

Differential Equations with Maple

Mth 256 March 4 2001

Bent E. Petersen

Filename: 256winter2001_differential_equations.mws

```
[ > restart;  
[ > with(plots):
```

In this worksheet I give a few hints on using Maple to solve differential equations. Maple can solve most of the differential equations encountered in Mth 256. Before we get to the differential equations let's look at a few preliminaries.

Equations

Maple deals well with expressions. Among the expressions that are of interest to us are equations. In Maple any expression, including an equation, may be assigned to a label. Thus we can take a simple equation such as $A=B$ and assign it to a label for ease in dealing with it. Thus

```
[ > eqn1 := A = B;
```

eqn1 := A = B

Here eqn1 is the label (or name) of the equation $A=B$. Sometimes we wish to extract the left or right (hand) side of an equation.

```
[ > lhs(eqn1);
```

A

```
[ > rhs(eqn1);
```

B

There are devious, sometimes useful, ways to achieve the same thing. For example,

```
[ > subs(eqn1, A);
```

B

Here we have taken the expression A and substituted $A=B$ in it. The result is of course B . Note eqn1 is not altered in any way - it simply provides the specification for what to substitute for what. Note also the subs() approach has the advantage that, unlike rhs(), it correctly handles expressions containing several equations.

Another approach is to use the assign() function. It assigns the right side of each equality in a list to the

left side, that is the left side becomes a label for the right side. Thus

```
> assign(eqn1);
```

```
> A; B;
```

B

B

```
> B:=6; A;
```

$B := 6$

6

In case we want to use A and B again let's unassign them:

```
> unassign('A','B');
```

The single quotes here prevent Maple from evaluating A and B. Otherwise we would be trying to unassign 6, which would produce an error.

Functions

A function in Maple is defined by the "arrow" notation:

```
> f:=x->x^2+2*x-3;
```

$f := x \rightarrow x^2 + 2x - 3$

A suitable expression may be converted to a function by using unapply(). Thus

```
> expr:=x^2+2*x-3;
```

$expr := x^2 + 2x - 3$

```
> g:=unapply(expr,x);
```

$g := x \rightarrow x^2 + 2x - 3$

Functions are evaluated in the usual familiar way, whereas expressions are evaluated by using the substitute command subs(). Thus

```
> f(t); f(6);
```

$t^2 + 2t - 3$

45

```
> subs(x=6,expr);
```

45

One thing to be careful about is the x in the arrow notation above is local to the function we are defining, that is, it is just a dummy variable. Thus

```
> h:=x->expr;
```

$$h := x \rightarrow expr$$

does not define the same function as f above, but instead defines the constant function whose value is $expr$,

```
> h(t); h(6);
```

$$x^2 + 2x - 3$$
$$x^2 + 2x - 3$$

It is best not to be too devious when defining functions. Errors like this can be hard to find!

Maple works well with expressions, but is not quite as competent with functions. If one plans to use Maple to simplify complicated expressions, it is best to leave them as expressions, even if habit makes functions feel more natural to work with.

```
> is(f=g);
```

FAIL

```
> is(f(x)=g(x));
```

true

Note FAIL does not mean false. It means Maple does not know the answer.

The differentiation operator for functions is $D()$. The differentiation operator for expressions is $\text{diff}()$. Thus

```
> D(f);
```

$$x \rightarrow 2x + 2$$

```
> diff(f(x),x);
```

$$2x + 2$$

Note that $\text{diff}()$ is very easy to use for partial derivatives and for higher order derivatives. Thus

```
> diff(arctan(y/x),x,y): simplify(%);
```

$$-\frac{x^2 - y^2}{(x^2 + y^2)^2}$$

Note the use of Maple ditto operator % above. Be very careful! This operator refers to the previously evaluated expression (in time), not the previous expression on the worksheet. It makes a difference since one can move around and evaluate expressions in numerous places on the worksheet, so previous in time need not be the same as previous in location. The two are the same of course if you restrict % to refer to an expression on the same line, as above. That is the safest way to use the % operator.

