Laboratory Manual

Of

NUMERICAL METHODS AND PROGRAMMING ANALYTICAL

LAB COURSE

(EE256)

DEPARTMENT OF ELECTRICAL ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ANDHRA PRADESH

(AY: 2021-2022)

DEPARTMENT OF ELECTRICAL ENGINEERING

NUMERICAL METHODS AND PROGRAMMING ANALYTICAL LAB

Student Details	
Year / Semester	II B. Tech (EEE) – II Semester
Subject	NUMERICAL METHODS AND PROGRAMMING ANALYTICAL LAB
Regulation	
Subject Code	
Roll Number	
Name	

Student Performance Evaluation

Week / Exercise Number	**Date**	**Marks**				**Signature of Faculty**	**Remarks**
		Lab (5)	**Record (5)**	**Viva (5)**	**Total (15)**		
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							

Total Marks Awarded: _____ (In Words: _____)

EE256- NUMERICAL METHODS AND PROGRAMMING ANALYTICAL LAB

IMPORTANT INSTRUCTIONS FOR STUDENTS

- Leave your foot-wear outside the Lab.
- Every student must carry his/her ID card.
- Keep your mobile(s) in SILENT.VIBRATION mode.
- Bags, edible items are not allowed inside the laboratory.
- Please maintain silence.
- Installation of any software without a written permission from the lab-in-charge is strictly prohibited.
- Access of any illegal proxy websites will result in a strict punishment.
- While watching lecture videos, headphones must be used.
- Watching entertainment videos is strictly prohibited.
- Don’t reset passwords for the computers.
- Every student must shutdown the computer and arranges the chair properly before leaving the lab.
- Every student must sign in the utilization register of the lab.

EE256- NUMERICAL METHODS AND PROGRAMMING ANALYTICAL LAB

(B.Tech.-EED-II year-II Semester)

Externals: 60 Marks **L-T-P-C**
Internals: 40 Marks **0-0-3-2**

Course Objectives: To expose the students to the operation of MATLAB software, coding and give them software skills.

Course Outcomes: At the end of the course the student will be able to:

CO1	Select range of apparatus based on the ratings of DC Machines and Transformers.
CO2	Determine equivalent circuit parameters of transformers
CO3	Evaluate the efficiency of the machine by analyzing test results
CO3	Evaluate the efficiency of the machine by analyzing test results

List of Experiments for the Numerical methods and programming analytical Lab

S.No.	Name of the Experiment	Page No.
1	Study of Introduction to MATLAB	
2	Study of basic matrix operations	
3	To solve linear equation	
4	Solution of linear equations and different case studies.	
5	Determination of Eigen values and Eigen vectors of a square matrix.	
6	Solution of Difference Equations.	
7	Solution of Difference Equations using Euler Method.	
8	Solution of differential equation using 4^{th} order Runge- Kutta method.	
9	Determination of roots of a polynomial.	
10	Determination of polynomial using method of Least Square Curve Fitting.	
11	Determination of time response of an R-L-C circuit.	
12	Simultaneous Equations: Gauss Seidel Method	
13	Simultaneous Equations: Gauss Elimination Method	

EXP. NO.: 01
DATE :

STUDY OF INTRODUCTION TO MATLAB

AIM: To study about the introduction of MATLAB.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB software	Version-2018	1

INTRODUCTION:

The name MATLAB stands for Matrix Laboratory. The basic building block of MATLAB is the matrix. It is not confined to the solution of Matrix related problems. With its inbuilt functions, it is an excellent tool for linear algebraic computations, data analysis; signal processing, optimization, numerical solutions of ordinary differential equations (ODE), quadrature, 2D & 3D, graphics and many other types of scientific computation. Therefore, we can say:

MATLAB is a software package in high performance language for technical computing. It integrates computation, visualization and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

1. Math and computation
2. Algorithm and development
3. Data acquisition modeling
4. Simulation and prototyping
5. Data analysis, exploration, and visualization
6. Scientific and engineering graphics
7. Application development, including graphical user interface building.

BASICS:

MATLAB WINDOWS: There are three basic windows which are as follows: MATLAB DESKTOP: This is the window which appears by default when MATLAB is launched. It consists of:

COMMAND WINDOW**:** This is the main window, where commands are written. It is characterized by MATLAB command prompt (>>). The results also appear on this window (except figures, which appear on figure window) which command is written. Commands cannot be edited in this window.

CURRENT DIRECTORY: This appears on the bottom left side of MATLAB desktop. It is where all files are listed. With a mouse right click, you can run M-files, rename, and delete them etc. after selecting a file from here.

WORKSPACE: This sub-window shows all the variables generated so far and also shows their type and size.

COMMAND HISTORY: All commands typed on the MATLAB prompt are recorded here. Also, commands can be selected from here and create as M-file. Thus, it remains records of MATLAB functions run.

