***Name-Tushar Chaudhary***

***Roll No.-2100290310152***

***Section -C***

***Subject- CS Lab***

***EXPERIMENT NO:-1***

***OBJECTIVE***: Introduction to MATLAB Control System Toolbox.

***SOFTWARE REQUIRED:***- MATLAB 2022, Control System Tool Box

***THEORY***:

***1.1 Introduction to Control System***

Control System Toolbox is a package for Matlab consisting of tools specifically developed for

control applications. The package offers data structures to describe common system

representations such as state space models and transfer functions, as well as tools for analysis

and design of control systems. There are also tools for simulation of systems.

Here we will get to know the basic commands of Control System Toolbox.

After completion of this experiment you should be able to understand and use Control Systems

Toolbox to create and analyze linear systems. Extensive use of the Matlab help command is

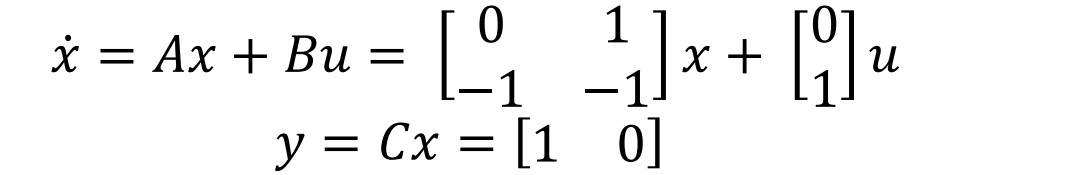
recommended. Student is recommended to create a script file (e.g. myscript.m) in which

commands can be written and by running a script file instead of typing the commands directly at

the Matlab prompt, it is quite easier to correct mistakes, and also, the work will be saved for later

use.

The system we will use, of the form



To enter a matrix in MATLAB, for example matrix A, do

𝑦 = 𝐶𝑥 = [1 0]

To enter a matrix in MATLAB, for example matrix A, do

𝐴 = [0 1; −1 −1]

***Creation and Conversion of systems***

Control System Toolbox supports several system representations of linear time invariant

systems. In this exercise, we will use two of the most common representations; state space

models and transfer functions.

Define the system matrices A, B, C and D given above. (What is the value of D in the model?)

Create a state space description of the system using ss, and name it sys\_ss. Find out how to use

ss by using the help function (help ss). At this stage, you should have obtained a state space

description of the system.

Let us now create an equivalent transfer function model of the system above. This could, as you

know, be done by using the formula 𝐺(𝑠) = 𝐶(𝑠𝐼 − 𝐴)

−1 𝐵 + 𝐷. However, Matlab may also be

used for the task. Use the command tf to convert the state space model to a transfer function and

name it sys\_tf. Notice that tf may be used for creation of transfer functions as well as conversion.

What is the syntax in the two cases respectively?

***Stability Analysis***

Stability of a linear system is determined by the location of its poles in the complex plane. (What

is the condition for stability?) Use the commands ssdata and tfdata to extract the necessary data

from the models, and eig and roots to determine stability of the system. Verify that the roots of the characteristic polynomial of the transfer function are the same as the eigenvalues of the

system matrix.

What are the eigenvalues / poles? Is the system stable? You could also use the command pole, or

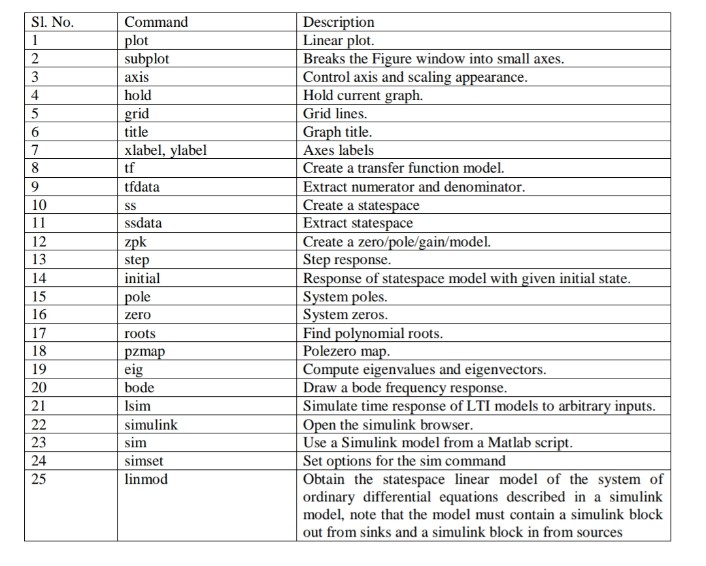
for a graphical view, pzmap.