Differential Equations

We use the dsolve() command to solve differential equations. It returns an expression (which may be empty) which contains equations giving the solution. There are also numerous numeric versions which in a sense return a procedure for approximating a solution.

Here's an example:

```
> ode1:=diff(y(x),x)=x*sec(y(x));
```

$$ode1 := \frac{\partial}{\partial x} y(x) = x \sec(y(x))$$

```
> init1:=y(0)=2;
```

$$init1 := y(0) = 2$$

```
> soln1:=dsolve({ode1,init1},y(x));
```

$$soln1 := y(x) = \arcsin\left(\frac{1}{2}x^2 + \sin(2)\right)$$

If one wants the solution expressed as a function y, that can be achieved as follows:

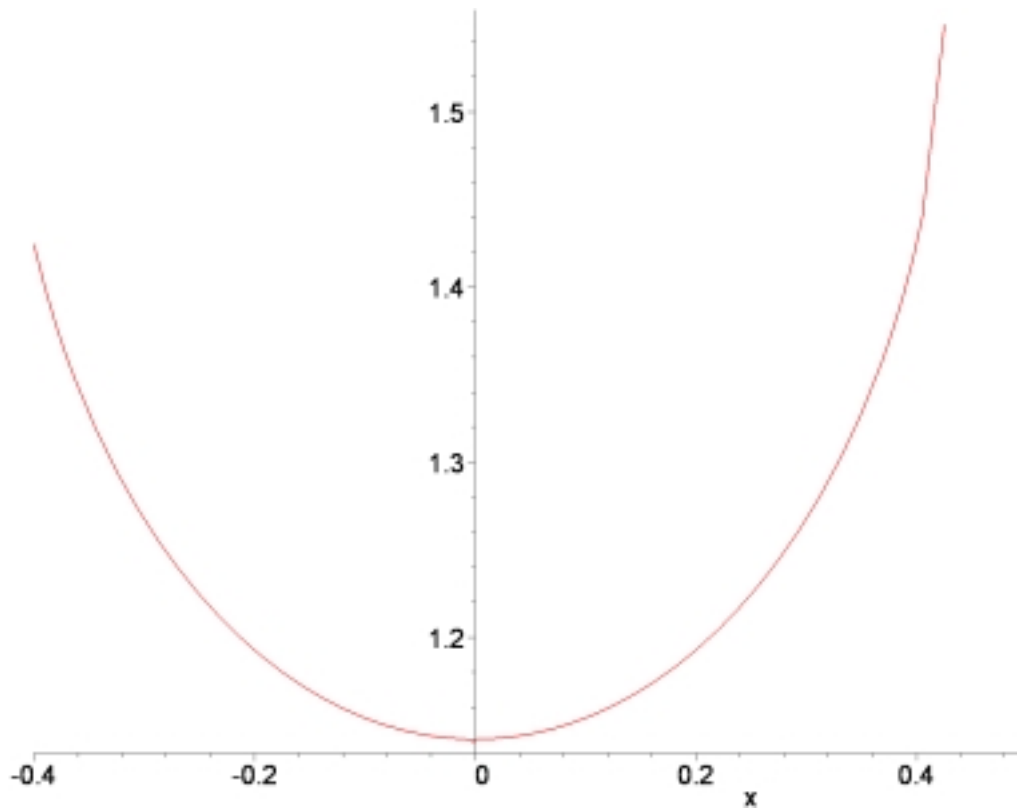
```
> y:=unapply(rhs(soln1),x);
```

$$y := x \rightarrow \arcsin\left(\frac{1}{2}x^2 + \sin(2)\right)$$

The disadvantage here is we have now assigned a value to y. We will not be able to use y again as a "variable" in a differential equation without first unassigning it.

