Solving the Motion Equations in the Mechanical System of Multiple Input Multiple Output by Transfer Function Blocks using MATLAB/Simulink

Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman[1],Musa Adam Abdullah[2], Sulima Ahmed Mohammed Zubir[3],Subhi Abdalazim Aljily Osman[4], Mohammed Yasin Bashir Mohager[5] and Balgis Ahmed Mohammed Albur[6]

[1]Department of Mathematics, Faculty of Education, Omdurman Islamic University, Omdurman, Sudan.
[2]Department of Mathematics, College of Computer Science and Information Technology, University of the Holy Quran and Tassel of Science, Sudan.
[3]Department of Mathematics, College of Science and Arts, Qassim University, Ar Rass, Saudi Arabia
[4]Department of Mathematics, Faculty of Computer Science and Information Technology, University of ALBUTANA, Sudan.
[5]Department of Mathematics, College of Education, Blue Nile University, Eldamazin, Sudan
[6]Department of Mathematics, Faculty of Education, University of Gezira, Elhasahisa, Sudan
***Corresponding Author: -**Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman

Abstract

A mechanical system is a system composed of the mechanical elements mass, damper and spring. The transfer function of a linear, time-invariant, differential equation system is defined as the ratio of the Laplace transform of the output (response function) to the Laplace transform of the input (driving function) under the assumption that all initial conditions are zero. The aims of this paper is to describe and investigate the solution of the motion equations in the mechanical system of multiple input multiple output by using Transfer Function blocks in MATLAB/Simulink. It deals with the mechanical system with two-mass systems .We followed two applied mathematical methods to obtain the free response for$x_1(t)$. Firstly we found the free response for $x_1(t)$ and the second we used the new mathematical technique by multiply the numerator of $X_1(s)$ by s .We found that the solution of MATLB and MATLAB/Simulink gives the response plot but not the closed-form solution also we found the free response. Finally we can say that the first method gives the response plot but not the closed-form solution and the second method gives the closed-form solution for the free response.

Keywords: Mechanical System, Transfer Function blocks, State Space,Model MATLAB/Simulink.

DOI Number: 10.14704/nq.2022.20.9.NQ440130 Neuro Quantology 2022; 20(9):1159-1165

1. Introduction

A mechanical system is a Systems that possess significant mass, inertia and spring and energy dissipation (damper) components driven by forces, torques, specified displacements are considered to be mechanical systems. Multiple-input, multiple-output systems have more than one component at either the input or the output; multiple degree-of-freedom systems, which have been studied in previous chapters, are MIMO systems because they possess more than one degree-of-freedom and therefore have multiple outputs. Figure No. (1)Illustrates a generic dynamic system defined by $p$differential equations, denoted by D_1toD_p, and therefore by p outputs (the differential equations solution components$x_1, x_2, \dots, x_p$); also $m$inputs,y_1, y_2, …, y_m, are acting on the system [8]. The Laplace transform, as the name implies, is a transformation of functions between two domains. The independent variables in the two domains are commonly denoted "t" and "s" the domains are referred to as the time domain (or t-domain) and s-domain, respectively. The state-space representation offers several advantages over the input–output transfer function method of describing the dynamics of a linear system [7].There are several computer packages for finding solutions of a mechanical system, such as Maple, Mathematica, Maxima, MATLAB, etc.

These systems provide both symbolic and numeric approaches to finding solutions. They often require a bit of coding. However, there are graphical environments for solving problems, including differential equations [1]. One such environment is Simulink, which is closely connected to MATLAB [6].The MATLAB® and Simulink® environments are integrated into one entity, and thus we can analyze, simulate, and revise our models in either environment at any point. We invoke Simulink from within MATLAB [6].

2. Transfer Function of multiple input multiple output system

The definition of a transfer function is easily extended to a system with multiple inputs and outputs. A system of this type is often referred to as a multivariable system [1].The Laplace transforms of the input components,$Y_1(s)$, $Y_2(s), \dots,$ $Y_m(s)$, can be collected into an input vector$\{X(s)\}$, whereas the Laplace transforms of the output components are gathered, similarly, into another vector, the output vector $\{X(s)\}$ [8].