Time domain analysis

Use the command step to plot the step response of the system. Relate the characteristics of the

step response to the location of the poles. If there is time, use initial and lsim to study the system

response.

**Some useful Matlab commands:**

***1.2 Introduction to Simulink***

Simulink is a simulation program based upon Matlab. There are several ways to define a model.

One can work graphically and connect blockdiagrams with predefined blocks. Alternatively one

can give the mathematical description in forms of differential equations in an mfile (the format

for programs written in the Matlab programming language). Matlab/Simulink supports both these

representations as well as combinations. Furthermore one can use descriptions that include a

hierarchy of connected subsystems.

To understand how models are described and simulated using block diagrams, it is best to run

small examples on a computer. The rest of Section shows some examples.

*How to Start Simulink*

Login through “https://in.mathworks.com/products/matlab-online.html”

Then give the command simulink in Matlab command prompt. This gives a window with blocks

as in Figure 1.

****

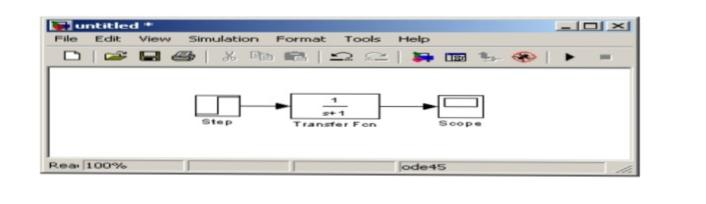
***A Simple System***

Click on File in the Simulinkwindow and choose New->Model. Click on the block Continous

and move a Transfer Fcn to the new window called “Untitled”. Do the same with Source->Step

Fcn and Sinks->Scope. Draw arrows (left mouse button) and connect the ports on the block. You

**should now have a block diagram as in Figure 2.**

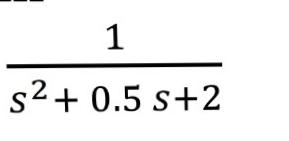
****

Choose Simulation->Parameters in the window called “Untitled”. Set Stop time to 5. Open the

window Scope by double clicking on it. Put Horizontal Range to 6. Start a simulation by

Simulation->Start (or by pressing Ctrl-t in the window called “Untitled’).

***How to Change a System***

To change the system to

you doubleclick on the block Transfer Fcn and change Denominator to [1 0.5 2]. Simulate the

new system (Simulation->Start or Ctrl-t). Change parameters in the Simulation menu and the

scales in the block Scope until you are satisfied.How to Change an Input Signal To change the

input signal, start with removing the block Step Fcn by clicking on it and delete it by using Edit-

>Cut (or pressing Ctrlx). Replace it by a Sources->Signal Gen block. Doubleclick on Signal Gen

and select signal, amplitude and frequency. Also change Simulation-> Start->Stop Time to

99999 and press Simulation->Start. This gives an “infinite” simulation that can be stopped by

pressing Simulation->Stop (or Ctrlt).

Can the amplitude of the input signal be changed during simulation? Also try to change the block

Transfer Fcn during simulation.

***How to Use Matlab Variables in Blocks***

Note that variables defined in the Matlab environment can be used in Simulink. Define

numerator and denominator by writing the following in the Matlab window.

num=[1 1]

den=[1 2 3 4]

Change Transfer Fcn->Numerator to num and Transfer Fcn->Denominator to den.