If you just want to plot the solution, or manipulate it in some other way, it is not necessary to convert it to a function. Thus

```
> plot(rhs(soln1),x=-2/5..1/2,color=red,thickness=2);
```



Let's clean up:

```
> unassign(y);
```

System of differential equations

Let's look at a simple system

```
> ode2:=diff(y(t),t)=3*x(t)-2*y(t), diff(x(t),t)=2*x(t)+3*y(t);
```

$$ode2 := \frac{\partial}{\partial t} y(t) = 3 x(t) - 2 y(t), \frac{\partial}{\partial t} x(t) = 2 x(t) + 3 y(t)$$

```
> init2:=x(0)=3,y(0)=-1;
```

$$init2 := x(0) = 3, y(0) = -1$$

```
> soln2:=dsolve({ode2,init2},{x(t),y(t)});
```

$$soln2 := \{x(t) = \frac{3}{2} e^{(\sqrt{13} t)} - \frac{3}{26} \sqrt{13} e^{(-\sqrt{13} t)} + \frac{3}{26} \sqrt{13} e^{(\sqrt{13} t)} + \frac{3}{2} e^{(-\sqrt{13} t)},$$

$$y(t) = -\frac{11}{26} \sqrt{13} e^{(-\sqrt{13} t)} + \frac{11}{26} \sqrt{13} e^{(\sqrt{13} t)} - \frac{1}{2} e^{(\sqrt{13} t)} - \frac{1}{2} e^{(-\sqrt{13} t)}\}$$

The easiest way to extract the individual pieces is to use assign(). Also assign() makes no assumptions

about the order of the solutions.

```
> assign(soln2);
```

Note this assigns x(t) and y(t) as labels for the appropriate expressions.

```
> x(t);
```

$$\frac{3}{2}e^{(\sqrt{13}t)} - \frac{3}{26}\sqrt{13}e^{(-\sqrt{13}t)} + \frac{3}{26}\sqrt{13}e^{(\sqrt{13}t)} + \frac{3}{2}e^{(-\sqrt{13}t)}$$

```
> y(t);
```

$$-\frac{11}{26}\sqrt{13}e^{(-\sqrt{13}t)} + \frac{11}{26}\sqrt{13}e^{(\sqrt{13}t)} - \frac{1}{2}e^{(\sqrt{13}t)} - \frac{1}{2}e^{(-\sqrt{13}t)}$$

It is important to realise that x and y remain unassigned. If you want them to be functions corresponding to the solutions you have to say so explicitly:

```
> x:=unapply(x(t),t); y:=unapply(y(t),t);
```

$$x := t \rightarrow \frac{3}{2}e^{(\sqrt{13}t)} - \frac{3}{26}\sqrt{13}e^{(-\sqrt{13}t)} + \frac{3}{26}\sqrt{13}e^{(\sqrt{13}t)} + \frac{3}{2}e^{(-\sqrt{13}t)}$$
$$y := t \rightarrow -\frac{11}{26}\sqrt{13}e^{(-\sqrt{13}t)} + \frac{11}{26}\sqrt{13}e^{(\sqrt{13}t)} - \frac{1}{2}e^{(\sqrt{13}t)} - \frac{1}{2}e^{(-\sqrt{13}t)}$$

Let's clean up before going to the next example

```
> unassign(x,y);
```

We do not need to unassign x(t) and y(t). Those labels were clobbered when we defined the functions x and y, since, after defining the functions, x(t) became the function x evaluated at t and y(t) became the function y evaluated at t.