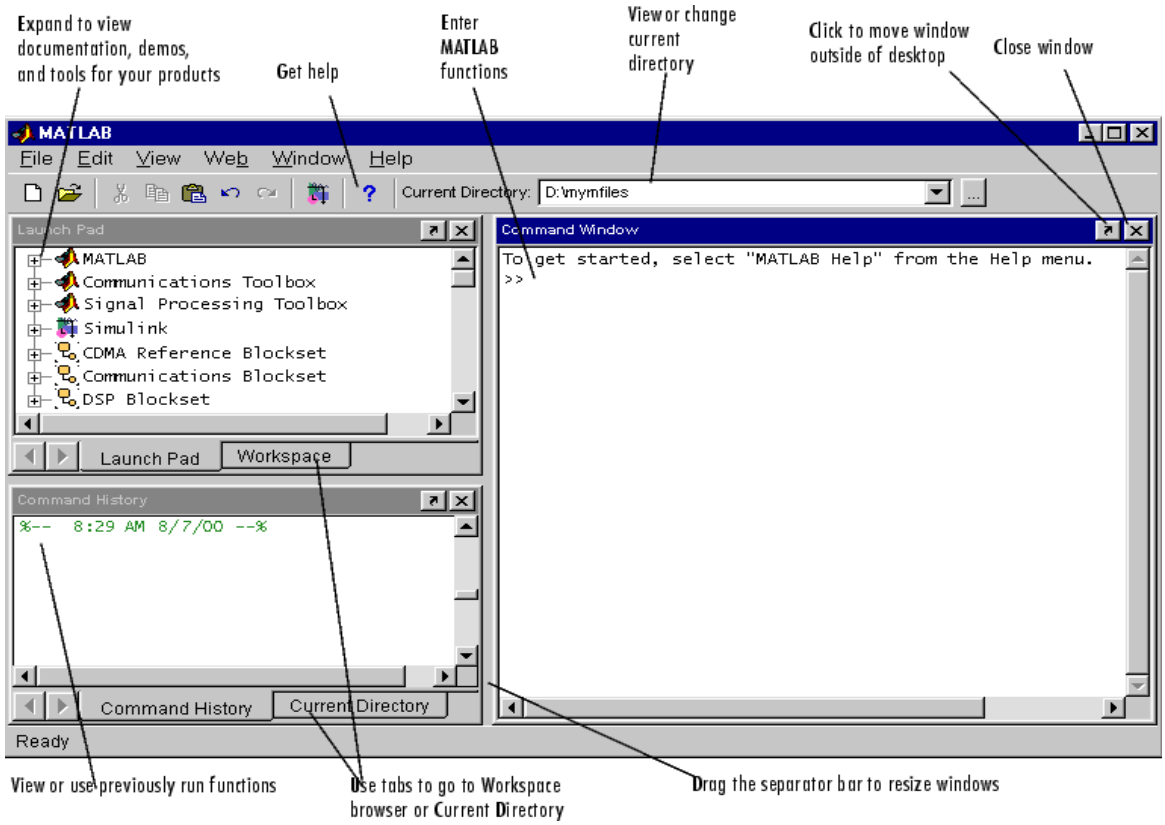
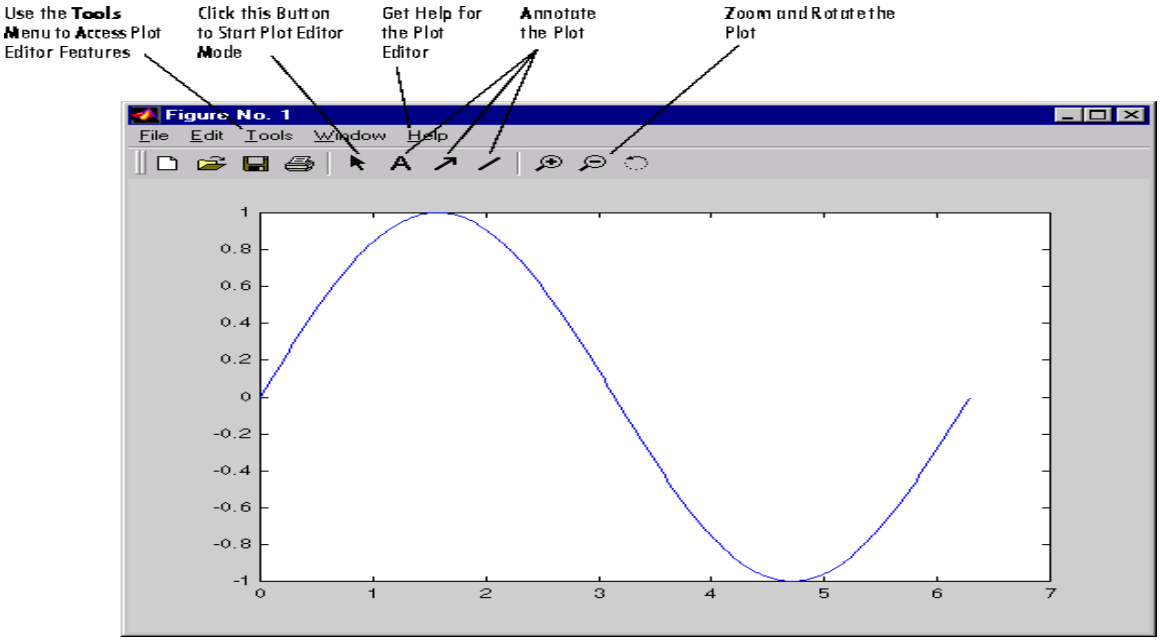


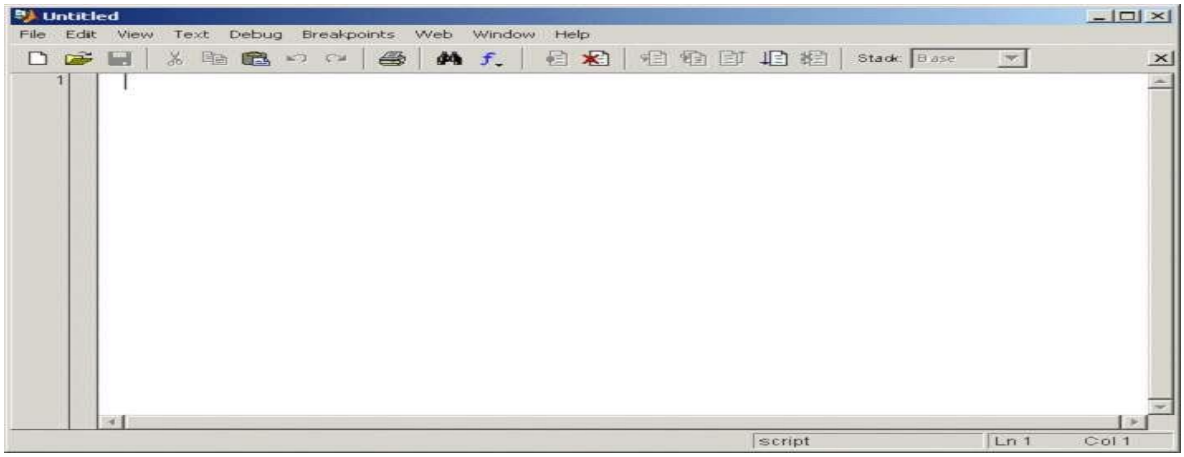
FIGURE WINDOW:

The output of all the commands written on the command window or executed by writing in M-file, whose output is a graph, appears on the window. The user can create as many figure windows as the system memory allows. A figure window is showing sine curve is shown in figure.



EDITOR WINDOW:

This is where we can write, create, edit and save programs in a file. The file is known as M-file. To select editor window, go to file and then select M-file. The programmers written on the file are first saved and then run to get the results. To save and run the programme, go to debug and select 'save and run'. The results appear on the command window. The figures appear on the figure window. An editor window is shown as in figure:



LANGUAGES:

MATLAB is a high-level language that includes matrix-based data structures, its own internal data types, an extensive catalog of functions and scripts, and the ability to import and export to many types of data files, object-oriented programming capabilities, and interfaces to external technologies such as COM, Java, and program written in C and FORTRAN, and serial port devices.

INPUT-OUTPUT:
MATLAB takes input from the screen and rushes output to the screen
i.e., it supports interactive computation. It can read input files and write output files.
Data Type: There is no need to declare a data to be real or complex. When a real number is entered as a variable, MATLAB automatically sets the variable to be real. Fundamental data type is the array.