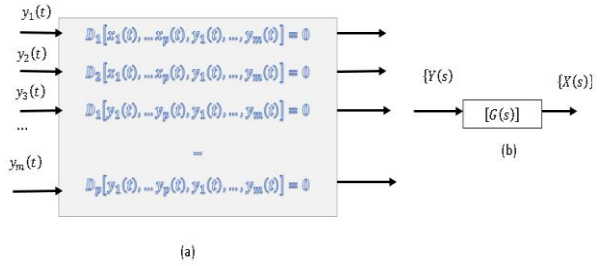

Figure No. (1):MIMO System with Input and Output Components Connected in the (a) Time Domain by a Differential Equations System; (b) Laplace Domain by the Transfer Function Matrix.

Transfer function matrix definition: For a MIMO system, the transfer function matrix $[G(s)]$ connects the Laplace transform of the output vector,$[X(s)]$, to the Laplace transform of the input vector,$[Y(s)]$:

$$[\mathrm{X(s)}] = [\mathrm{G(s)}][\mathrm{Y(s)}] \quad (1)$$

3. First-Order MIMO Systems are Described by the Differential Equation:

$$a_1\dot{y}(t) + a_0 y(t) = u(t) \quad (2)$$

The number of inputs, m, is normally smaller than the number of outputs (which is assumed equal to the number of differential equations of the mathematical model), p; however, by adding p – m zeroes to the input, the two vectors have the same dimension. Application of the Laplace transform to equation. (2).

The transfer function matrix corresponding to zero initial conditions for the first-order system is obtained from equation (3) as

$$(a_1 s + a_0)Y(s) = U(s) \quad (3)$$

$$G(s) = (s[a_1] + [a_0])^{-1} \quad (4)$$

4. Second-Order Systems:

The vector-matrix differential equation of second-order MIMO systems is

$$a_2\ddot{y}(t) + a_1\dot{y}(t) + a_0 y(t) = u(t) \quad (5)$$

Where, again, the input and output vectors are assumed to have the same dimension. When the initial conditions are zero, the Laplace transform that is applied to equation (5) yields

$$(a_2 s^2 + a_1 s + a_0)Y(s) = U(s) \quad (6)$$

Taking into account that the regular transfer function matrix $[G(s)]$ is the one connecting $\{Y(s)\}$ to $\{U(s)\}$ [2],

$$[G(s)] = (s^2[a_2] + s[a_1] + [a_0])^{-1} \quad (7)$$

In general, if a linear system has$p$ inputs and q outputs, the transfer function betweenthejth input and the ith output is defined as

$$G_{ij}(s) = \frac{Y_i(s)}{R_j(s)} \quad (8)$$

With$R_k(s) = 0, k = 1,2,\dots,p, k \neq j$.
When all the p inputs are in action, the ith output transform is written

$$Y_i(s) = G_{i1}(s)R_1(s) + G_{i2}(s)R_2(s) + \cdots + G_{ip}(s)R_p(s) \quad (9)$$

We write the simultaneous equation in matrix form as [10].

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \\ \vdots \\ Y_q(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1p}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2p}(s) \\ \vdots & \vdots & \cdots & \vdots \\ G_{q1}(s) & G_{q1}(s) & \cdots & G_{qp}(s) \end{bmatrix} \begin{bmatrix} R_1(s) \\ R_2(s) \\ \vdots \\ R_p(s) \end{bmatrix} \quad (10)$$

or simply

$$Y(s) = G(s)R(s) (11)$$

5. Solution of theTransfer Function of Multiple Input Multiple Output withe Cramer's Rule:

Now we present a transfer function of multiple input multiple output mechanical system with two-mass system as solved in MATLAB. The model consists the coordinates x_1 and x_2 as the displacements of the masses from their equilibrium positions. In equilibrium, the static forces in the springs cancel the weights of the masses. Note that the dampers have no effect in equilibrium and thus do not determine the location of the equilibrium position [9].

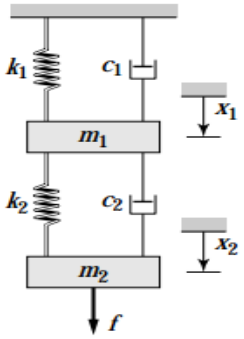

Figure No. (2):A Two-Mass Systems.

