# On MATLAB command: **dsolve**

The MATLAB command dsolve computes symbolic solutions to ordinary differential

equations.

**Syntax**

```
dsolve('eq1','eq2',...,'cond1','cond2',...,'v')
```

**Description**

dsolve('eq1','eq2',...,'cond1','cond2',...,'v') symbolically solves the ordinary differential equations eq1, eq2,... using v as the independent variable. Here cond1,cond2,... specify boundary or initial conditions or both. You also can use the following syntax:

dsolve('eq1, eq2',...,'cond1,cond2',...,'v').

The default independent variable is t.

The letter D denotes differentiation with respect to the independent variable. The primary default is d/dx. The letter D followed by a digit denotes repeated differentiation. For example, D2 is $d^2/dx^2$. Any character immediately following a differentiation operator is a dependent variable. For example, D3y denotes the third derivative of y(x) or y(t).

You can specify initial and boundary conditions by equations like y(a) = b or Dy(a) = b, where y is a dependent variable and a and b are constants. If the number of the specified initial conditions is less than the number of dependent variables, the resulting solutions contain the arbitrary constants C1, C2,....

You can input each equation or a condition as a separate symbolic equation.

The dsolve command accepts up to 12 input arguments.

dsolve can produce the following three types of outputs:

For one equation and one output, dsolve returns the resulting solution with multiple solutions to a nonlinear equation in a symbolic vector.

For several equations and an equal number of outputs, dsolve sorts the results alphabetically and assigns them to the outputs.

For several equations and a single output, dsolve returns a structure containing the solutions.

If dsolve cannot find a closed-form (explicit) solution, it attempts to find an implicit solution. When dsolve returns an implicit solution, it issues a warning. If dsolve cannot find either an explicit or an implicit solution, then it issues a warning and returns the empty sym. In such a case, you can find a numeric solution, using the MATLAB ode45 functions (we will learn how to use it later). In some cases involving nonlinear equations, the output is an equivalent lower order differential equation or an integral.

**Example 1**: Solve

$$\frac{dy}{dt} = 2 * y + t$$

Solution:

```
>> dsolve('Dy=2*y+t',’t’)

ans =

(C7*exp(2*t))/4 - t/2 - 1/4
```

Or

```
>> dsolve('Dy=2*y+t')

ans =

(C7*exp(2*t))/4 - t/2 - 1/4
```

Remark: the default independent variable is t, if you do not specify in the command.

**Example 2**: Solve

$$\frac{dy}{dt} = 2 * y + t, y(0) = 1$$

Solution:

```
>> dsolve('Dy=2*y+t','y(0)=1','t')


ans =


(5*exp(2*t))/4 - t/2 - 1/4
```

**Example 3**: Solve

$$\frac{dy}{dx} = 2 * y + x^2, y(0) = 1$$

```
>> dsolve('Dy=2*y+x^2','y(0)=1','x')


ans =


(5*exp(2*x))/4 - x/2 - x^2/2 - 1/4
```

**Example 4**:  Solve

$$\frac{d^2y}{dx^2} + 2\frac{dy}{dx} = e^x, y(0) = 1, y'(0) = 0$$

Solution:

```
>> dsolve('D2y+2*Dy=exp(x)','y(0)=1','Dy(0)=0','x')


ans =


1/(6*exp(2*x)) + exp(x)/3 + 1/2
```

**Example 5**:  Solve

$$\frac{d^2y}{dx^2} + 2\frac{dy}{dx} = \sin(x), y(0) = -1, y'(0) = 1$$

Solution:

```
>> dsolve('D2y+2*Dy=sin(x)','y(0)=-1','Dy(0)=1','x')

 ans =- 3/(5*exp(2*x)) - (2*cos(x))/5 - sin(x)/5
```

**How to plot the solution of differential equation** after you obtain the solution to the differential equation? Use **ezplot** See the following example.

**Example 5**: Solve

$$\frac{d^2y}{dx^2}+2\frac{dy}{dx}=\sin(x), y(0)=1, y'(0)=0$$

and plot the solution curve.

