3. 3x3 Example
Customizing plots in MATLAB

Start of transcript. Skip to the end.

0:00 / 2:23 2.0x

MATLAB contains a variety of functions for creating common visualizations, along with

customizable options for appearance and annotation to help make them clearer to others.

So, how do we customize plots in MATLAB?

Let's start by looking at a line plot of average daily temperatures.

LTI Consumer (External resource) (1.0 points possible)

3x3 System

Let's now revisit the case where we have three identical tanks. This was described by the following system of ODEs

$$\frac{dh_1}{dt} = a_{12}(h_2 - h_1) + a_{13}(h_3 - h_1),$$

$$\frac{dh_2}{dt} = a_{23}(h_3 - h_2) + a_{12}(h_1 - h_2),$$

$$\frac{dh_3}{dt} = a_{23}(h_2 - h_3) + a_{13}(h_1 - h_3).$$

We can then recast this in matrix form as

$$\dot{\mathbf{x}} = \begin{bmatrix} -a_{12} - a_{13} & a_{12} & a_{13} \\ a_{12} & -a_{12} - a_{23} & a_{23} \\ a_{13} & a_{23} & -a_{13} - a_{23} \end{bmatrix} \mathbf{x}$$

where $\mathbf{x} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}$. Let's firstly set $a_{12} = a_{13} = a_{23} = 1$, and use the initial conditions

$$\mathbf{x}(0) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

We know that the solution can be written in the form

$$\mathbf{x}(t) = c_1 e^{\lambda_1 t}\mathbf{v}_1 + c_2 e^{\lambda_2 t}\mathbf{v}_2 + c_3 e^{\lambda_3 t}\mathbf{v}_3.$$

The c_i, λ_i and v_i are a lot easier to find using MATLAB. Use the template below to construct the analytic solution to the above system of equations using MATLAB, then plot the solution in the range $0 < t < 4$.

Your Script

Save Reset MATLAB Documentation (https://www.mathworks.com/help/)

```
% Input the 3x3 matrix describing your linear system as a variable A
A = [-2,1,1;1,-2,1;1,1,-2];
% Input your initial conditions as a column vector x0
x0 = [0;1;0];
```

```
x0 =[0;1;0];
% Use eig(A) to find the eigenvalues and eigenvectors of A
% Define the eigenvectors as column vectors v1, v2 and v3,
% and the eigenvalues as lambda1, lambda2, lambda3
% so that A*v1 = lambda1*v1, etc.
[V,D] = eig(A);
v1 = V(:,1);
v2 = V(:,2);
v3 = V(:,3);
lambda1 = D(1,1);
lambda2 = D(2,2);
lambda3 = D(3,3);
% Calculate the column vector c = [c1;c2;c3] from the initial conditions using inv
c = inv(V)*x0;
c1 = c(1,1); c2 = c(2,1); c3 = c(3,1);
%%% Define a row vector t with 100 equally spaced entries,
%%% beginning with 0 and ending at 4.
t = linspace(0,4);
%%% Define three row vectors h1, h2 and h3, with entries corresponding
%%% to h1(t), h2(t) and h3(t) evaluated at each time in t.
h = c1*v1*exp(lambda1*t) + c2*v2*exp(lambda2*t) + c3*v3*exp(lambda3*t);
h1 = h(1,:);
h2 = h(2,:);
h3 = h(3,:);
%%% Now use plot to plot the three vectors against time on the same figure
plot(t, h1);
hold on
plot(t, h2);
plot(t, h3);
set(gca,'fontsize',18)
xlabel('Time')
ylabel('Volume')
title('Time series')

```

Run Script ()

Assessment: Correct

Submit ()

- h1
- h2
- h3

plot used

Output

3. 3x3 Example

Hide Discussion

Topic: Unit 3: Solving systems of first order ODEs using matrix methods / 3. 3x3 Example

Add a Post

Show all posts ▼ by recent activity ▼

There are no posts in this topic yet.

✖

Learn About Verified Certificates