## Introduction to Maple features relevant to differential equations

This document is based on one prepared by Professor R. Bumby for Math 244, Spring 06. It concentrates on the special features of Maple that are relevant to differential equations. A general introduction to Maple can be found in in the Math 251 Maple instructions; it should be reviewed for an introduction to features of Maple used in Rutgers courses. That introduction, and other useful Maple references, can be reached through links on the Math 244 web page for Spring 2009.

The commands described here are all included in an accompanying *Maple worksheet*. You should download the worksheet and step through the examples to see how the different commands are used. The worksheet has been structured to help identify the role of each Maple command.

**A. Libraries.** Maple groups instructions for treating particular classes of problems into *libraries*. Commands for dealing with differential equations are in the *DEtools* library, and commands for plotting various sorts of graphs, including solutions to differential equations, are in the *plots* library. Later in the course, the *LinearAlgebra* library is also useful for dealing with systems of differential equations. To access these libraries, the *seed files* for this course will begin with a *Setup* section which includes the commands `with(plots):` and `with(DEtools):` , and for labs 4 and 5, `with(LinearAlgebra):`. Note that these commands end with a colon instead of the usual semicolon to suppress output (which would be a list of all objects defined in the library).

**B. Input.** Consider as an example the differential equation

$$\frac{d^2y}{dx^2} - y = x^3. \tag{D}$$

In entering this equation into a Maple worksheet, we must indicate explicitly that the unknown variable $y$ is a function of the variable $x$, by writing $y(x)$ instead of $y$. To describe derivatives, we use Maples *diff* notation: the first derivative is `diff(y(x),x)` and the second derivative is `diff(y(x),x$2)` or `diff(y(x),x,x)`, with similar expressions for higher derivatives. Thus, the differential equation (D) is expressed in Maple notation as `diff(y(x),x,x)-y(x)=x^3` .

For maximum flexibility in using Maple, it is a good idea to introduce specific names for the various equations and initial conditions that one is studying, so that the Maple functions that work with differential equations can be applied using short commands. For example, the differential equation (D) can be defined in Maple, and given the name `de0`, with the instruction

```
de0 := diff(y(x),x,x)-y(x)=x^3;
```

Once this command has been given, the symbol `de0` may be used to whenever we wish to refer to the equation. Note that the first equal sign in this line is preceded by a colon, identifying it as an *assignment* symbol, but the second is not, since it represents part of an *equation*.

We may wish to solve equation (D) subject to certain conditions on $y(x)$, for example, initial conditions such as

$$y(0) = 1, \quad y'(0) = 2. \tag{I0}$$

These conditions can be described to Maple as an *expression sequence*, and given a name so that they can be easily incorporated into different Maple functions, by the command

```
ic0 := y(0)=1,D(y)(0)=2;
```

Note that the syntax used to describe the value of $y'(0)$ in the initial conditions is not the `diff` notation used in the equation itself. Initial conditions involving derivatives higher than the first are more complicated to describe in Maple notation, and we will not do so here, since the problem will not arise in our labs. Should you ever wish to use Maple in a situation where such initial conditions are needed so it can be helpful to set them using the *interactive tools* that are discussed below.

**C. Maple commands for explicit solutions of differential equations.**

• **dsolve.** The basic command for solving differential equations is `dsolve`. If we ask Maple to solve the equation (D), by the command

```
dsolve(de0);
```

it will find the *general solution*, which will involve several integration constants. (Maple identifies a parameter which the system introduces in the course of solving a problem by an initial underscore, so the integration constants for this problem will be called something like $\_C1$ and $\_C2$.) We may also use `dsolve` to solve the initial value problem (D)–(I0) by grouping the equation and the initial conditions together, inside braces, as the argument to `dsolve`:

```
dsolve({de0,ic0});
```

Using other, optional, arguments to `dsolve` we can also force Maple to try particular methods of obtaining the solution. All methods described in Boyce and DiPrima—and many more—are available. The various *options* which may be specified for `dsolve` are described on the help page.

The function `dsolve` can also be used to solve systems of differential equations; an example is given in the accompanying worksheet.

• **odeadvisor.** The Maple command `odeadvisor` allows one to see Maple's classification of *single equations*—possibly of high order. For the most part, this classification should agree with that of Boyce and DiPrima. For example, the command

```
odeadvisor(de0);
```

gives some of the properties of (D). More information on this command can be found on the Maple12 help pages *DEtools,odeadvisor* and *dsolve,education*. The tools described on these pages will be an important part of the Maple projects in this course, so you are urged to return to those help pages often.