The equations of motion are given by equations (12) and (13) of A two-mass system. With the given parameter values, they become

$$m_1\ddot{x}_1 + (c_1 + c_2)\dot{x}_1 + (k_1 + k_2)x_1 - c_2\dot{x}_2 - k_2x_2 = 0 \quad (12)$$

$$m_2\ddot{x}_2 + c_2\dot{x}_2 + k_2x_2 - c_2\dot{x}_1 - k_2x_1 = f(t) \quad (13)$$

Taking the Laplace transform of the above equations, assuming zero initial condition, gives

$$m_1s^2X_1(s) + (c_1 + c_2)sX_1(s) + (k_1 + k_2)X_1(s) - c_2sX_2(s) - k_2X_2(s) = 0$$

$$m_2s^2X_2(s) + c_2sX_2(s) + k_2X_2(s) - c_2sX_1(s) - k_2X_1(s) = F(s)$$

Collecting terms results in

$$(m_1s^2 + (c_1 + c_2)s + (k_1 + k_2))X_1(s) - (c_2s + k_2)X_2(s) = 0 \quad (14)$$

$$(m_2s^2 + c_2s + k_2)X_2(s) - (c_2s + k_2)X_1(s) = F(s) (15)$$

We put equations (14) and (15) in a linear transfer matrix systems

$$\begin{bmatrix} (m_1s^2 + (c_1 + c_2)s + (k_1 + k_2)) & -(c_2s + k_2) \\ -(c_2s + k_2) & (m_2s^2 + c_2s + k_2) \end{bmatrix} \begin{bmatrix} X_1(s) \\ X_2(s) \end{bmatrix} \begin{bmatrix} 0 \\ F(s) \end{bmatrix} \quad (16)$$

The linear transfer matrix systems can solve by the most general method for our purposes is to solve them using determinants (this is known as Cramer's method) [9].

The determinant of the left-hand side of the linear system is

$$D(s) = \begin{bmatrix} (m_1s^2 + (c_1 + c_2)s + (k_1 + k_2)) & -(c_2s + k_2) \\ -(c_2s + k_2) & (m_2s^2 + c_2s + k_2) \end{bmatrix}$$

$$= (m_1s^2 + (c_1 + c_2)s + (k_1 + k_2))(m_2s^2 + c_2s + k_2) - (c_2s + k_2)^2 \quad (17)$$

The solutions for $X_1(s)$and $X_2(s)$can be expressed as

$$X_1(s) = \frac{D_1(s)}{D(s)}X_2(s) = \frac{D_2(s)}{D(s)} (18)$$

Where

$$D_1(s) = \begin{bmatrix} 0 & -(c_2s + k_2) \\ F(s) & (m_2s^2 + c_2s + k_2) \end{bmatrix} = (c_2s + k_2)F(s) \quad (19)$$

$$D_2(s) = \begin{bmatrix} (m_1s^2 + (c_1 + c_2)s + (k_1 + k_2)) & 0 \\ -(c_2s + k_2) & F(s) \end{bmatrix}$$

$$= (m_1s^2 + (c_1 + c_2)s + (k_1 + k_2))F(s) (20)$$

The transfer functions for the input $f(t)$are found in $D_1(s)$and$D_2(s)$ [9]:

$$\frac{X_1(s)}{F(s)} = \frac{D_1(s)}{D(s)} = \frac{(c_2s + k_2)}{D(s)} (21)$$

$$\frac{X_2(s)}{F(s)} = \frac{D_2(s)}{D(s)}$$

$$= \frac{(m_1s^2 + (c_1 + c_2)s + (k_1 + k_2))}{D(s)} (22)$$

The unit-step response for x_1 *and* $x_2$are plotted with the following MATLAB program [9].

