### Maple Recipes for Linear Systems of DE’s

**Characteristic polynomial of a matrix**

```maple
with(linalg):
A := matrix([[3,1,0], [0,3,1], [0,0,3]]);
charpoly(A, lambda);
```

The characteristic polynomial is the determinant

```
| A - \lambda I | =
| 3 - \lambda 1 0 |
| 0 3 - \lambda 1 |
| 0 0 3 - \lambda |
```

**Exact eigenvalues of a matrix A**

```maple
with(linalg):
A := matrix([[3,1,0], [1,3,1], [0,1,3]]):
ev := eigenvals(A);
map(evalf, [ev]);
ev := 3, 3 + \sqrt{2}, 3 - \sqrt{2}
```

Eigenvalues are solutions of the characteristic equation \( |A - \lambda I| = 0 \). The `map` function can be used to apply `evalf` to each eigenvalue. `[ev]` converts a set to a list since `map` doesn’t work on sets.

**Finding eigenvectors of a matrix A from null spaces**

```maple
A := matrix([[3,1,0], [1,3,1], [0,1,3]]):
Id := matrix([[1,0,0], [0,1,0], [0,0,1]]):
lambda := [eigenvals(A)]:
k1 := nullspace(A-lambda[1]*Id):
k2 := nullspace(A-lambda[2]*Id):
k3 := nullspace(A-lambda[3]*Id):
lambda, op(k1), op(k2), op(k3);
```

The nullspace of \( A - \lambda_i I \) is the solution space for \((A-\lambda_i I)k = 0\). We use `op` since `nullspace` returns a set. In this example we get 3 linearly independent eigenvectors.

**Diagonalizing a matrix from its eigenvectors**

```maple
P := transpose(matrix([k1,k2,k3]));
D := map(simplify, evalm(inverse(P) &* A &* P));
op(P), op(D);
```

Since we get 3 linearly independent eigenvectors we can make a matrix \( P \) using them as columns so \( P^{-1}AP \) is the diagonal matrix with the eigenvalues on the diagonal. The `op` function is used here to print the matrices instead of their names. Note that `&*` denotes matrix multiplication.
Exact eigenvalues, multiplicities, eigenvectors of a matrix $A$

\[ A := \text{matrix}([[3,2,4], [2,0,2], [4,2,3]]) \]
\[ \text{eigenvects}(A); \]
\[ [8, 1, \{(2, 1, 2)\}], [-1, 2, \{[1, -2, 0], [0, -2, 1]\}] \]
The output is a sequence of lists. Each list has three entries: the first is the eigenvalue $\lambda$, the second is its multiplicity, and the third is a set of the eigenvectors, each given as a list.

Numerical eigenvalues and eigenvectors of a matrix $A$

\[ A := \text{matrix}([[3,1,0], [1,3,1], [0,1,3]]) \]
\[ \text{lambda} := \text{evalf}(	ext{Eigenvals}(A)); \]
\[ \text{lambda} := \text{evalf}(	ext{Eigenvals}(A,'P')); \]
\[ \text{print}(P); \]
\[ \lambda := [1.585786438, 3.000000001, 4.414213563] \]
\[ \begin{bmatrix} \cdot5000000000 & -.7071067811 & .5000000002 \\ -.7071067813 & -10^{-11} & .7071067816 \\ .5000000001 & .7071067816 & .5000000002 \end{bmatrix} \]
The capital $E$ gives the inert form which can be evaluated numerically. $P$ will be defined as the matrix whose columns are the eigenvectors. It is very important to unassign $P$ if you are going to use it again for a new matrix $A$.