• **infolevel.** After the command

```
infolevel[dsolve]:=2;
```

the `dsolve` command will give additional information about its activities. To reset to the default level of information, use `infolevel[dsolve]:=1;` . The function `DEplot` discussed below will sometimes use the function `dsolve`, so that changing the infolevel for `dsolve` may affect the behavior of `DEplot`.

• **dsolve[interactive].** There is a `dsolve[interactive]` function that is of some help in constructing the commands to solve equations and plot solutions. As indicated above, for example, commands giving initial conditions involving derivatives of order two or higher can be generated automatically by `dsolve[interactive]`; they must then be *copied* to an *input line* in order to be reproducible. Thus, from the point of view of our Maple labs, the tool can help *build* commands, but must be *replaced* by those commands in the final worksheet. Unfortunately, if one executes `dsolve[interactive]` more than once, it does not remember from call to call the settings previously entered, so it is necessary to re-enter anything that is not part of the command. Moreover, `dsolve[interactive]` is limited to constructing the input for the `dsolve` command, and uses the `plot` command to graph the solution it finds, contrary to the conventions used in our lab projects, which use `DEplot` (see below). *This command will not be used in the labs of this course, but is included here, and in the accompanying worksheet, for completeness.*

**D. Maple commands for plotting numerical solutions and direction fields.**

• **DEplot for numerical solutions.** Despite the wealth of methods for solving differential equations, there are many equations that cannot be solved in closed form. Maple can use *numerical methods* to solve these equations. The `DEplot` instruction combines these numerical methods with graphical techniques in order to graph the solutions. For example, the command

```
DEplot(de0, y(x), x = -2 ..  2, [[ic0]], title="First equation");
```

produces a graph the solution of (D)–(I0) that we have studied above. The *title* is optional in Maple, but expected in course lab projects. One curious feature of this command is that initial conditions must be given as a *list of lists*, so our expression sequence `ic0` needs to be bracketed *twice*. However, a second set of initial conditions,

$$y(0) = 1, \quad y'(0) = 0, \tag{I0a}$$

could be added to *the same* graph (with a different title) via

```
ic0a:=y(0)=1,D(y)(0)=0;
DEplot(de0,y(x),x=-2..2,[[ic0],[ic0a]],title="Two solutions");
```

• **DEplot for direction fields.** For first order equations, `DEplot` can draw a a *direction field*, or can combine this field with plots of solutions. To illustrate we consider the first order equation and initial conditions

$$\frac{dy}{dx} = -y + \frac{1}{1+e^x}, \tag{D1}$$

$$y(0) = -2, \tag{I1a}$$

$$y(0) = 1. \tag{I1b}$$

The following Maple commands, included in the worksheet, introduce the equation, find its general solution, and find its solution with initial condition (I1a):

```
de1:= diff(y(x),x) = - y(x) + 1/(1 + exp(x));
ans1:=dsolve(de1,y(x));#ans1 denotes an equation
s1:=eval(y(x),ans1); #the solution can be recovered
ans1a:=dsolve({de1,y(0)=-2},y(x));
s1a:=eval(y(x),ans1a);
```

The next commands plot the direction field alone, and with the two solutions satisfying (I1a,I1b):

```
DEplot(de1, y(x), x=-1..5, y=-6..4);
initval:={[y(0)=-2],[y(0)=1]};
DEplot(de1, y(x), x=-1..5, initval,y=-6..4);
```

• **odeplot, dfieldplot, and the numeric option for dsolve.** There is a `numeric` option to the `dsolve` command which gives numeric solutions of differential equations, as well as an `odeplot` function for graphing solution curves that have been previously calculated numerically, and a `dfieldplot` for direction fields. For illustrative purposes the worksheet contains examples of these commands:

```
plot(s1a,x=-1..5);
s1n:=dsolve({de1,y(0)=-2},y(x),numeric);
odeplot(s1n,[x,y(x)],-1..5);
```

```
dfieldplot(de1,y(x), x=-1..5,y= -6..4);
```

However, the `DEplot` command will be used for all these purposes in this course, since it is both powerful and easy to use.

**E. Linear Algebra.** This course will use the newer `LinearAlgebra` package instead of the older `linalg` package. A guide to using this package will be part of Lab 4.

End of Maple instructions document