Homogeneous Linear Systems of ODEs in Matlab

Kurt Bryan and SIMIODE

This script illustrates how to analyze homogeneous linear systems of ODEs in Matlab.

Example System: Consider the homogeneous linear system of ODEs

```
syms x1(t) x2(t); %Declare functions
de1 = diff(x1(t),t) == x1(t) + 3*x2(t)
de2 = diff(x2(t),t) == 3*x1(t) + x2(t)
```

for functions x1(t) and x2(t), with initial data x1(0) = 2, x2(0) = 6.

Solution via dsolve: The solution can be obtained with Matlab's ***dsolve*** command:

```
sol = dsolve([de1, de2],[x1(0)==2,x2(0)==6]);
x1sol(t) = sol.x1
x2sol(t) = sol.x2
```

Solution via Eigen-analysis: The system above is of the form **x' = Ax** where **x** = <x1,x2> and

```
A = sym([1 3;3 1]); %Here "A" is a symbolic matrix. We could also take A = [1 3;3 1] in which c
```

We solve using the eigenvector/value analysis of Section 6.2.2. The eigenvalues and vectors of A can be computed as

```
[P,D] = eig(A)
```

The columns of P are the eigenvectors and D is a diagonal matrix whose diagonal consists of the corresponding eigenvalues.

To construct a solution with the desired initial data, let

```
x0 = sym([2;6])
```

be a symbolic column vector with the desired initial data .As per the procedure in Section 6.2.2, we can construct the solution to this ODE system by first solving **Pc = x0** for c, which in Matlab takes the form

```
c = P\x0
```

From (6.17) in the text the solution is then

```
xgen(t) = c(1)*exp(D(1,1)*t)*P(:,1) + c(2)*exp(D(2,2)*t)*P(:,2)
```

Example 2: Consider the system

```
syms x1(t) x2(t); %Declare functions
de1 = diff(x1(t),t) == -5*x1(t) + 6*x2(t)
de2 = diff(x2(t),t) == -3*x1(t) + x2(t)
```

with initial data x1(0) = 1, x2(0) = 2. This system is governed by the matrix

```
A = sym([-5 6;-3 1])
```

Compute the eigenvalues/vectors as

```
[P,D] = eig(A)
```

Following the analysis in the previous example and using the general solution (6.17) in the text, the desired initial conditions are obtained by taking the c[k] as the components of the vector c that satisfies **Pc = x0** where **x0** = <1,2> (equation (6.18)). Compute

```
x0 = sym([1;2]);
c = P\x0
xgen(t) = c(1)*exp(D(1,1)*t)*P(:,1) + c(2)*exp(D(2,2)*t)*P(:,2)
```

We can obtain a real-valued solution (and we know the solution is real-valued) by telling Matlab to assume t is real with "syms t real" and then picking off the real part of xgen(t):

```
syms t real;
real(expand(xgen(t)))
```