MATLAB Program 1

```
m1=1; m2=1;c1=2; c2=3; k1=1; k2=4;
D=conv([m1,c1+c2,k1+k2], [m2,c2,k2])-conv([0,c2,k2], [0,c2,k2]);
x1=tf([0, c2, k2],D), x2=tf([m1,c1+c2,k1+k2],D)
step(x1,x2)
x1 =
 3 s + 4
 ----------------------------
s^4 + 8 s^3 + 15 s^2 + 11 s + 4

Continuous-time transfer function.
x2 =
 s^2 + 5 s + 5
 ----------------------------
s^4 + 8 s^3 + 15 s^2 + 11 s + 4
Continuous-time transfer function.
```

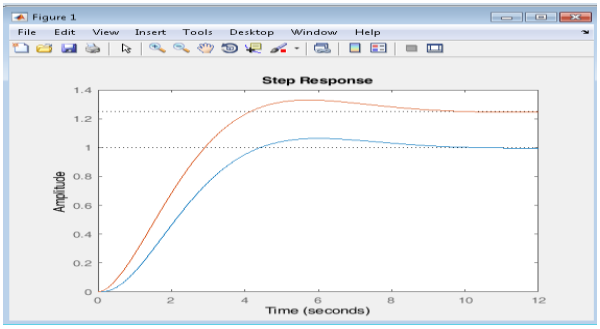

Figure No. (3): Simulation Results in Simulink of the Unit-Step Response for x_1 *and* x_2

Suppose that$m_1 = m_2 = 1, c_1 = 2, c_2 = 3, k_1 = 1, and\ k_2 = 4$. And substitute these in equations (12) and (13) they become

$$\ddot{x}_1 + 5\dot{x}_1 + 5x_1 - 3\dot{x}_2 - 4x_2 = 0 \qquad (23)$$

$$\ddot{x}_2 + 3\dot{x}_2 + 4x_2 - 3\dot{x}_1 - 4x_1 = f(t) \qquad (24)$$

We use two methods with MATLAB to obtain the free responsefor$x_1(t)$, for the initial conditions$x_1(0) = 3, \dot{x}_1(0) = 2, x_2(0) = 1, and\ \dot{x}_2(0) = 4$.

We need to find the free response, we must now keep the initial conditions in the analysis. So, transforming the equations with the given initial conditions gives

$$s^2X_1(s) - sx_1(0) - \dot{x}_1(0) + 5[sX_1(s) - x_1(0)] + 5X_1(s) - 3[sX_2(s) - x_2(0)] - 4X_2(s) = 0 \qquad (25)$$

$$s^2X_2(s) - sx_2(0) - \dot{x}_2(0) + 3[sX_2(s) - x_2(0)] + 4X_2(s) - 3[sX_1(s) - x_1(0)] - 4X_1(s) = F(s) \qquad (26)$$

Collecting terms results in

$$(s^2 + 5s + 5)X_1(s) - (3s + 4)X_2(s) = sx_1(0) + \dot{x}_1(0) + 5x_1(0) - 3x_2(0) = I_1 \qquad (27)$$

$$-(3s + 4)X_1(s) + (s^2 + 3s + 4)X_2(s) = sx_2(0) + \dot{x}_2(0) + 3x_2(0) - 3x_1(0) = I_2(s) + F(s) \qquad (28)$$

Where the terms due to the initial conditions are denoted by $I_1(s)$and$I_2(s)$.

$$I_1(s) = sx_1(0) + \dot{x}_1(0) + 5x_1(0) - 3x_2(0) \qquad (29)$$

$$I_2(s) = sx_2(0) + \dot{x}_2(0) + 3x_2(0) - 3x_1(0) \qquad (30)$$

Equations (27) and (28) are two algebraic equations in two unknowns. Their solution can be obtained in several ways, but the most general method for our purposes is to solve them using determinants (this is known as Cramer's method).