Dimension: Dimension state is not required in the MATLAB. It is automatic.

Case Sensitivity: It differentiates between lower and upper cases. It is case sensitive.

Output Display: The output of every command appears on the screen unless MATLAB is directed otherwise. A semi-colon (;) suppress the output when used at the end of a command except for the graphics.

GETTING STARTED

STARTING MATLAB: On windows platform, start MATLAB by double clicking the MATLAB shortcut icon on your Windows desktop.

WRITING COMMAND: When you start MATLAB, the MATLAB desktop appears containing tools for managing files, variables, and applications associated with MATLAB. You can start writing your command at the prompt appears on command Window. You can also write command in M-file.

PRINTING GRAPHICS: The simplest way to get print out of the graph is to type print command window after the graph appears in the figure window. Alternatively activate the figure window and then select print from the file menu.

QUITTING MATLAB: To end your MATLAB session, select file >Exit MATLAB in the desktop or type quit in the command window. You can run a script file named finish .m each time MATLAB quits that, for example, executes function to save the workplace, or display a quit confirmation dialog box.

ACCESSORIES

TOOLBOXES: MATLAB features a family of add-on-application- specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others. There are 78 toolboxes available. Thus, we can say MATLAB basically works as a platform and for solving a particular problem, concerned toolbox is required. Few of the toolboxes are as follows:
Communication toolbox
Control system toolbox
Curve fitting toolbox
Data acquisition toolbox
Filter design toolbox
Fuzzy logic toolbox
Instrument control toolbox

Optimization toolbox Statistics toolbox
Symbolic math's toolbox, etc.

SIMULINK: Simulink is a software package that enables you to model, simulate, and analyze systems whose outputs change over time. Such systems are often referred to as dynamic systems. Simulink can be used to explore the behavior of a wide range of real- world dynamic systems, including electrical circuits, shock absorbers, braking systems, and many other electrical, mechanical, and thermodynamic system. This section explains how Simulink works.

Simulating a dynamic system is a two-step process with Simulink. First, a user creates a block diagram, using a block diagram Simulink model editor that graphically depicts time dependent mathematical relationship among the system's inputs, states, and outputs. The user then commands Simulink to simulate the system represented by the model from a specified start time to a specified stop time.

CONCLUSION: Hence the study and learn known facts about study of introduction to MATLAB is done successfully.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What is a MATLAB?
2. What is acronym of MATLAB?
3. What is a command window?
4. What is a workspace in MATLAB?
5. What is an editor window?
6. What are the differences between editor and command windows?
7. Where the data of variables are stored?
8. How to display the output in a command window?
9. What are few toolboxes in MATLAB?
10. What is an extension of MATLAB while saving the file?

EXP. NO.: 02
DATE :

STUDY OF BASIC MATRIX OPERATIONS

AIM: To study about the basic matrix operations.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB software	Version-2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

GIVEN NUMERICAL:

A= 3 2 1
0 3 4
-1 1 -1

B= 1 3 0

2 6 4
-1 0 2

Where A and B are two 3 X 3 matrixes

COMMANDS:

```
>> A= [3  2  1; 0  3  4; -1  1  -1]

>> B= [1  3  0; 2  6  4; -1  0  2]
```

Sum: A+ B
Subtraction: A – B
Multiplication:A * B
Division: A/B or A\B
Inverse of A: inv (A)

RESULTS:

>> A + B % Sum up matrix A and B

Ans:

>> A – B % Subtract matrix A and B

Ans:

>> A * B % multiply matrix A and B

Ans:

>> A/B % Divides matrix A and B (take inverse of B and multiply with A)

Ans:

>> A\B % Divides matrix A and B (take inverse of A and multiply with B)

Ans:

ADDITIONAL COMMANDS AND RESULTS:

>>a= magic (3) % gives 3 X 3 matrix whose sum from any angle is same

Ans:

>>a= rand (3) % gives any 3 X 3 random matrix

Ans:

>> a= ones (3) % gives 3X 3 matrix whose elements are one

Ans:

>> b= 2*ones (3) % multiplication of 2 with ones (3)

Ans:

>> a+2 % summation of 2 with matrix 'a'

Ans:

>> a (2,2) % gives second row and second column element of matrix 'a'

Ans:

>>a (2:3, :) %gives second and third row of matrix 'a'

Ans:

>> a (: 2:3) **% Gives second and third column of matrix 'a'**

Ans:

>> a (2:3,1:2) **%gives second and third row and first and second column of matrix 'a'**

Ans:

>>a (:,2) **% gives second column of matrix 'a'**

Ans:

>>a(:,1:2)=[] **%delete first and second column**

Ans:

>>eye(3) **%gives 3 X 3 matrix whose diagonal are one**

Ans:

>> diag(a) **% gives diagonal element of matrix 'a'**

Ans:

>> b = a' **% gives inverse of matrix 'a'**

Ans:

CONCLUSION: We have studied various commands to solve the basic matrix operations.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What is a Matrix?
2. What is the dimension of a matrix?
3. What are the elements of a matrix?
4. What is a square matrix?
5. What is an orthogonal matrix?
6. What are the basic matrix operations?
7. What is the Transpose of a matrix?
8. What are the commands in MATLAB to solve the various basic matrix operations?
9. What is the operator used to multiply two matrices?
10. What is a rank of matrix?

EXP. NO.: 03
DATE :

TO SOLVE LINEAR EQUATION

AIM: To study and write code about the solving of linear equation.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB software	Version-2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

LINEAR EQUATION:

Solving a linear algebraic equation is easy in MATLAB. It is, perhaps, also the most used computation in science and engineering. We will solve a set of linear algebraic equations given below:

$$5x= 3y-2z+10$$

$$8y+4z= 3x+20$$

$$2x+4y-9z=9$$

PROCEDURE:

STEP 1: Rearrange equations: Write each equation with all unknown quantities on the left-hand side and all known quantities on the right-hand side. Thus, for the equations given above, rearrange them such that all terms involving x, y, and z are on the left-hand side of the equal sign:

$$5x-3y+2z = 10$$

$$-3x+8y+4z = 20$$

$$2x+4y-9z = 9$$

STEP 2: Write the equations in matrix form: To write the equation in the matrix form [A]{x} ={b} where {x} is the vector of unknowns, we have to arrange the unknowns in matrix A and the constants on the right hand side of the equations in vector b.In this particular example, the unknown vector is

$$X=\begin{bmatrix} x \\ y \\ Z \end{bmatrix}$$

The co-efficient matrix is

$$A=\begin{bmatrix} 5 & -3 & 2 \\ -3 & 8 & 4 \\ 2 & 4 & -9 \end{bmatrix}$$

And the known constant vector is

$$B=\begin{bmatrix} 10 \\ 20 \\ 9 \end{bmatrix}$$

Note that the columns of A are simply the coefficients of each unknown from all the three equations.