Consider now a second order equation

```
> ode3:=diff(y(x),x$2)+4*y(x)=sec(x);
```

$$ode3 := \left(\frac{\partial^2}{\partial x^2} y(x) \right) + 4 y(x) = \sec(x)$$

Note we used x\$2 rather than x,x to indicate the second derivative. It's not a big deal here, but for higher order derivatives the x\$n notation has obvious advantages.

```
> soln3:=dsolve(ode3,y(x));
```

$$\text{soln3} := y(x) = \left(\sin(x) - \frac{1}{2} \ln(\sec(x) + \tan(x)) \right) \sin(2x) + \cos(x) \cos(2x) + _C1 \sin(2x) + _C2 \cos(2x)$$

Note that Maple returns a solution containing arbitrary constants. We could of course have solved an initial value problem instead as above. In fact, we can even have symbolic initial values.

```
> init3:=y(0)=A,D(y)(0)=B;
```

$$\text{init3} := y(0) = A, D(y)(0) = B$$

```
> soln3b:=dsolve({ode3,init3},y(x));
```

```
soln3b := y(x) =
```

$$\left(\sin(x) - \frac{1}{2} \ln(\sec(x) + \tan(x)) \right) \sin(2x) + \cos(x) \cos(2x) + \frac{1}{2} B \sin(2x) + (-1 + A) \cos(2x)$$

Numeric Methods

Maple has a number of numeric method built in that can be used to solve differential equations that Maple can not solve symbolically.

```
> ode4:=diff(y(x),x)=x^2+x*y(x)+y(x)*tan(y(x));
```

$$\text{ode4} := \frac{\partial}{\partial x} y(x) = x^2 + x y(x) + y(x) \tan(y(x))$$

```
> init4:=y(0)=1;
```

$$\text{init4} := y(0) = 1$$

```
> soln4:=dsolve({ode4,init4},y(x));
```

```
soln4 :=
```

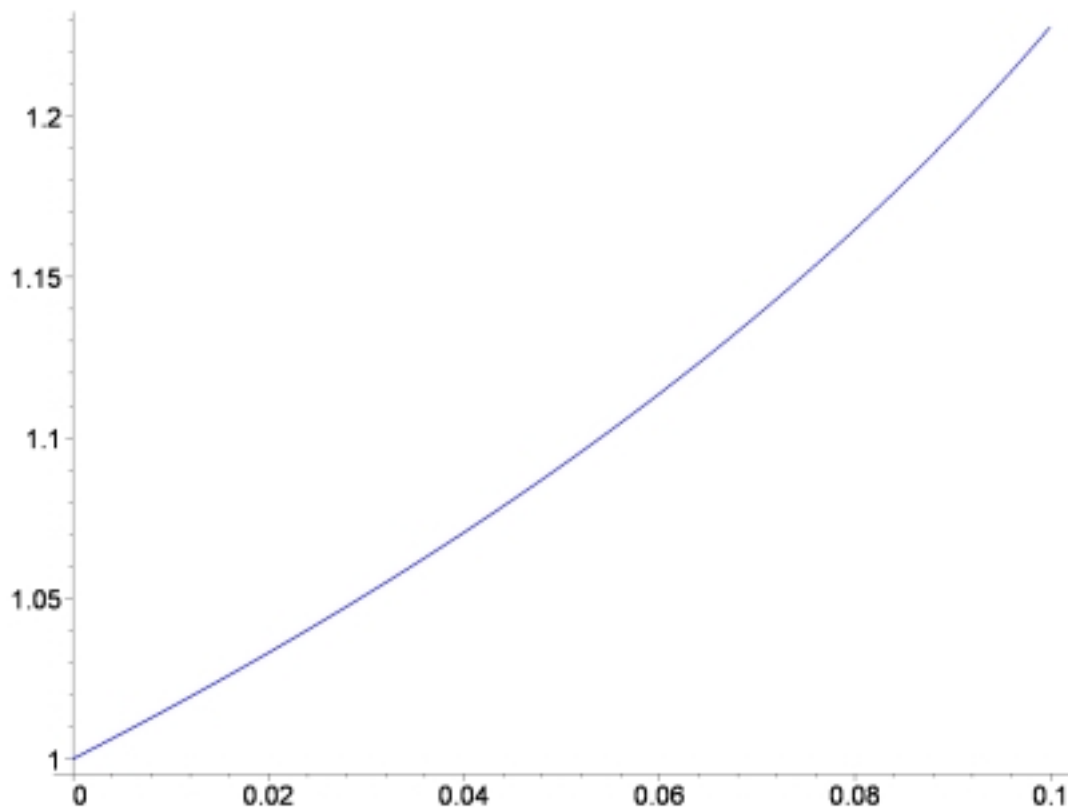
The empty solution indicates Maple could not find a solution.

```
> soln4n:=dsolve({ode4,init4},y(x),numeric);
```

```
soln4n := proc(rkf45_x) ... end
```