The determinant of the left-hand side of the equations (27) and (28) is

$$D(s) = \begin{bmatrix} (s^2 + 5s + 5) & -(3s + 4) \\ -(3s + 4) & (s^2 + 3s + 4) \end{bmatrix}$$

$$= (s^2 + 5s + 5)(s^2 + 3s + 4) - (3s + 4)^2 \qquad (31)$$

The solutions for $X_1(s)$and $X_2(s)$can be expressed as

$$X_1(s) = \frac{D_1(s)}{D(s)} X_2(s) = \frac{D_2(s)}{D(s)} \qquad (32)$$

Where

$$D_1(s) = \begin{bmatrix} I_1(s) & -(3s + 4) \\ I_2(s) + F(s) & (s^2 + 3s + 4) \end{bmatrix}$$

$$= (s^2 + 3s + 4)I_1(s) + (3s + 4)I_2(s) + (3s + 4)F(s) \qquad (33)$$

$$D_2(s) = \begin{bmatrix} (s^2 + 5s + 5) & I_1(s) \\ -(3s + 4) & +F(s) \end{bmatrix}$$

$$= (s^2 + 5s + 5)I_2(s) + (s^2 + 5s + 5) F(s) + (3s + 4)I_1(s) \qquad (34)$$

Using the new technique multiply the numerator of $X_1(s)$ by$s$. The following MATLAB program performs the calculations. Note that the multiplication by s is accomplished by appending a 0 to the polynomials used to compute D_1 (with$F(s) = 0$).

MATLAB Program 2

```
% Specify the initial conditions.
x10 = 3; x1d0 = 2; x20 = 1; x2d0 = 4;
% Form the initial-condition arrays.
I1 = [x10,x1d0+5*x10-3*x20]; I2 = [x20,x2d0+3*x20-3*x10];
D1 = conv([1,3,4,0],I1)+conv([0,3,4,0],I2);
D = conv([1,5,5],[1,3,4])-conv([0,3,4],[0,3,4]);
sys = tf(D1,D);
step(sys)
```

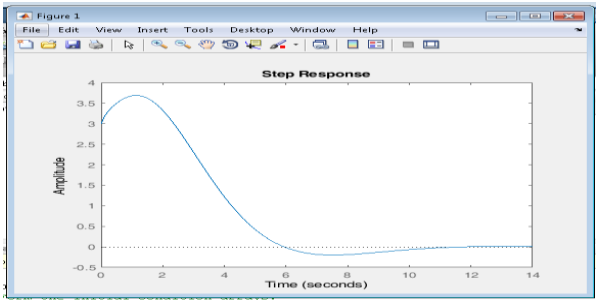

Figure No. (4): Graph Results of the new technique multiply the numerator of $X_1(s)$ by*s*.

The first method gives the response plot but not the closed-form solution. The following method gives the closed-form solution for the free response.

The characteristic roots can be found with the following MATLAB file.

MATLAB Program 3

```
% Find the roots.
D = conv([1,5,5],[1,3,4])-conv([0,3,4],[0,3,4]);
R = roots(D)
R =
-5.6773 + 0.0000i
-1.3775 + 0.0000i
-0.4726 + 0.5368i
-0.4726 - 0.5368i
```

Note that if all the characteristic roots are distinct, the partial fraction expansion gives the following solution form, even if the roots are complex.

$$x_1(t) = \sum_{i=1}^{4} r_i e^{p_i t} \quad (35)$$

Where the$r_i$are the residues of the partial-fraction expansion and the p_i are the poles (the roots). So if the roots are all distinct, we can use the following MATLAB file to perform the rest of the algebra.

MATLAB Program 4

```
% Specify the initial conditions.
x10 = 3; x1d0 = 2; x20 = 1; x2d0 = 4;
I1 = [x10,x1d0+5*x10-3*x20];
I2 = [x20,x2d0+3*x20-3*x10];
A1=[1,3,4,0];A2=[0,3,4,0];
D1 = conv(A1,I1)+conv(A2,I2);
B1=[1,5,5]; B2=[1,3,4]; B3=[0,3,4]; B4=[0,3,4];
D = conv(B1,B2)-conv(B3,B4);
sys = tf(D1,D);
step(sys)
xlabel('t'),ylabel('x 1')
```

Although in the expression for x_1 the imaginary parts should cancel, giving a real result, we use the real function because numerical round-off errors can produce a small imaginary part. The plot is shown inFigure No. (5).we can use a similar approach to plot$x_2(t)$.

If a closed-form expression for the free response is required, you can use the roots (the poles) in the array p and the expansion coefficients (the residues) in the array r. If the roots are complex, you can use the Euler identities $e^{jx} = cos\, x \pm j\, sin\, x$ with the expansion coefficients to obtain the solution.Here the residues corresponding in order to the four roots given already are −0.3554, 3.7527, and −0.1987 ± 4.6238 j [9].

