Lab 4: The Linear Systems

This Maple lab is closely based on earlier versions prepared by Professors R. Falk and R. Bumby of the Rutgers Mathematics department.

Introduction. In this lab we use Maple to find *eigenvalues* and *eigenvectors* of matrices, and solve linear systems of algebraic equations and systems of first order linear differential equations. We also obtain pictures of the slope fields of these equations in the *phase plane*.

The first step is to obtain the *seed file* from the web page and save it in your directory on eden. You will be asked to execute the Maple commands that have been placed in this worksheet and to add some of your own. Please turn in only the printout of your main Maple worksheet. Include explicit answers to all questions asked, using the *text* feature of Maple to insert them in the worksheet. Remove any extraneous material from the worksheet.

Please obtain the seed file from the web page and save it in your directory on eden, then prepare the Maple lab according to the instructions and hints in the introduction to Lab 0. Turn in only the printout of your Maple worksheet, after removing any extraneous material and any errors you have made. For Lab4 you wil also need to obtain a *supplementary worksheet* that contains some commands which you are asked to try out but which should not appear in the final lab.

0. Setup. As usual, the seed file begins with commands which load the required Maple packages: `with(plots):` and `with(DEtools):`. In this lab we also require a Linear Algebra package. There are now two such packages in Maple, not completely compatible. The newer package, called `LinearAlgebra` is used here since it has more respect for its user. The seed file also includes the definitions of two matrices, A and B:

```
A:= <<10,0>|<-2,6>|<-7,6>>;
B:= <<14|8>,<2|14>,<-5|10>>;
```

1. Matrix Operations. We first try some operations. Since some will give errors, and some may give unexpected results, these will be investigated in the *supplementary worksheet*. A few examples are already in the supplementary worksheet, but you should add others to allow a full discussion of these operations. Some of these examples will lead to errors. The errors will find their place in the *discussion*. The Maple commands leading to them *should not appear in the main worksheet*, but your comment should include a description of the failed command and your interpretation of the error message. You should consult the help pages to find properties of the operations used in these statements.

The goal of the comment is to summarize how + and . behave as operators on matrices. If you have any doubt about your interpretation of a result, be sure to consult *Maple help*. In most cases, a *topic search* will give the pages that you need, but there is also a *text search* that will find all mention of the word (those of you who know Professor Zeilberger will find it interesting to do a text search for his name in Maple help). Here are the questions to guide your discussion:

(1) How is `M1+M2` computed when `M1` and `M2` are matrices? Is it *always defined*? If not, how does Maple indicate that it is not defined?

(2) How is `M+c` computed when `M` is a matrix and `c` is a scalar constant? Is it *always defined*? If not, how does Maple indicate that it is not defined?

(3) How is `M1.M2` computed when `M1` and `M2` are matrices? Is it *always defined*? If not, how does Maple indicate that it is not defined?

(4) How is `M.c` computed when `M` is a matrix and `c` is a scalar constant? Is it *always defined*? If not, how does Maple indicate that it is not defined? What is `M.t` when `M` is a matrix and `t` is a symbol? In particular, is it a matrix? If it is a matrix, asking for a particular entry will return the expected *simple* expression. Any other response suggests that you have

not created a matrix. The `whattype` command will tell you how Maple has built the object representing an expression by returning the *top level* data type of the internal representation. The *Maple help pages* give more information. Can one use $*$ in place of $\cdot$ to perform any of these multiplications?

(5) How is `M^c` computed when `M` is a matrix and `c` is an integer constant (possibly negative)? Is it *always defined*? If not, how does Maple indicate that it is not defined?

2. Eigenvalues and eigenvectors. The following lines appear in the seed file. They use the `Eigenvectors` command to obtain eigenvectors *and* eigenvalues of three related matrices.

```
M1:=A.B;
M2:=B.A;
M3:=M1 ^ (-1);
(Vals1,Vecs1):=Eigenvectors(M1);
(Vals2,Vecs2):=Eigenvectors(M2);
(Vals3,Vecs3):=Eigenvectors(M3);
C:=<<1|1|0>,<0|1|0>,<0|0|2>>;
Eigenvectors(C);
```

(1) Begin the discussion by stating the relation between the *eigenvalues* of `M1` and `M2`. *Note*: The relation between the eigenvalues of `M1` and `M2` is an example of a general theorem, which we do not give.