***How to Save Results to Matlab variables***

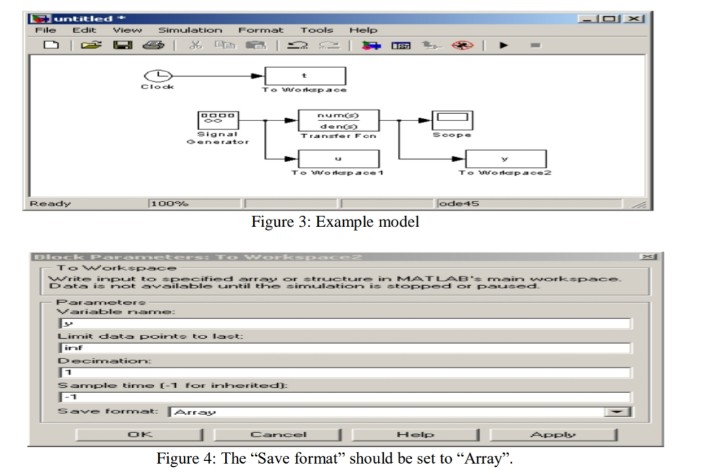
To save input and output move two copies of the block Sinks->To Workspace. Make sure that

the “save format” option is set to “Array”, se figure 2. Connect these with the input and output to

the block Transfer Fcn. Also get a Source->Clock and connect it to a Sinks->To Workspace.

Change the variable names to u,y,t respectively. The window should look something like the

figure 3.



***How to Use Simulation Results in Matlab Calculations***

Let the input signal be a sinusoidal with frequency 0.1 rad/s and amplitude 1. Do a simulation.

That is long enough for the output to become stationary. Compute the maximum value of y when

The system has settled.

N=length(y)

Max(y(n/2:n))

***Using Simulink Models in Matlab Scripts***

Often, it is convenient to work with Matlab scripts (mfiles), in order to save a sequence of

Commands. It is possible to use Simulink models from within a Matlab script, using the

Command sim. By using the command simset options for the sim command may be specified.

Use the model from the previous example. Save the model, and name it “mymodel.mdl”. Create

A Matlab script named “mysim.m”, and enter the following commands:

Tfinal = 300;

Options = simset(’reltol’,1e-5,’refine’,10,’solver’,’ode45’);

Sim(’mymodel’,tfinal,options);

%plot results

Figure(1)

Clf

Subplot(211)

Plot(t,u);

Ylabel(’u’)

Subplot(212)

Plot(t,y)

Ylabel(’y’)

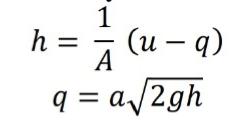
When you run the script, you should see a plot showing the input and the output of the transfer

Function. Use the help command to learn more about how to use the simset and sim commands.

***How to Save Systems***

Use File-Save As or File->Save.