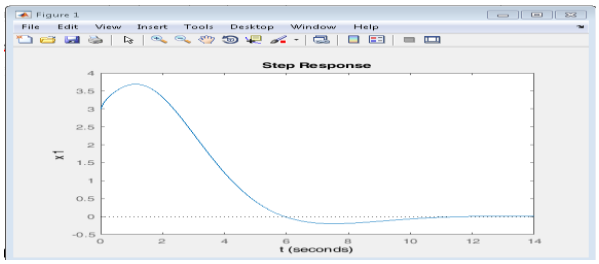

Figure No. (5): Graph Results of the Real Function Because Numerical Round-off Errors can produce a Small Imaginary Part.

6. Transfer Function of Multiple Input Multiple Output in MATLAB Simulation:

A MIMO system is defined by the transfer function matrix shown in equation (20), (21)

$$\frac{X_1(s)}{F(s)} = \frac{D_1(s)}{D(s)} = \frac{(c_2 s + k_2)}{D(s)} \quad (20)$$

$$\frac{X_2(s)}{F(s)} = \frac{D_2(s)}{D(s)}$$

$$= \frac{(m_1 s^2 + (c_1 + c_2)s + (k_1 + k_2))}{D(s)} (21)$$

We use Simulink® to determine and plot the time response of this system under the action of the input$f(t)$. The number of output components is

two, as indicated by the number of rows of the transfer function, which connects the input to the output Laplace-transformed vectors according to $\{X(s)\} = [G(s)]\{F(s)\}$. Simulink® has the capability of modeling MIMO systems and transfer function matrices by means of the same Transfer Function command. Specifically, the following specifications are needed in the Function Block Parameters:

Numerator coefficient:
[0,3,4;1,5,5]

Denominatorcoefficient:
[1, 8, 15, 11, 4]

Notice that while the numerator contains the numerator coefficients of the transfer function (and can be a matrix), the denominator can only be a vector collecting the coefficients of the denominator polynomial [4], [9].

Because the transfer function block accepts formally one input and generates one output, the Merge block (taken from the signal Routing library and shown in the Simulink® diagram of Figure No. (6) has to be used to model the three input components.

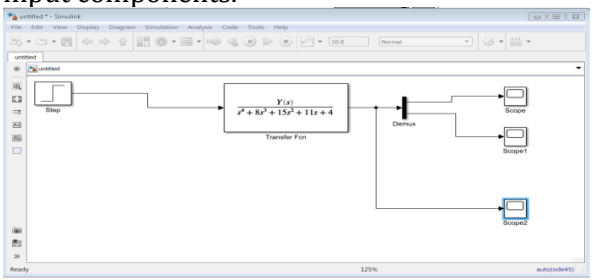

Figure No. (6): The Block Diagram in Simulink Representation of the Transfer Function Equations.

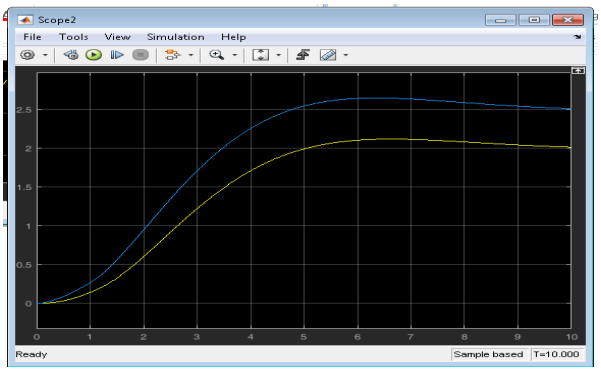

Figure No. (7): Simulation Results in Simulinkof the Transfer Function Equations.