Exact solution of $x' = Ax$

\[ a11 := 3; \quad a12 := 1; \quad a21 := 1; \quad a22 := 3; \]
\[ \text{eq1} := \text{diff}(x(t),t) = a11*x(t) + a12*y(t); \]
\[ \text{eq2} := \text{diff}(y(t),t) = a21*x(t) + a22*y(t); \]
\[ \text{vars} := \{x(t), y(t)\}; \]
\[ \text{sol} := \text{dsolve}([\text{eq1,eq2}, \text{vars}]); \]
\[ \text{sol} := \{y(t) = -e^{2t} + e^{4t}, x(t) = e^{2t} + e^{4t}\} \]

Solution of initial value problem $x' = Ax, x(t_0) = x_0$

\[ t0 = 0; \quad x0 := 1; \quad y0 := 0; \]
\[ \text{sol} := \text{dsolve}(\{\text{eq1,eq2,x(t0)=x0,y(t0)=y0}, \text{vars}\}); \]
\[ \text{sol} := \{y(t) = -1/2 e^{2t} + 1/2 e^{4t}, x(t) = 1/2 e^{2t} + 1/2 e^{4t}\} \]

It is very important here to use $x(t)$ and $y(t)$ instead of just $x$ and $y$. The solutions contain arbitrary constants denoted by $C1$ and $C2$.

If the initial conditions are $x(t_0) = x_0$ (which means $x(t_0) = x_0, y(t_0) = y_0$) then sol is the particular solution.
Evaluation of solutions at particular values of $t$

To evaluate the solutions $x(t)$ and $y(t)$ use the result $\text{sol}$ returned by $\text{dsolve}$ to obtain $x$ and $y$ as functions of $t$.

\[
x := \text{unapply}(\text{subs(}\text{sol}, x(t), t));
\]
\[
y := \text{unapply}(\text{subs(}\text{sol}, y(t), t));
\]
\[
x(0.5), y(0.5), x(1.0), y(1.0);
\]

Plotting the exact solution of initial value problem

Use the $\text{plot}$ and the functions $x$ and $y$ obtained above to plot $x$ as a function of $t$ and $y$ as a function of $t$.

Numerical solution and plotting of a single DE $x'(t) = f(x,t)$

Here $\text{de}$ and $\text{ic}$ are the DE and initial condition expressed as equations. $\text{dsolve}$ returns a procedure $s$ which can be evaluated to obtain values of the solution, or the $\text{odeplot}$ function in the $\text{plots}$ library can be used to plot the solution $x(t)$ for the range $0 \leq t \leq 10$.

Numerical solution and plotting of a system $x' = f(x,t)$

This is a non-linear system of two first order DEs. The system, solution functions, and initial conditions are specified as the expression sequences $\text{sys}$, $\text{funcs}$ and $\text{ic}$. Here $\text{dsolve}$ returns a procedure $s$ which can be evaluated to obtain numerical solution values, or the functions $x(t)$ and $y(t)$ can be plotted using $\text{odeplot}$ and the range $0 \leq t \leq 30$. 

\[
\begin{align*}
x1 & := t \rightarrow \frac{1}{2} e^{(2t)} + \frac{1}{2} e^{(4t)} \\
y1 & := t \rightarrow \frac{1}{2} e^{(4t)} - \frac{1}{2} e^{(2t)}
\end{align*}
\]

5.053668964, 2.335387136, 30.99360307, 23.60454697

\[
\begin{align*}
x & := \text{unapply}(\text{subs(}\text{sol}, x(t), t)) \\
y & := \text{unapply}(\text{subs(}\text{sol}, y(t), t))
\end{align*}
\]

\[
\begin{align*}
x(0.5), y(0.5), x(1.0), y(1.0);
\end{align*}
\]

\[
\begin{align*}
[\text{t=0, } x(t) = 1.], [\text{t=.1, } x(t) = 1.093667858860398] \\
[\text{t=0, } y(t) = 2.], [\text{t=.1, } y(t) = 1.99667082362999, y(t) = 1.990041540703626]
\end{align*}
\]
Plotting solutions in phase space

with(DEtools): with(linalg): 

a := 1: b := 1: c := 1: d := -1: 
de1 := D(x)(t) = a*x(t)+b*y(t): 
de2 := D(y)(t) = c*x(t)+d*y(t): 
DEplot([de1,de2],[x,y], t=-1.5..1.5, 
[[x(0)=0,y(0)=-1],[x(0)=0,y(0)=-0.5], 
[x(0)=0.5,y(0)=1], 
[x(0)=-1,y(0)=0], 
[x(0)=1,y(0)=0]], 
x = -1.5..1.5, y=-1.5..1.5, 
color=magenta,thickness=0, 
style=line, linestyle=1, 
linecolor=red, arrows=MEDIUM);

