Chapter 12 - Solving Differential Equations

Table of Contents

Introduction

Matlab is quite powerful when it comes to solving differential equations. The standard command is dsolve (think Differential Solve) and the acceptable way to use it is to first symbolically declare the function and then use it inside `dsolve`. For example Suppose we wish to find the function `y`, a function of `t`, such that `y'=2*y`. In Matlab you should think about this differential equation as `diff(y)==2*y`

We can type this into Matlab as:

```
syms y(t)
dsolve(diff(y)==2*y)

ans =
C2*exp(2*t)
```

Note that as we've seen `diff(y)` represents the derivative of the variable y. Also we should note here that you may get a differently numbered constant `C1`, `C2`, etc. depending upon how many constants that have shown up in Matlab for you. Don't worry for now about how they're numbered.

Additional Independent Variables

If we have additional variables that's fine too:

```
syms y(t)
dsolve(diff(y)+3*t==2*y)

ans =
(3*t)/2 + (C3*exp(2*t))/4 + 3/4
```

You don't have to use `t` although that's pretty common:

```
syms y(p)
dsolve(diff(y)+3*p==2*y)

ans =
(3*p)/2 + (C4*exp(2*p))/4 + 3/4
```

We can use higher derivatives like `diff(y,2)` and `diff(y,3)` as well. For example we can solve `y''=2y` treating `y` as a function of `t` by typing:

```
syms y(t)
dsolve(diff(y,2)==2*y)

ans =
C5*exp(2^(1/2)*t) + C6*exp(-2^(1/2)*t)
```

Initial Value Problems

We can also state initial conditions by adding the initial condition as a second condition. For example if we wanted to solve `y'=2y+sin(t)` with `y(0)=1` we would do:

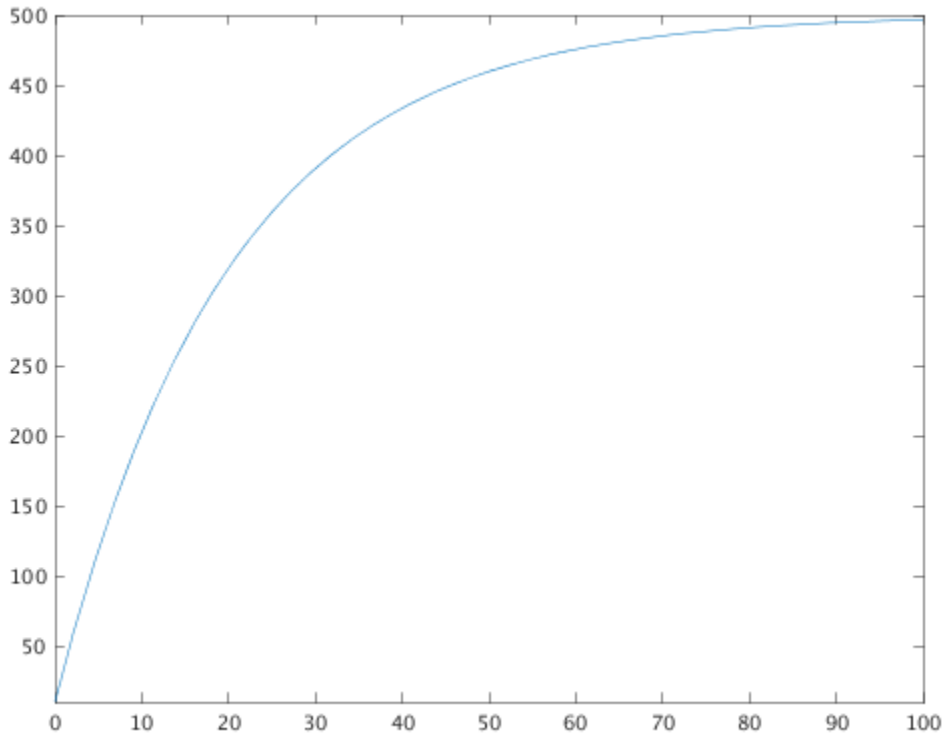
```
syms y(t)
dsolve(diff(y)==2*y+sin(t),y(0)==1)

ans =
(6*exp(2*t))/5 - (5^(1/2)*cos(t - atan(2)))/5
```

Plotting a solution.

Plotting a solution is as easy as wrapping `dsolve` in `fplot`:

```
syms f(t)
fplot(dsolve(diff(f)==0.05*(500-f),f(0)==10),[0,100])
```



Two More Compound Examples

The following solves a differential equation and plugs in a value, all together:

```
syms y(t);
subs(dsolve(diff(y)+3*y+10==0,y(1)==2),t,6)
```

```
ans =
(16*exp(-15))/3 - 10/3
```

The following solves a differential equation, sets it equal to 7 and solves, all together.

```
syms y(t);
solve(dsolve(diff(y)+3*y+10==0,y(1)==2)==7)
```

```
ans =
-log((31*exp(-3))/16)/3
```

This is a little tricky to parse at first, so closely observe that the command is built from the inside out:

```
diff(y)+3*y+10==0
dsolve(diff(y)+3*y+10==0,y(1)==2)
solve(dsolve(diff(y)+3*y+10==0,y(1)==2)==7)
```

An Example with Constants

You may also have other constants lying around in your differential equation, for example:

```
syms a y(t)
dsolve(diff(y)==a*y+a)
```

```
ans =
C16*exp(a*t) - 1
```

Published with MATLAB® R2016b