**A Flow System**

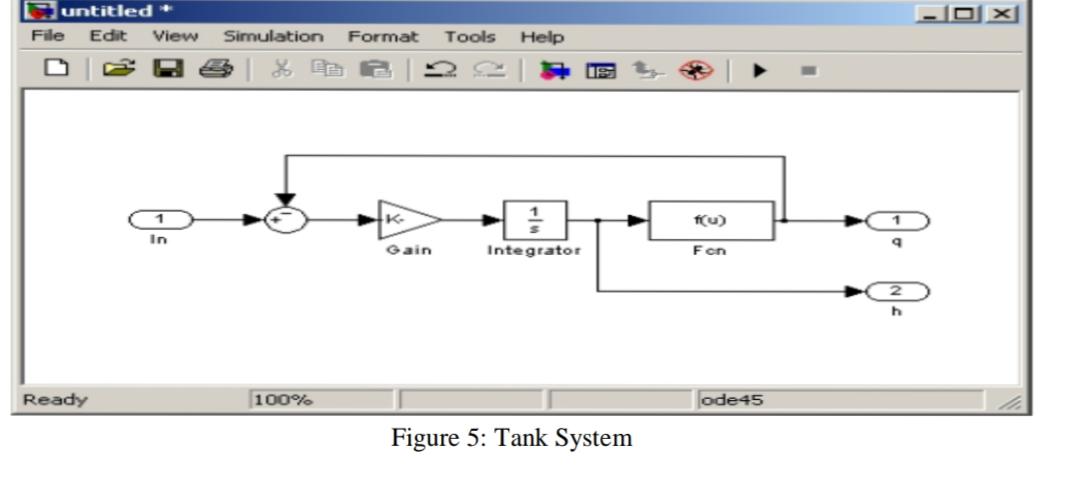


Consider a simple tank as in the basic control

This can be implemented in Simulink as in Figure 5. The function f (u) has the value

A\*sqrt(2\*g)\*sqrt(u[1]). The block Sum has been given two inputs with different signs by

Assigning the string “-+|” to Sum->List of Signs. The Input and Output blocks can be found

Under Sources and Sinks respectively.

These blocks tell Simulink what should be considered inputs and outputs to this (sub) system.

The block titles can be changed by clicking on them. Mark the entire system by holding the left

mouse bottom pressed and drawing a square around it. Then choose Edit->Create Subsystem.

The result is that the system is represented by one block. Use Edit->Copy to create the following

doubletank system (Figure 6). Use the command linmod to find a linearized model of the double

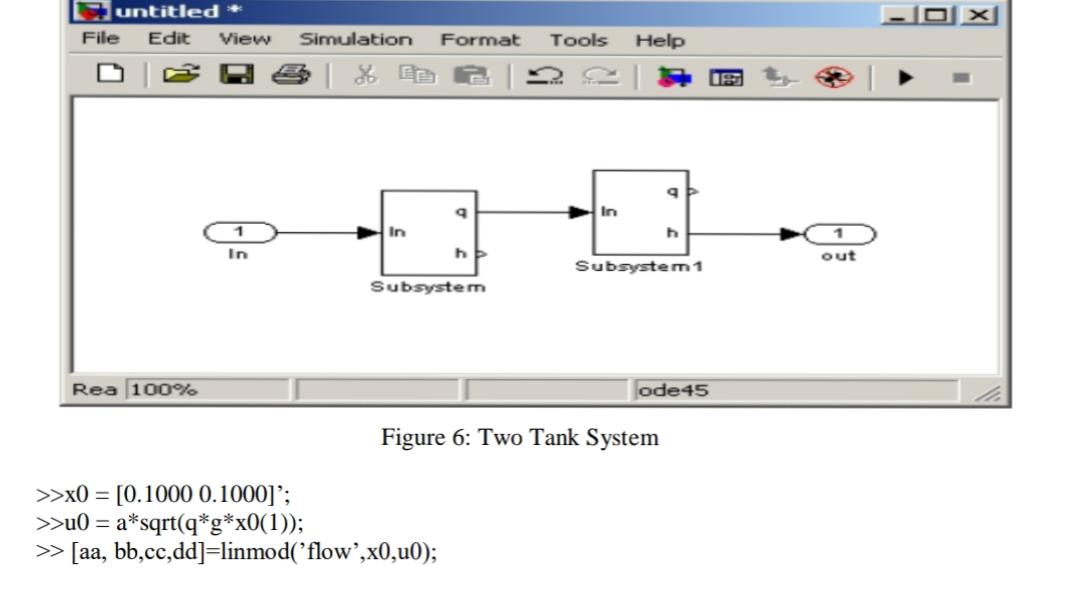
tank around ℎ1

0 = ℎ2

0 = 0.1 . Use the parameters A1 = A2 = 2.7 × 10−3, a1 = a2 = 7.0 ×10−6

, g =

9.8. Notice also in figure the simulink block in from sources and the simulink block out from

sinks, which are necessary for the linmod command.