STEP 3: Solve the matrix equation in MATLAB: Enter the matrix A and vector b, and solve for vector x with x=A\b (note that the \ is different from the division /):

```
>> A= [5    -3    2;   -3    8    4;   2    4    -9];          % enter matrix A

        10

>> b= [20].;                                          % enter column vector b

         9

>> x= A\b                                              % solve for x

x=

   3.4442                                          the \ is used to solve a linear system

   3.1982                                          of equations [A]{x}={b}.

   1.1868

>> c= A* x                                             % check for solution

c=
```

10.0000

20.0000

9.0000

Program: Write a program to solve linear equation

15x= 5y-8z

9y+3z= x+6

10x+4y-z=7

```
>>A=[15   -5  8; -1   9  3;10  4 -1];
>> b = [0
        6
        7]
>>x=A\b;
>> c=A*x
```

RESULT:

CONCLUSION: Hence we solve the given linear equation with the help of MATLAB.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What are linear equations in two variables?
2. How do you solve the linear equation-8N + 4 (1 + 5N)?
3. Is 3x + 5 = 11 a linear equation?
4. What is linear equation in MATLAB?
5. How do you convert linear equations to matrix?
6. What is the matrix representation of system of linear equations?
7. What is the best way to solve linear systems in MATLAB?
8. How to create a matrix of Zero(s) in MATLAB?
9. How to add non-zero elements to a matrix?
10. How to create a matrix in MATLAB?

EXP. NO.: 04
DATE :

SOLUTION OF LINEAR EQUATIONS AND DIFFERENT CASE STUDIES

AIM: To study and write code about the solving of linear equation and different case studies.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version-2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

1. GIVEN NUMERICAL

I). Over determined: Under such conditions, number of equations are more than the unknowns. Over determined systems of simultaneous linear equations are often encountered in various kinds of curve fitting to experimental data.

e.g., + 2y – z = 3 3x – y + 2z = 1 2x -2y +3z = 2 x- y + z = -1

II). Underdetermined: Underdetermined linear systems involve more unknowns than equations. When they are accompanied by additional constraints, they are the purview of linear programming. By itself, the backslash operator deals only with the unconstraint system. The solution is never unique. MATLAB finds a basic solution, which has at most m nonzero components, but even this may not be unique. The particular solution actually computed is determined by the QR factorization with column pivoting. The complete solution of the underdetermined system can be characterized by adding an arbitrary vector from the null space, which can be found using the null function with an option requesting a "rational" basis Z = null (a, 'r'). When the equations are expressed by a* x = b.
It can be confirmed that a*Z is zero and that any vector x, where x = b +Z*q
for any arbitrary vector q satisfies a*x = b.

e.g., x + 2y – z = 3, 3x – y + 2z = 1

COMMANDS:

The numerical given is expressed in matrix form i.e., ax = b and the commands are as follows:

i) Over determined

```
>> a = [1    2   -1; 3   -1   2; 2   -2   3; 1 -1  1];
>> b = [3 ; 1; 2; -1];
>> x = a\b
```

ii) Underdetermined

```
>> a = [1 2 -1; 3 -1  2];
>> b = [3;1];
>> x = a\b
>> Z = null(a, 'r')
```

RESULTS:

iii) Over determined

iv) Underdetermined

CONCLUSION: Hence, we have found out the solution of underdetermined and over determined cases and then verified again.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What do you mean by underdetermined system?
2. What do you mean by over determined system?
3. How do you solve linear equations of under determined system and over determined system manually?
4. What are the commands to solve linear equations of under determined system and over determined system in MATLAB?
5. What are the 3 types of system of linear equation?
6. How many solutions does the linear system have?
7. What is the main idea of linear equations?
8. What is a homogeneous linear system?
9. What is a non-homogeneous linear system?
10. Why linear equation is called linear?

EXP. NO.: 05
DATE :

DETERMINATION OF EIGEN VALUES AND EIGEN VECTORS OF A SQUARE MATRIX

AIM: Determination of Eigen Values and Eigen Vectors of a Square Matrix.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

EIGEN VALUES AND EIGEN VECTORS:

For an ***n*** **x** ***n*** matrix, **A**, the real number λ is called an ***Eigen value*** of A if there exists a nonzero vector **x** in R*n* such that Ax =λ **x**. The vector x is called an eigenvector belonging to λ. The equation Ax =λ **x** is equivalent to (A-λI) **x** = 0, so all of the following are equivalent:

1. λ is an Eigen value of A.
2. (A-λI) **x** = **0** has a nontrivial solution.
3. A-λI is singular.
4. det(A-λI) = 0.

The eigenvectors for λ are the nonzero solutions x to (A-λI)x=0. These vectors together with the 0 vector is called the ***eigen space*** corresponding to eigen value λ. The expression det(A-λI) is a polynomial in λ of degree ***n***, called the

characteristicpolynomial. By property 4, the eigen values are the roots of the ***characteristic equation*** det(A-λI) = 0.

Determining Eigen values and Eigenvectors with MATLAB:

Method 1: In MATLAB we can find the characteristic polynomial of a matrix A by entering **poly(A)**. If A is an *n* x *n* matrix, **poly(A)** is a row vector with *n*+1 elementsthat are the coefficients of the characteristic polynomial. The command **roots(C)** compute the roots of the polynomial whose coefficients are the elements of the vector C. Thus, **roots (poly(A))** returns the Eigen values of A in a column vector.

To find the eigenvectors corresponding to each eigenvalue found above, we needto find the nonzero solutions **x** to (A-I) **x = 0**. One way of doing this in MATLAB isto compute **rref(A-I)** and then use Gauss-Jordan elimination.