The exponential of a matrix

with(linalg): 

A := matrix([[3,2,4], [2,0,2], 
[4,2,3]]); 
B := exponential(A,t); 
The matrix exponential $e^{At}$ is not easy to calculate unless $A$ is a diagonal matrix. An important property is $e^{0t} = I$, the $n \times n$ identity matrix. Verify this for the matrix $B$.

\[
B := \begin{bmatrix}
\frac{5}{9} e^{-t} + \frac{4}{9} e^{8t} & \frac{2}{9} e^{8t} - \frac{2}{9} e^{-t} & \frac{4}{9} e^{8t} - \frac{4}{9} e^{-t} \\
\frac{2}{9} e^{8t} & \frac{8}{9} e^{-t} + \frac{1}{9} e^{8t} & \frac{2}{9} e^{8t} - \frac{2}{9} e^{-t} \\
\frac{4}{9} e^{8t} + \frac{4}{9} e^{-t} & \frac{2}{9} e^{8t} & \frac{5}{9} e^{-t} + \frac{4}{9} e^{8t} \\
\end{bmatrix}
\]

The columns of $e^{At}$ are solutions of $x' = Ax$

v1 := col(B,1); 
map(diff, v1, t); 
multiply(A,v1); 
The \texttt{col} function is used to extract the first column of $B$. Then \texttt{map} is used to differentiate each component of $v1$. Finally \texttt{multiply} shows that $v1$ is a solution of $x' = Ax$. 

\[
v1 := \left[ \frac{5}{9} e^{-t} + \frac{4}{9} e^{8t}, \frac{2}{9} e^{8t} - \frac{2}{9} e^{-t}, \frac{4}{9} e^{8t} - \frac{4}{9} e^{-t} \right] \\
\left[ \frac{5}{9} e^{-t}, \frac{32}{9} e^{8t}, \frac{16}{9} e^{8t} + \frac{2}{9} e^{-t}, \frac{32}{9} e^{8t} + \frac{4}{9} e^{-t} \right] \\
\left[ \frac{5}{9} e^{-t} + \frac{32}{9} e^{8t} + \frac{16}{9} e^{8t} + \frac{2}{9} e^{-t}, \frac{32}{9} e^{8t} + \frac{4}{9} e^{-t} \right]
\]
Fundamental matrix of $x' = Ax$

\[
\begin{align*}
col1 & := [2*exp(8*t),1*exp(8*t),2*exp(8*t)]: \\
col2 & := [1*exp(-t),-2*exp(-t),0]: \\
col3 & := [0,-2*exp(-t),1*exp(-t)]: \\
psi & := concat(col1,col2,col3);
\end{align*}
\]

Use 3 linearly independent solutions of $x' = Ax$ as the columns of a matrix $\Psi(t)$. This matrix is called a fundamental matrix. It satisfies $\Psi(t)' = A\Psi(t)$.

\[
\psi := \begin{bmatrix}
2e^{(8t)} & e^{(-t)} & 0 \\
e^{(8t)} & -2e^{(-t)} & -2e^{(-t)} \\
2e^{(8t)} & 0 & e^{(-t)}
\end{bmatrix}
\]

\[e^{At} \text{ is the fundamental matrix } \Psi(t)\Psi^{-1}(0)\]

\[
\begin{align*}
psi0 & := subs(t=0,op(psi)); \\
Bc & := multiply(psi, inverse(psi0));
\end{align*}
\]

This verifies the property that $e^{At} = \Psi(t)\Psi^{-1}(0)$. Note the use of \texttt{op} again: psi is the name of the matrix and \texttt{op}(psi) is the matrix.