This looks mysterious, but fortunately some Maple functions such as odeplot() understand the returned data

```
> odeplot(soln4n,[x,y(x)],0..0.1,thickness=3,color=blue);
```



Higher Order

Here is an example of a simple equation of fifth order

```
> ode5:=diff(y(t),t$5)-4*diff(y(t),t$3)+3*diff(y(t),t$2)=0;
```

$$ode5 := \left(\frac{\partial^5}{\partial t^5} y(t) \right) - 4 \left(\frac{\partial^3}{\partial t^3} y(t) \right) + 3 \left(\frac{\partial^2}{\partial t^2} y(t) \right) = 0$$

```
> init5:=y(1)=2,D(y)(1)=0,(D@@2)(y)(1)=-3,(D@@3)(y)(1)=2,(D@@4)(y)(1)=1;
```

$$init5 := y(1) = 2, D(y)(1) = 0, (D^{(2)})(y)(1) = -3, (D^{(3)})(y)(1) = 2, (D^{(4)})(y)(1) = 1$$

```
> soln5:=dsolve({ode5,init5},y(t));
```

$$soln5 := y(t) = \frac{37}{9} + \frac{13}{3}t - 12 \frac{e^t}{(e^{(1/2)})^2} + \frac{1}{117} (325 + 94\sqrt{13}) e^{(1/2-1/2\sqrt{13})t} e^{(1/2(-1+\sqrt{13})t)}$$

$$- \frac{1}{117} (-325 + 94\sqrt{13}) e^{(1/2+1/2\sqrt{13})t} e^{(-1/2(1+\sqrt{13})t)}$$

The solution is a bit difficult to interpret. Unless you really need the exact solution a floating point approximation might be preferable.

```
> expr5:=evalf(rhs(soln5),6);
```

$$\text{expr5} := 4.11111 + 4.33333 t - 4.41456 e^t + 1.54220 e^{(1.30278 t)} - 1.19014 e^{(-2.30278 t)}$$

Maple's numeric routines default to "Digits" decimal digits of precision. The default is 10, but it can be changed by assigning a different number to Digits, for example, Digits:=20; The routine evalf() converts to floating point using the number of digits specified in Digits. Alternately one can specify the desired precision as the second variable to evalf(). This has the advantage of not altering the precision for the whole worksheet.

An LCR circuit

Consider a simple LCR circuit driven by a voltage $4 \cos(\omega t)$ where $0 < \omega$. Assume initially there is no charge on the capacitor and no current flowing in the circuit. Then at time $t = 0$ we switch on the source of electromotive force. We have

```
> ode6:=L*diff(Q(t),t,t)+R*diff(Q(t),t)+(1/C)*Q(t)=4*cos(omega*t);
```

$$\text{ode6} := L \left(\frac{\partial^2}{\partial t^2} Q(t) \right) + R \left(\frac{\partial}{\partial t} Q(t) \right) + \frac{Q(t)}{C} = 4 \cos(\omega t)$$

```
> init6:=Q(0)=0,D(Q)(0)=0;
```

$$\text{init6} := Q(0) = 0, D(Q)(0) = 0$$

where $Q(t)$ is the charge on the capacitor at time t .