Finding the eigenvector and eigenvalues of a square matrix A.

```
» A = [3 2 -2; -3 -1 3; 1 2 0]        Enter matrix A.

A =
     3     2    -2
    -3    -1     3
     1     2     0

» roots(poly(A))                     Compute the roots of characteristic equation
                                     det(A-λI)=0;  returns eigenvalues 2, 1, and -1.
ans =
    2.0000
    1.0000
   -1.0000

» rref(A-2*eye(3))                   Use rref to find the solutions to (A-2I)x = 0
                                     -- the eigenvectors corresponding to λ = 2.
ans =
     1     0     0
     0     1    -1
     0     0     0

» rref(A-1*eye(3))                   Use rref to find the solutions to (A-1I)x = 0
                                     -- the eigenvectors corresponding to λ = 1.
ans =
     1     0    -1
     0     1     0
     0     0     0

» rref(A-(-1)*eye(3))                Use rref to find the solutions to (A-(-1)I)x = 0
                                     -- the eigenvectors corresponding to λ = -1.
ans =
     1     0    -1
     0     1     1
     0     0     0
```

The reduced form of echelon form for A-2I gives the general solution to(A-2I)x =0

as $x=\begin{bmatrix}0\\ r\end{bmatrix} = r\begin{bmatrix}0\\ 1\end{bmatrix}$, the form for eigenvectors corresponding to λ = 2.

Similarly, the reduced echelon forms for A-1I and A-(-1)I allow us to determine

eigenvectors of the form s $\begin{bmatrix}1\\0\\1\end{bmatrix}$ for λ = 1, and t $\begin{bmatrix}1\\-1\\1\end{bmatrix}$ corresponding to λ = -1.

Method 2: Determining the **eigenvalues** and **eigenvectors** in **MATLAB** is to use the ***eig*** function. For an **nxn** matrix A, **eig(A)** returns a n x1 column vector whose elements are the eigenvalues of A. The command in the form.

$$[V\ D] = eig(A)$$

Computes both the eigenvalues and eigenvectors of A. V will be a matrix whose columns are eigenvectors of A and D will be a diagonal matrix whose entries along the diagonal are eigenvalues of A. The ith column of V, V(:, i), is the eigenvector corresponding to the eigenvalue D(i,i). A sample session is shown for the matrix A above.

Finding the eigenvector and eigenvalues of a square matrix A.

```
» eig(A)                          Compute the eigenvalues for A.
                                  Note same answer as with roots(poly(A)).
ans =
  -1.0000
   1.0000
   2.0000

» [V D] = eig(A)                  This form gives us both the eigenvectors in V
                                  and the eigenvalues in D.
V =                               The columns of V are the eigenvectors with
  -0.5774   0.7071   0.0000      norm 1.
   0.5774   0.0000   0.7071
  -0.5774   0.7071   0.7071


D =                               Column 1 of V is the eigenvector corresponding
  -1.0000        0        0       to eigenvalue D(1,1) = -1; column 2 of V is for
        0   1.0000        0       D(2,2) = 1; and column 3 of V is for D(3,3) = 2.
        0        0   2.0000
```

PROGRAM: **Write a program to determine the Eigen vector and Eigen valuesof A = [1 2 3; 4 5 6; 7 8 9]**

```
>> A = [ 1 2  3;  4  5  6; 7  8  9]
>> eig(A)
>> [V D] =eig(A)
```

RESULT:

CONCLUSION: Hence we have written program to determine Eigenvectors and Eigen values of a square matrix and the result has been found out.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What do you mean by Eigen-values of a square matrix?
2. What do you mean by Eigen Vectors of Square Matrix?

3. How do you solve Eigen values and Eigen Vectors of a Square Matrix manually?

4. How do you solve Eigen-values and Eigen-vectors of a square matrix in MATLAB?

5.What is the relationship between Eigen Values and Eigen Vectors?
6.Why do we study Eigen Values and Eigen Vectors?
7.What is special about Eigen Vectors?
8.Where are Eigen Values are used?
9.Are Eigen Values and Eigen Vectors are same?
10.What are Eigen Vectors used for?

EXP. NO.: 06
DATE :

SOLUTION OF DIFFERENCE EQUATIONS

AIM: To Determine the solution of Difference Equations.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

DIFFERENTIAL EQUATIONS:

A **differential equation** is a mathematical equation for an unknown function of one or several variables that relates the values of the function itself andits derivatives of various orders. Differential equations play a prominent role in engineering, physics, economics, and other disciplines.

GIVEN NUMERICAL:

The function 'dsolve' computes symbolic solutions to ordinary differential equations. The equations are specified by symbolic expressions containing the letter 'D' to denote differentiation. The symbols D2, D3…. DN, correspond to the second, third, Nth derivative, respectively. Thus, D2y is d^2/dt^2. The dependent variables are those preceded by D and the default independent variable is t. Note that names of symbolic variables should not contain D. The key issues in this example are the order of the equation and the initial conditions.

To solve the ordinary differential equation, simply type:

y = dsolve ('D3y=y', 'y (0) =1', 'Dy (0) =-1', 'D2y (0) =pi', 'x')

where D3y represents $d3y/dx3$ and D2y (0) represents $d2y/dx2$ at x = 0.

EXAMPLES:

(i) dy/dt = - ay.

(ii) dy/dt = -ay and y(0) = 1.
(iii) d^2y/dt^2 = -a^2y and y(0)=1, dy/dt(Π) =0.
(iv) dy/dx = (xy - y^2)/ x^2 .
(v) dy/dx = Tan(y/x) + (y/x) .

COMMANDS:

(i)

>> y=dsolve('Dy=-a*y') %write ODE in inverted comma.

(ii)

>> y=dsolve('Dy=-a*y', 'y(0)=1') %write ODE in inverted comma followed by initial condition, separated by comma.

(iii)

>> y=dsolve('D2y=-a^2*y', 'y(0)=1, Dy(pi/a)=0')

%write ODE in inverted comma followed by initial conditions.

Note: In all the above cases, the independent variable is 't' by default.

(iv)

>> y = dsolve ('Dy=(x*y –y^2)/ x^2', 'x')

% define independent variable as 'x'

(v)

>> y = dsolve ('Dy=tan(y/x) + y/x)', 'x')

RESULTS:

CONCLUSION: Hence we have written program to determine solution of difference equations and the result has been found out.

Lab Instructor **Faculty**

Date: **Date:**

Viva Questions:

1.How do you solve Differential Equations?
2.What is Differential equations used for?
3.What is First Order Differential Equation?
4.What are the different types of Differential Equations?
5.What is the Order in Differential Equation?
6.What is the difference between ODE and PDE?
7.What is Linear and Non-Linear Differential Equations?
8. What is Homogenous and Non-Homogenous Differential Equations?
9.How can you find if Differential Equation is separable or not?
10.What is the difference between Linear and Homogenous?