Solution:

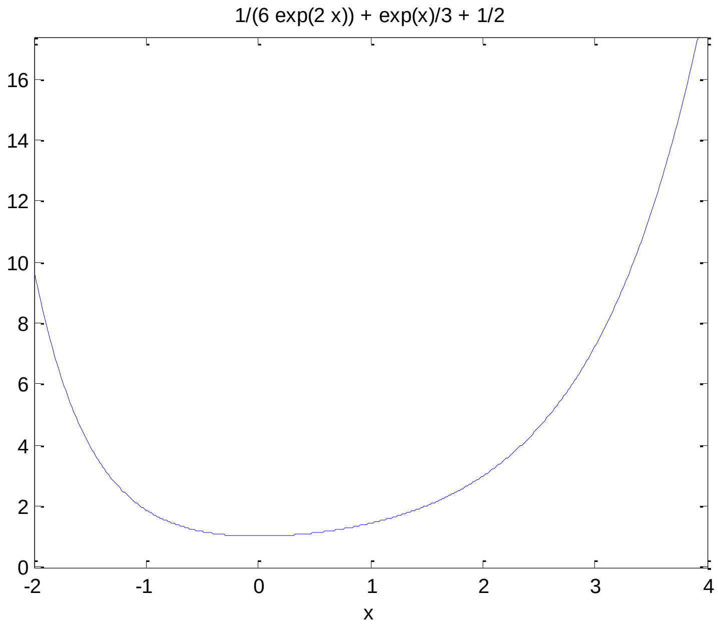
```
>> sol=dsolve('D2y+2*Dy=exp(x)','y(0)=1','Dy(0)=0','x')


sol =


1/(6*exp(2*x)) + exp(x)/3 + 1/2


>> ezplot(sol,[-2,4])

>>
```



**Example 6**: Solve

$$\frac{dy}{dt} = 2 * y + t$$

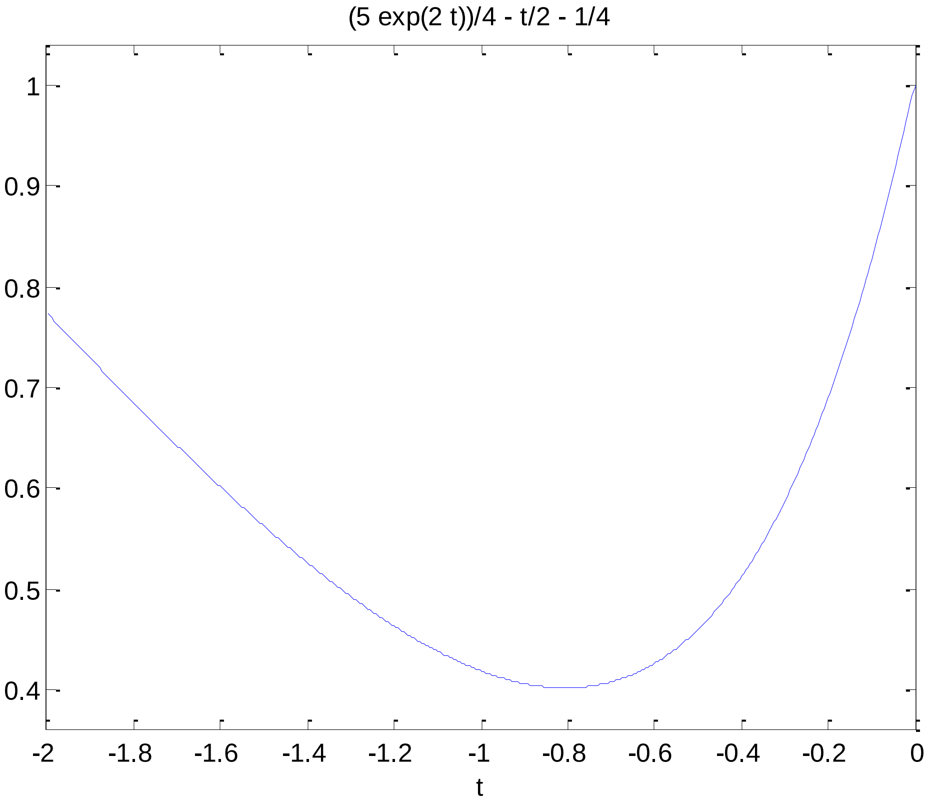
and plot the solution curve with independent variable in [-2,0].

```
>> s=dsolve('Dy=2*y+t','y(0)=1','t')


s =


(5*exp(2*t))/4 - t/2 - ¼

>> ezplot(s,[-2,0])
```


(5 exp(2 t))/4 - t/2 - 1/4
t