Suppose the inductance is $1/4$ henry, the resistance is 100 ohms and the capacitance is 10^{-6} farads. Then we have

```
> ode6b:=subs(L=1/4,R=100,C=10^(-6),ode6);
```

$$\text{ode6b} := \frac{1}{4} \left(\frac{\partial^2}{\partial t^2} Q(t) \right) + 100 \left(\frac{\partial}{\partial t} Q(t) \right) + 1000000 Q(t) = 4 \cos(\omega t)$$

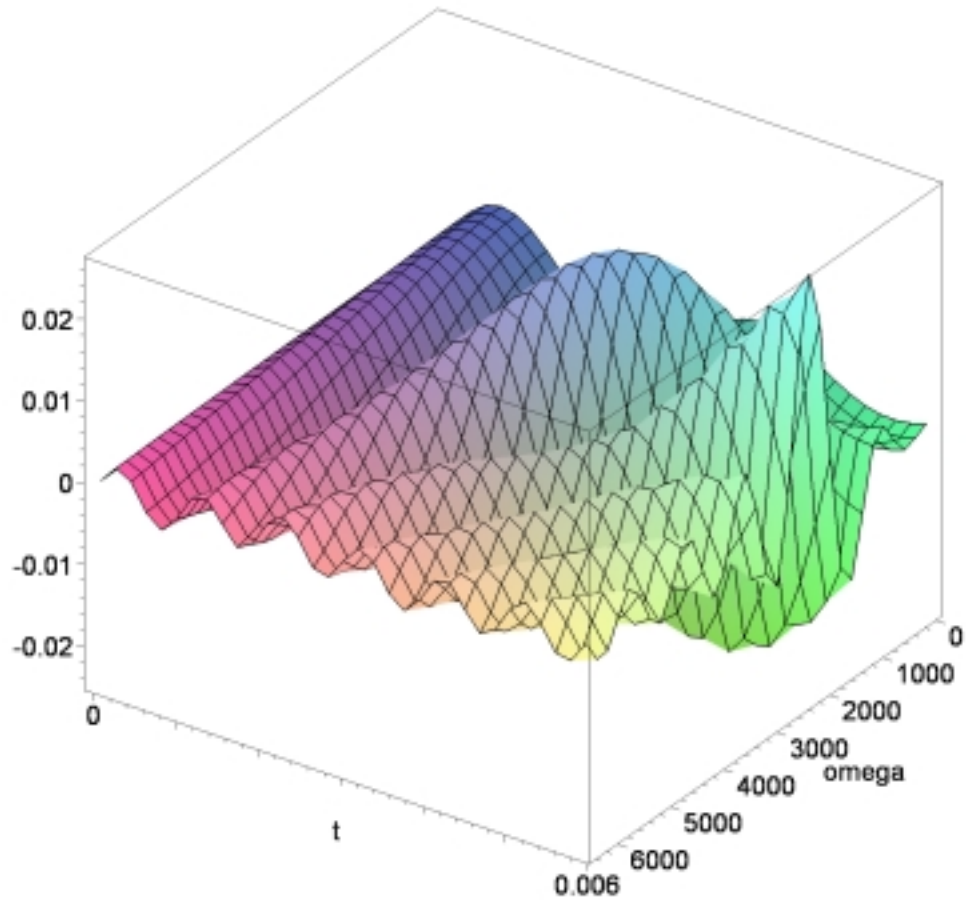
```
> soln6b:=dsolve({ode6b,init6},Q(t));
```

The current flowing in the system is the rate of change of the charge on the capacitor. Thus

```
> expr6:=diff(rhs(soln6b),t):
```

We can think of the current J as a function of t and ω . We have to unapply two variables in this case

```
[ > J:=unapply(expr6,[omega,t]):  
> plot3d(J(omega,t),omega=0..6400,t=0..0.0059,axes=boxed,orientation  
=[35,55],thickness=2,numpoints=1000);
```



Here the resonance peak at $\omega = 2000$ appears to be showing up, as does attenuation of higher frequencies.

```
[ >
```