EXP. NO.: 07
DATE :

SOLUTION OF DIFFERENTIAL EQUATION USING EULER METHOD

AIM: To Determine the solution of Difference Equation using Euler Method.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

DIFFERENTIAL EQUATION USING EULER METHOD:

Let's consider a first-order differential equation:

$$y'(t) + ay(t) = r \quad with\ y\ (0) = y_0 \qquad (1)$$

It has the following form of analytical solution

$$y(t) = (y0 - r/\mathrm{a})e^{-at} + r/\mathrm{a} \qquad (2)$$

which can be obtained by using a conventional method or the Laplace transform technique.

First of all, we have to replace the derivative $y'(\mathrm{t}) = dy/dt$ in the differential equation by a numerical derivative, where the step size h is determined based on the accuracy requirements and the computation time constraints. Euler's method approximates as

$$\frac{y(t+h) - y(t)}{h} + ay(t) = r$$

$$y\,(t+h) = (1-ah)\,y(t) + hr\;with\;y\,(0)$$

And solve this difference equation step-by-step with increasing t by h each time from t = 0.

$$y(h) = (1-ah)y(0) + hr = (1-ah)y_0 + hr$$

$$y(2h) = (1-ah)y(h) + hr = (1-ah)^2\,y_0 + (1-ah)hr + hr$$

$$y(3h) = (1-ah)y(2h) + hr = (1-ah)^3\,y_0 + \sum_{m=0}^{2}(1-ah)^m\,hr$$

This is a numeric sequence $\{y(kh)\}$ which we call a numerical solution of Eqn (1).
To be specific, let the parameters and the initial value of Eqn (1) be a = 1, r = 1, and $y0 = 0$. Then the analytical solution of Eqn (2) becomes
$y(t) = 1 - e{-}at$

PROGRAM:

% Euler method to solve a 1st order differential equation of
$y(t) = 1 - e{-}at$%

```
a = 1;          r = 1; y_0 = 0 ; tf = 2;

t = [0:0.01:tf];

yt = 1-exp(-a *t);

plot (t, yt, 'k'), hold on

klasts = [8 4 2]; hs = tf. /klasts;

y(1) = y_0;

for int = 1: 3

klast = klasts (int) ; h = hs(int) ; y(1) = y_0;
for k = 1: klast

    y(k+1) = (1-a*h)*y(k) + h* r ;

    plot ([k-1 k]*h, [y(k) y(k+1)], 'b' , k* h , y(k+1), 'r')

    if k< 4, pause ; end
```

```
        end
  end
```

RESULT & OBSERVATIONS:

2.GIVEN NUMERICAL:

dy/dx = (y - x)/(y + x)

COMMANDS:

```
>> b = 3; a = 0;m = 4; x = 0; y = 1;
>> h = (b-a)/m;
>>x = a:h:b;
>> for j = 1:m;
>> y(j+1)=y(j)+h*((y(j)-x(j))/ (y(j)+x(j)));
end
>>E = [x' y']
```

RESULTS:

CONCLUSION: Hence we have written program to determine solution of differential equation using Euler method and the result has been found out.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What are the advantages with Euler Method?
2. Is Euler Method Single Step Method?
3. Is Euler Method the most accurate method?
4. Compare Euler Method with other method?
5.What is the Euler Method formula?
6.What are Euler differential equations?
7. What is Euler method used for?
8.What is the disadvantage of Euler's method?
9.Can you work with first order differential equations using Euler's method?
10.Is Euler method a stable?

EXP. NO.: 08
DATE :

SOLUTION OF DIFFERENTIAL EQUATION USING 4th ORDER RUNGE-KUTTA METHOD

AIM: To Solve a differential equation using 4th order Runge-Kutta method.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

4th ORDER RUNGE-KUTTA METHOD:

The fourth-order Runge -Kutta (RK4) method having a truncation error of O(h4) is one of the most widely used methods for solving differential equations. Its algorithm is described below:

$$y_{k+1} = y_k + \frac{h}{6}(f_{k1} + 2f_{k2} + 2f_{k3} + f_{k4}) \quad (1)$$

Where

$$f_{k1} = f(t_k, y_k) \quad (2)$$

$$f_{k2} = f(t_k + \frac{h}{2}, y_k + f_{k1}\frac{h}{2}) \quad (3)$$

$$f_{k3} = f(t_k + \frac{h}{2}, y_k + f_{k2}\frac{h}{2}) \quad (4)$$

$$f_{k4} = f(t_k + h, y_k + f_{k3}h) \quad (5)$$

PROGRAM:

```
function [t, y] = ode_RK4(f , tspan , y0, N, varargin)

% Runge- Kutta method to solve vector differential equation y'(t) = f(t, y(t))
```

```
% for tspan = [t0, tf] and with the initial value y0 and N time steps
if nargin < 4 |          N<= 0, N = 100; end
if nargin < 3, y0 = 0; end
y(1, :) = y0(:)' ;                                   % make it a row vector
h = (tspan(2) – tspan(1)) / N ; t = tspan(1) + [0: N]' * h ;
for k = 1: N

f1 = h* feval (f , t(k), y(k, :) , varargin{:}); f1 = f1(:)';
f2 = h* feval (f , t(k) + h/2 , y(k, :) + f1/2 , varargin{:}); f2 = f2(:)';
f3 = h* feval (f , t(k) + h/2 , y(k, :) + f2/2 , varargin{:}); f3 = f3(:)';
f4 = h* feval (f , t(k) + h , y(k, :) + f3 , varargin{:}); f4 = f4(:)';
y(k + 1 , :) = y(k, :) + (f1 + 2*(f2 +f3) + f4 )/6 ;
end
```

RESULTS & DISCUSSIONS:

CONCLUSION: Hence we have written program to determine solution of differential equation using 4^{th} Order Runge-Kutta method and the result has been found out.

Lab Instructor **Faculty**

Date: **Date:**

Viva questions:

1. What is the difference between Runge-Kutta Method and Euler Method?
2. What are the advantages with Runge-Kutta Method?
3. Is Runge-Kutta Method is Single Step Method?
4. Why Runge-Kutta Method is the most accurate method?
5. How does Runge-Kutta 4th order method work?
6. How many steps does the fourth order?
7. Is Runge-Kutta 4th order implicit?
8. How many steps does the fourth order?
9. What is Runge-Kutta 2nd order?
10. What is 3rd order Runge-Kutta method?