(2) Continue the discussion by stating the relation between the *eigenvalues* of `M1` and `M3`. Note: This also depends on a relation between the *eigenvectors*.

(3) The discussion section of this part should also discuss the significance of a zero column in the matrix of eigenvectors of C, since we know that *the zero vector is never an eigenvector.* Hint: see Section 7.8 of the text.

3. Real matrix Exponentials. The `LinearAlgebra` package provides a good interface to do numerical work with matrices, but it needs to be cajoled into doing symbolic work. This can be done with the `map` function. This constructs a new matrix without affecting the original one.

There is also a `Map` function that modifies `Matrices` in place. Be careful to distinguish these two instructions. If one is used where the other is needed, the results will be *very different from what you intend* and *subsequent instructions will not work correctly.*

When eigenvalues of an n by n matrix M are real and distinct, the eigenvectors form a *basis* of the space $\mathbb{R}^n$. For each eigenvalue λ_i, the corresponding eigenvector $\mathbf{v}_i$ is the numerical vector of coefficients of $e^{\lambda_i t}$ in a solution of $d\mathbf{y}/dt = M\mathbf{y}$. If we let Φ denote the matrix whose i^{th} column is $\mathbf{v}_i$ and e^{Dt} the *diagonal* matrix whose entries are $e^{\lambda_i t}$, then Φe^{Dt} is a solution of the differential equation whose value at $t = 0$ is the *constant* matrix Φ. Thus, given any *vector* of initial conditions $\mathbf{y}(0)$, $\Phi^{-1}\mathbf{y}(0)$ is the vector of coefficients of the special solutions $\mathbf{v}_i e^{\lambda_i t}$. This leads to the expression

$$\mathbf{y} = \Phi e^{Dt}\Phi^{-1}\mathbf{y}(0).$$

The construction of the matrix $e^{Mt} = \Phi e^{Dt}\Phi^{-1}$, and checking that it has all required properties, when M is the 3×3 matrix `M2` from Section 2, uses the following Maple commands (requiring results found in Section 2) that are included in the seed file.

```
EL2:=DiagonalMatrix(map(c->exp(c*t),Vals2));
Y2:=Vecs2.EL2.Vecs2 ^ (-1);
DY2:=map(diff,Y2,t);
```

```
MY2:=M2.Y2;
DY2-MY2; # checks equation if zero matrix
subs(t=0,Y2); # checks initial condition if identity matrix
```

The matrix `Y2` is the *matrix exponential* (also known as the *fundamental matrix*). (Note that this matrix does not depend on the scaling of the individual eigenvectors $\mathbf{v}_i$). In this computation, `Y2` is found by translating the definition of e^{Mt} into Maple commands. Then, there is a check that it satisfies both the differential equation and the initial conditions. The nature of each check is included as a comment on the line giving the check. Since *exact* values are available, these checks will be exact. If numerical approximations of the eigenvalues were used, these checks could differ slightly from the expected values.

To illustrate the connection between the matrix exponential and solutions of individual initial value problems, you should:

(1) use the matrix `Y2` to find the solution of

$$\frac{d\mathbf{y}}{dt} = M_2\mathbf{y} \quad \text{with} \quad \mathbf{y}(0) = \begin{pmatrix} 2 \\ -1 \\ 5 \end{pmatrix}$$

(2) check that this vector satisfies the differential equations; and check that this vector satisfies the initial conditions.

4. Saddle points and nodes. Consider the matrices

$$M_{4A} = \begin{pmatrix} 4 & -2 \\ 3 & -3 \end{pmatrix} \quad \text{and} \quad M_{4B} = \begin{pmatrix} 2 & -5 \\ 4 & -7 \end{pmatrix}.$$

For each, we will use the method of Section 2 (which will be a little easier here because these systems have 2×2 matrices) to solve the equation $d\mathbf{y}/dt = M\mathbf{y}$ with initial conditions

$$(a) \quad \mathbf{y}(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad (b) \quad \mathbf{y}(0) = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \qquad (c) \quad \mathbf{y}(0) = \begin{pmatrix} -1 \\ -1 \end{pmatrix}.$$