7. Transfer Function to State Space of Multiple Input Multiple Output in MATLAB Simulation:

The obtained transfer function model of the mechanical system can be in the Matlab easily transformed to an equivalent state-spacerepresentationFigure No. (8)by tf2ssfunction [3], [5]. The syntax of this command in Matlab is:

>> [A, B, C, D] =tf2ss ([0 3 4; 1 5 5], [1 8 15 11 4])

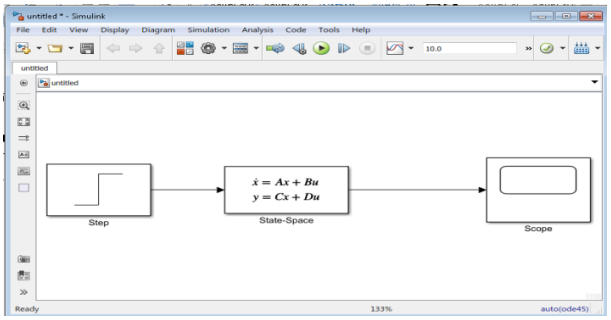

Figure No. (8): The Block Diagram in Simulink Representation of the State Space Equations.

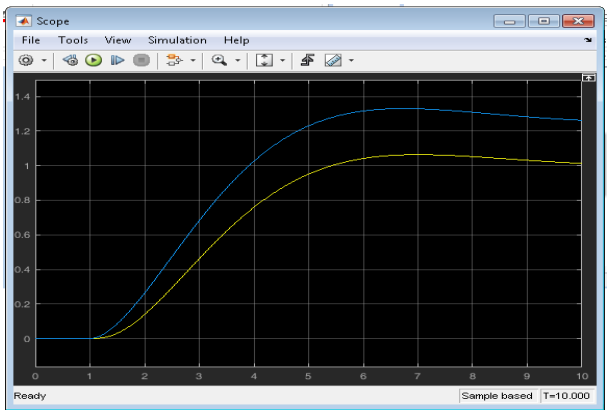

Figure No. (9): Simulation Results in Simulink of the Transfer Function model of the mechanical system

Result:

We used determinants (this is known as Cramer's method) tosolvethe Transfer Function of multiple input multiple output and weused two methods with MATLAB to obtain the free response for$x_1(t)$, for the initial conditions$x_1(0) = 3, \dot{x}_1(0) = 2, x_2(0) = 1, and\ \dot{x}_2(0) = 4$. The first we find the free response for $x_1(t)$ and the second we used the new technique by multiply the numerator of $X_1(s)$ by s. We find the free response.The first method gives the response plot but not the closed-form solution. The secondly method gives the closed-form solution for the free response.

Conclusion:

We described the solution of the equations of motion of the mechanical system of multiple input multiple output by using Transfer Function blocks by MATLAB/Simulink. It deals with the mechanical system with two-mass systems. Dynamicsystems can be modeled (or represented) using transfer functions or a set of state-space equations. Transfer functions are used primarily for single-input, single-output systems. State space equations can be used for multiple-input, multiple-output systems, are very versatile, and can be used to model very complex systems.

References:

Ahmed, Musa Adam Abdallah. Transfer Functions and Mathematical Modeling of Dynamic Systems with MATLAB. Diss. Sudan University of Science and Technology, 2018.

De Andrade, Flávia, Miguel Castilla, and Benedito Donizeti Bonatto. *Basic Tutorial on Simulation of Microgrids Control Using MATLAB® & Simulink® Software*. Springer Nature, 2020.

Gogoi, Parismita. "MATLA and SIMULINK." *National Level Workshop On MATLAB and Selected Applications*.

Golnaraghi, Farid, and Benjamin Kuo. *Automatic control systems*. McGraw-Hill Education, 2017.

Herman, Russell. "Solving differential equations using Simulink." *Published by RL Herman* (2016): 259-268.

Karris, Steven T. *Introduction to Simulink with engineering applications*. Orchard Publications, 2006.

Klee, Harold, and Randal Allen. *Simulation of dynamic systems with MATLAB® and Simulink®*. Crc Press, 2018.

Lobontiu, Nicolae. *System dynamics for engineering students: Concepts and applications*. Academic Press, 2017.

Palm, William John. *System dynamics*. Vol. 2. New York: McGraw-Hill, 2010.

Richard C.. Dorf, and Robert H. Bishop. *Modern control systems*. Pearson Prentice Hall, 2008.