EXP. NO.: 09
DATE :

DETERMINATION OF ROOTS OF A POLYNOMIAL

AIM: To obtain and determination of roots of a polynomial.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

POLYNOMIAL:

Polynomials are functions that have the form

$$f(x) = a_0+a_1x+\ldots+a_nx^n$$

The coefficients $a_1, a_2\ldots, a_n$ are often real numbers and n which is a nonnegative integer, is the *degree* or *order* of the polynomial**.**

IN MATLAB:

Polynomials are described by using a row vector of the coefficients of the polynomial beginning with the highest power of *x* and inserting zeros for "missing" terms:

$$f(x)= 9x^3-5x^2+3x+7$$

$$g(x)=6x^2-x+2$$

f= [9 -5 3 7]

g= [6 -1 2]

ROOTS OF A POLYNOMIAL:

Roots of a polynomial

$$f(x) = a_0 + a_1 x + \ldots + a_n x^n$$

are the values x of for which $f(x) = 0$. For example, the roots of $f(x) = x^2 - 3x + 2$ are -1 and -2. There are n roots of a polynomial with degree n.

The command **"*roots*"** determines the roots of a polynomial. The usage of the function is:

r= roots (p)

where r is a column vector with the roots and p is a row vector with the coefficients of the polynomial.

PROGRAM:

Find the roots of the polynomial

f(x)=3x6+15x5-10x3+4x

>> p = [3 15 0 -10 0 4 0]

>> roots(p)

RESULT:

CONCLUSION: Hence we solve the given polynomial equation with the help of MATLAB.

Lab Instructor **Faculty**

Date: **Date:**

Viva Questions

1. What is a polynomial?
2. What are polynomials in math?
3. What is the Meaning of Polynomial?
4. How do you identify polynomials?
5. Which expressions are polynomials?
6. Which algebraic expression is a polynomial?
7. What do you mean by roots of a polynomial?
8. How do you simplify polynomials?
9. Which command is used to find the roots of a polynomial?
10. Solve the equation $4x^3-3x^2+5x-7$

EXP. NO.: 10
DATE :

DETERMINATION OF POLYNOMIAL USING METHOD OF LEAST SQUARE CURVE FITTING

AIM: To obtain and determination of polynomial Using Method of Least Square Curve Fitting**.**

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

GIVEN NUMARICAL

Fit a straight-line y = a+ bx to the following data:

x: 1 2 3 4 5

y: 14 27 40 55 68

COMMANDS

i) >>x = [1 2 3 4 5]; %write the elements with spacing

>> y = [14 27 40 55 68];
>> polyfit(x,y,1) %fits the data to linear equation ii) >> x=[0 1 2 3 4];

>> y= [1 5 10 22 38];

>> polyfit(x,y,2) %fits the data to quadratic equation

>>plot(x, y)

```
>>title ('curve fitting')
```

RESULT:

CONCLUSION: Hence we solve the given polynomial Using Method of Least Square Curve Fitting with the help of MATLAB.

Lab Instructor **Faculty**

Date: **Date:**

Viva Questions

1. What is Curve fitting?
2. What is the method of least squares for curve fitting?
3. What is polynomial fitting?
4. What is the formula for least square method?
5. What is polynomial curve?
6. What is the use of curve fitting?
7. Which method can be used to fit a curve through the given data points?
8. How do you decide order of polynomials?
9. What are polynomial models?
10. What are polynomial features?

EXP. NO.: 11
DATE :

DETERMINATION OF TIME RESPONSE OF AN R-L-C CIRCUIT

AIM: To Obtain Determination of Time Response of An R-L-C Circuit

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

GIVEN NUMARICAL

An R-L-C circuit has R = 180 ohms, C = 1/280 farads, L = 20 henries and an applied voltage E(t) = 10 sin t. Assuming that no charge is present but an initial current of I ampere is flowing at t = 0 when the voltage is first applied, find q and i =dq/dt at any time t. q is diven by the differential equation. $L\ d\ 2q/dt\ 2 + R\ dq\ dt + q\ c = E(t)$

COMMANDS

>> syms q t **% declare q, t as symbolic**

>> q = dsolve ('20*D2q +180*Dq +280*q = 10*sin(t)', 'q(0)=0')

>> simplify(q) **%simplify the result**

>> pretty(q) **%print in readable form**

>> i=diff(q) **%current(i) is differentiation of 'q' wrt 't'**

>>pretty (i

RESULTS

CONCLUSION: Hence we solve the given Time Response of an R-L-C Circuit with the help of MATLAB.

Lab Instructor **Faculty**

Date: **Date:**

Viva Questions

1. What is an R-L-C circuit?
2. What do you mean by time response of a circuit?
3. What is step response of RLC circuit?
4. What is the response condition of RLC?
5. What is the step response of a circuit?
6. What type of response will the parallel RLC circuit produce?
7. What will be the response of a series RLC circuit if the roots of its characteristic equation are complex conjugate?
8. What is parallel RLC circuit?
9. What is Q in a circuit?
10. What is XL and XC in RLC circuit?

EXP. NO.: 12
DATE :

SIMULTANEOUS EQUATIONS: GAUSS SEIDEL METHOD

AIM: To Solve Simultaneous Equations using Gauss Seidel method.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

GAUSS SEIDEL METHOD:

Perform the partial pivoting of equation i.e., arranging the equation such that diagonal element has maximum value.

Rewrite the equation 1st as
X1=f (x2, x3, ...xn) Rewrite nth equation as
xn=f (x1, x2....xn-1) These are known as characteristic equation

Approximate initial solution as x1=0, x2=0, xn=0. (Values taken arbitrarily)

Put initial value in equation (i) to obtain x1. Now use the latest value in all next equation to get x2, x3 etc. Now it is improved solution.

Repeat step 4 with new values till desired accuracy is achieved i.e., difference between previous value of x and this value of x is minimum.

EXAMPLE:

$x1 + 20x2 + 9x3 = -23$
$2x1 - 7x2 - 20x3 = -57$
$20x1 + 2x2 + 6x3 = 28$

PROGRAM:

```
function gausseidel

        clc;
        clear;
   [a, c, x] =read equation;
    [a, c] =pivote (a, c);
x=iterate (a, c, x);
end
function [a,c,x]=read_equation
        n=input('\nEnter no of equation:');
        for i=1:1:n
        for j=1:1:n
       fprintf('\na[%d][%d]=',i,j);
       a(i,j)=input('');
  end
  fprintf('\nc[%d]=',i);
  c(i)=input('');
   x(i)=0;
 end
end
function [a,c]=pivote(a,c)
       n=length(c);
      for i=1:1:n-1
            for k=i+1:1:n
     if (abs(a(k,i))>abs(a(i,i)))
          for j=1:1:n
          temp=a(i,j);
          a(i,j)=a(k,j);
          a(k,j)=temp;
         end
        temp=c(i);
        c(i)=c(k);
        c(k)=temp;
        end
   end
 end
end
function x=iterate(a,c,x)
      n=length(c);
      acc=input('\nEnter accuracy to be achived:');
      err=1;
       k=1;
    while (abs(err)>acc)
       fprintf('\n%d ',k);
       for i=1:1:n
             temp=c(i);
           for j=1:1:n
                if (i~=j)
```

```
        temp=temp-a(i,j)*x(j);
         end
   end
    err=x(i)-temp/a(i,i);

    x(i)=temp/a(i,i);
    fprintf('%f ',x(i));
     end
     k=k+1;
   end
end
```

RESULTS:

CONCLUSION: Hence we solve the Simultaneous Equations using Gauss Seidel Method.

