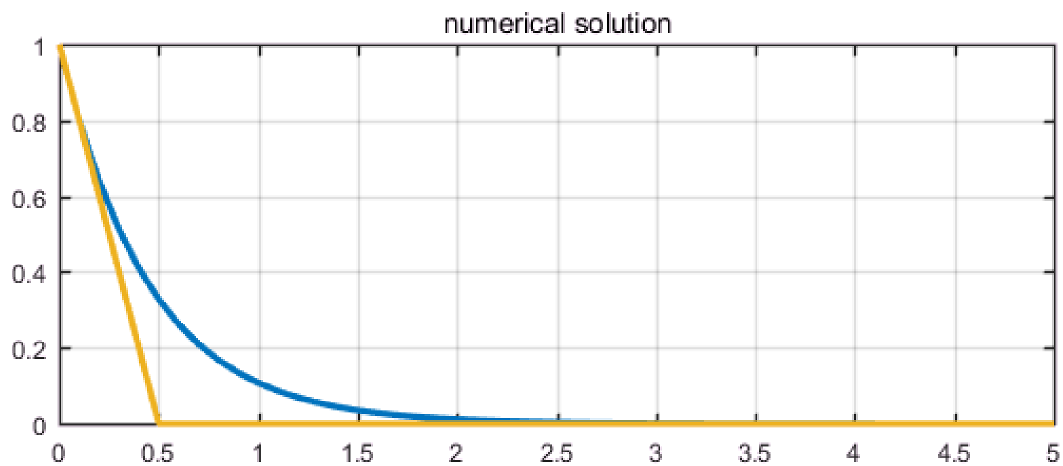
Hence the approximate solution is in code

```
clear all;clf
LW ='linewidth';
dt = 0.1;
t =0:dt:5;
x(1) = 1;
for i =1:size(t,2)-1
   x(i+1) = -2*x(i)*dt +x(i);
end
plot(t,x,LW,2); hold on;  grid on
title('numerical solution')
```



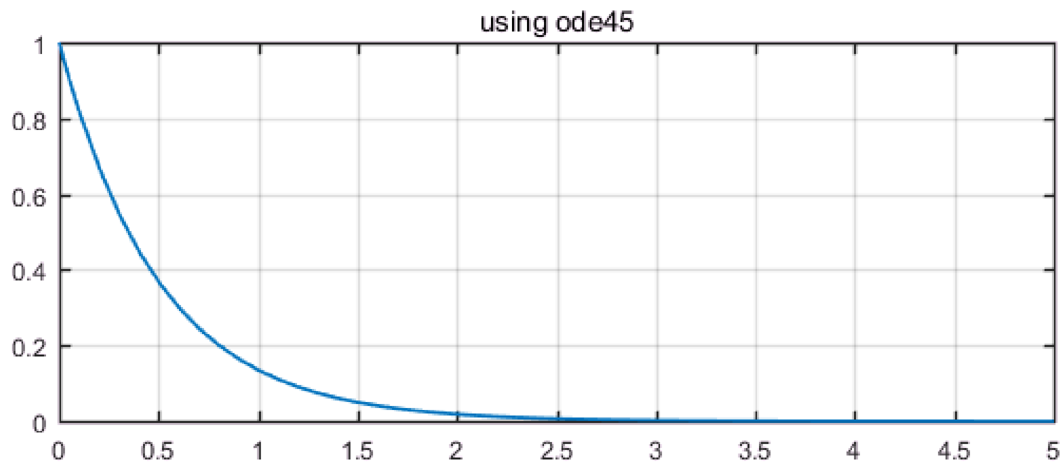
Here the step size h is assigned to 0.1. Then the graph is looks similar to that of symbolic math. However, if we define the sep size h = 0.5, (I exaggerate the effect due to the magnitude), we should define the step

```
clear all;
LW ='linewidth';
dt = 0.5;
t =0:dt:5;
x(1) = 1;
for i =1:size(t,2)-1
   x(i+1) = -2*x(i)*dt +x(i);
end
plot(t,x,LW,2); hold on;  grid on
title('numerical solution')
```



In the figure, the red curve is in the case of the step size $h = 0.5$. Here the blue one is a little bit good approximation, but the red one is not so good.The problem in here, to get an approximat solution to a given differential equation, the solution is dependent of the step size. In matlab there is another alternative way to get an approximate solution. Let us see the code

```
clear all; clc; clf;
tspan = [ 0 5];
y0 = 1;
[t,y] = ode45(@(t,x) -2*x, tspan, y0);
plot(t,y); grid on
title('using ode45')
```



Here the graph is good approximated. We use an anonymous function "ode45", provided by matlab.There are several anonymous functions to solve differential equation, you may chek it. In this case we did not define "syms" for the symbolic math neither the step size, which is automatically provided fby matlab. I may empahsize that to solve differential equation, in numerical methods, the initial conditions should be defined.Instead of the initial conditions, i,e,,y(t=1), you may change given differential equation accordingly. However, in symbolic math, it does not matter the initial or intermediate values.