A *graphical check* of the solution involves plotting the slope field of the equation in a *phase plane* whose coordinates are the components of $\mathbf{y}$ and superimposing a *parametric plot* of the *trajectories* of the solutions. Because the equations are *autonomous*, all solutions starting at a point on one of these trajectories will follow the trajectory — the only difference being the value of t at which it visits a particular point. This uses features not explored previously in this course, so the seed file contains instructions to construct the individual plots for the first of these equations. You will need to combine these plots with a `display` command and repeat the process, with modifications, for the second equation. Here are the instructions that appear in the seed file for getting graphs for the first system. *You will need to provide the corresponding instructions for the second system.*

```
M4A:=<|relax<4,3>|ttbar<-2,3>|relax>;
(Vals4A,Vecs4A):=Eigenvectors(M4A);
EL4A:=DiagonalMatrix(map(c->exp(c*t),Vals4A));
Y4A:=Vecs4A.EL4A.Vecs4A ^ (-1);
Y4Aa:=Y4A.<1,0>;
Y4Ab:=Y4A.<-1,1>;
Y4Ac:=Y4A.<-1,-1>;
```

```
VecVar:=<y1(t),y2(t)>;
listVar:=convert(VecVar,'list');
tvals:=t=-2..2;
range4:=y1=-2..2,y2=-2..2;
eq4A:=[diff(VecVar[1],t)=(M4A.VecVar)[1],
diff(VecVar[2],t)=(M4A.VecVar)[2]];
Field4A:=DEplot(eq4A,listVar,tvals,range4,color=GREEN):
Sol4A:=plot([[Y4Aa[1],Y4Aa[2],tvals],
[Y4Ab[1],Y4Ab[2],tvals],
[Y4Ac[1],Y4Ac[2],tvals]],
range4,color=[BLACK,RED,BLUE]):
display(Field4A,Sol4A,title="Equation 4A");
```

We need $y_1(t)$ and $y_2(t)$ to be combined into a `Vector` in order to use the `LinearAlgebra` package, and into a `list` to serve as a argument of the `DEplot` function. The `convert` instruction assures that the related objects will have the *same contents* in *different formats*. The entries of the vector appear in the differential equation because the `diff` operation in Maple only applies to scalar functions. (The same effect could be obtained with `map`, but the approach here is a good alternative when there are only two entries.) The expression `Sol4A` plots a *list* of objects, each of which is a parametric description of the trajectory of a solution in the phase plane. Colors are assigned to the plots in the same order that they appear in the list.

The *Discussion* portion of this section should investigate the role of these plots in checking this method of solving differential equations. In particular:

(1) do the claimed solution curves look like they follow the slope field of the equation?
(2) identify the type of equilibrium point at the origin for these two examples:
  (∗) for which of these equations has is the origin a saddle point?
  (∗) for the other equation, is the origin *stable* (i.e., attracting) or *unstable* (i.e., repelling)?
  (∗) do the shapes of the solution agree with what you expect?
(3) describe how both the eigenvalues of the coefficient matrix and the slope field illustrate your classification.

Finally, for the *saddle point* example, construct a second graph by adding to the graph already constructed the four "special" trajectories which lie along straight lines.

5. Spiral points. Consider the equation

$$\frac{dY}{dt} = \begin{pmatrix} 2 & 5 \\ -2 & 0 \end{pmatrix} Y. \qquad (S)$$

It can be shown that, if M is a 2×2 real matrix with eigenvalues $a \pm bi$ ($b \neq 0$) and J is defined by $M = aI + bJ$, then

$$e^{Mt} = e^{at}\big((\cos bt)I + (\sin bt)J$$

The seed file implements this solution in Maple for (S), leading to a matrix `Y5`. You should use methods explored in previous sections to *verify* that this matrix is e^{Mt} by

(1) showing that it *satisfies* both *differential equation* and *initial conditions* characterizing the matrix exponential, and by *illustrating* the solutions with the same initial values (a), (b) and (c) used in Section 4, and by

(2) graphing these solutions superimposed on a slope field of this equation.

Here are the instructions leading to `Y5`.

```
M5:=<|relax<2|ttbar|relax5>,<-2|ttbar|relax0>|relax>;
E5:=Eigenvalues(M5);
(a5,b5):=Re(E5[1]),Im(E5[1]);
J5:=(1/b5).(M5-a5);
Y5:=exp(a5*t)*cos(b5*t)*IdentityMatrix(2)+sin(b5*t)*J5;
```

In the expression for `Y5` the factor `IdentityMatrix(2)` is currently essential since only *constant* scalars are automatically added to matrices.

End of Lab 4