Lab Instructor **Faculty**

Date: **Date:**

Viva Questions:

1. What is limitation of Gauss-Seidel method?
2. What is the condition for applying Gauss-Seidel method?
3. What are the advantages of Gauss-Seidel method?
4. What are the disadvantages of Gauss-Seidel method?
5. What is the limitation of Gauss-Seidel method?
6. Why do we use Gauss-Seidel method?
7. Who introduced Gauss-Seidel method?
8. What is difference between Gauss elimination and Gauss-Seidel method?
9. How do you use the Gauss-Seidel method?
10. What is the condition for applying Gauss-Seidel method?

EXP. NO.: 13
DATE :

SIMULTANEOUS EQUATIONS: GAUSS ELIMINATION METHOD

AIM: To Solve Simultaneous Equations using Gauss Elimination method.

APPARATUS/ SOFTWARE REQUIRED:

SR.NO	APPARATUS/SOFTWARE	SPECIFICATION	QUANTITY
1	Personal Computer	Windows	1
2	MATLAB Software	Version 2018	1

PROCEDURE:

1. Let's start the computer and go into the MATLAB.
2. In MATLAB go to editor window where we perform coding according to the requirements and given equations.
3. While writing the code, the variables which can store the data is visible in workspace which helpful to retrieve the data to the editor window for corresponding equations.
4. After coding, RUN the program and the results are visible in a command window.
5. The results are stored in a workspace.
6. Compare the results get from command window with the theoretical analysis.
7. After completion of experiment, close the MATLAB software and shutdown the PC safely.

GAUSS ELIMINATION METHOD:

Gauss Elimination with partial pivoting
System of linear equation is represented as:
A11X1 + A12X2 + A13X3 .....A1nXn
A21X1 +A22X2 +A23X3.....A2nXn
An1X1+An2X2+An3X3 .....AnnXn

The above can be represented as:
[A][X]=[C]
[A]->Coefficient matrix
[X]->Variable matrix
[C]->Constant matrix

Step 1: Perform partial pivoting
Arrange the system of equation in such a way that diagonal elements will have maximum absolute value.

Step a: Find maximum absolute value of coefficient of X 1 from all equation and place that equation in first row.

Step b: Find maximum absolute value of coefficient X2 from rest of the equation and place that equation in second row.

Step n-1: find maximum absolute value of coefficient of Xn-1from rest of the equation and place the equation in (n-1)th row.

Now the system after partial pivoting as:
P11X1+P12X2+P13X3+.....P1nXn=PC1 P21X1+P22X2+P23X3+.....P2nXn=PC2
Pn1X1+Pn2X2 +Pn3X3+.....PnnXn=PCn

Step 2: reduce the system of equation to upper triangular form

Step a: Using first row make all elements in the first column below the first row to zero . Rk=Rk-mk*R1 where mk=Pk/P11 where k=2 to n

Step b: Using second row make all elements in second column below second row zero.Rk=Rk-mk*R2 where mk=Pk2/P22 .

Step n-1: Using (n-1)th row make all elements in (n-1)th column below (n-1)throw zero. Rk=Rk-mk*Rn-1 where mk=Pk2/Pn-1 where k= n to n .

Now system after reducing to upper triangular matrix

Step 2: Back substitution method

i. **Step a:** From last row Xn=UCn/U

ii. **Step b:** From second last row

$$x_{n-1} = \frac{UC_{n-1} - x_n U_{n-1,n}}{U_{n-1,n-1}}$$

iii. **Step n:** So in general

$$x_i = \frac{UC_i - \sum_{j=i+1}^{n} x_j * U_{i,j}}{U_{i,i}}$$

EXAMPLE:

1.48x1 + 0.93x2 - 1.3x3 = 1.03

2.51x1 + 1.48x2 + 4.53x3 = 0.05

3.04x1 + 2.68x2 - 1.48x3 = -0.53

PROGRAM:

```
function gausselimination

    clc;
   clear;
   [a,c]=read_equation;
   [a,c]=pivote(a,c);
```

```
    [x]=backsubstitution(a,c);
    fprintf('\nsolution is\n');
     x'
end
function [a,c]=read_equation
    n=input('\nEnter no of equation:');

    for i=1:1:n
    for j=1:1:n
          fprintf('\na[%d][%d]=',i,j);
         a(i,j)=input('');
end
       fprintf('\nc[%d]=',i);
      c(i)=input('');
end
end
function [a,c]=pivote(a,c)
       n=length(c);
      for i=1:1:n-1
      for k=i+1:1:n
    if (abs(a(k,i))>abs(a(i,i)))
         for j=1:1:n
               temp=a(i,j);
             a(i,j)=a(k,j);
             a(k,j)=temp;
          end
        temp=c(i);
       c(i)=c(k);
        c(k)=temp;
      end
  end
  end
end
function [a,c]=uppertriangular(a,c)
        n=length(c);
       for i=1:1:n-1
       for k=i+1:1:n
         temp=a(k,i)/a(i,i);
        for j=1:1:n
        a(k,j)=a(k,j)-temp*a(i,j);
end
      c(k)=c(k)-temp*c(i);
         end
     end
end
function [x]=backsubstitution(a,c)
          n=length(c);
          for i=n:-1:1
          temp=c(i);
         for j=i+1:1:n
```

```
            temp=temp-a(i,j)*x(j);
        end
        x(i)=temp/a(i,i);
    end
end
```

RESULT:

CONCLUSION: Hence we solve the Simultaneous Equations using Gauss Elimination Method.

Lab Instructor **Faculty**

Date: **Date**

Viva Questions:

1. What is limitation of Gauss-Elimination method?
2. What is the condition for applying Gauss- Elimination method?
3. What are the advantages of Gauss- Elimination method?
4. What are the disadvantages of Gauss- Elimination method?
5. What is the limitation of Gauss- Elimination method?
6. Why do we use Gauss-Elimination method?
7. Who introduced Gauss- Elimination method?
8. What is difference between Gauss elimination and Gauss-Seidel method?
9. How do you use the Gauss- Elimination method?
10. What is the condition for applying Gauss- Elimination method?