Islamic University of Technology (IUT)

Organization of Islamic Cooperation (OIC)

Department of Electrical and Electronic Engineering (EEE)

Course No.	Math 4522
Course Name	Numerical Methods Lab.
Experiment No.	01
Experiment Name	Introduction to the Sources of Error in Numerical Methods

Objective

- To be familiarized with the different types of measurement techniques for measuring the errors in the numerical methods.
- To calculate different numerical errors while approximating Maclaurin series.
- To learn different ways of performing the task in MATLAB.
- To learn how to use visualization to represent the results properly.
- To be familiarized with the concept of Vectorization, Time Complexity, and Data Viz.

Theory

Let us start with outlining different types of errors that occur in numerical analysis.

1. **True Error**: It is the difference between the true/exact value and the approximate value. In real world scenarios, the true value is rarely available. It can only be known when the functions can be solved analytically. When the true value is unknown, the best estimation can be considered as the true value.

   True Error, E_t= True value – Approximate value

2. **Relative True Error:** It is defined as the ratio between the true error and the true value. It is expressed as a percentage.

$$\text{Relative True Error}, \in_t = \frac{\text{True Error}}{\text{True value}}$$

Prepared by

ASIF NEWAZ
Lecturer, Department of EEE, IUT

Exercise-01: Suppose that you have the task of measuring the lengths of a bridge and a rivet and come up with 9999 cm and 9 cm, respectively. If the true values are 10,000 cm and 10 cm, respectively, compute true error and relative true error.

3. **Approximation Error:** It is defined as the difference between the current approximation and the previous approximation.

   Approximate Error, E_a = Current Approximation – Previous Approximation

4. **Relative Approximation Error:** It is defined as the ratio between the approximation error and the present approximation.

   Relative Approximation Error, $\in_a = \frac{\text{approximation error}}{\text{Current Approximation}}$

- These errors can be positive or negative depending on the scenario. Sometimes, we are not concerned with the sign of the error. In such cases, absolute value should be utilized.

Exercise-02: In mathematics, functions can often be represented by infinite series. For example, the exponential function can be computed using –

$$\boldsymbol{e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots}$$

This is called a *Maclaurin series* expansion. The approximation becomes a better and better estimate of the true value of $\boldsymbol{e^x}$ as more terms are added to the sequence. For each term added, calculate the relative true error and the relative approximation error.

5. **Round-off Error:** It occurs due to approximation in the numerical representation of a number. For instance, π = 3.14159265358979323846… . To represent such irrational numbers, we often approximate it up to certain decimal points. This kind of approximation causes the round-off error. In the above example, rounding-off the value of pi up to four decimal points results in 3.1416.

Prepared by

ASIF NEWAZ
Lecturer, Department of EEE, IUT

Programming languages usually have certain built-in functions to allow rounding off to certain decimal points. In MATLAB, using the format command (e.g., *format short, format long*), you can control the precision. While the '*round*' function allows to round-off to certain decimal places.

- *round (pi)* -- rounds to nearest integer
- *round (pi, 3)* -- rounds to 3 decimal points
- look into – *floor, fix, ceil*

6. **Truncation Error:** It is caused by truncating a mathematical expression like the Maclaurin series in exercise-02. The series has infinite number of terms. When using this series to calculate $\boldsymbol{e^x}$, only a finite number of terms can be used. This will lead to truncation error.

Programming Exercise:

- In this exercise, we will calculate different types of errors while approximating the Maclaurin series for e^x.

```
format long

n=20;   % the number of terms to use
x=0.5;  % value of x

true_value= exp(x)

%% solution

%% way - 01 (loop - substandard)
tic

for i=1:n
   vec1(i)=x^(i-1) / factorial(i-1);
end
vec1_fin= cumsum(vec1);
toc
```

```
%% way - 02 (loop)
tic

vec2= zeros(1,n);    % pre-allocation -> to store the values of 15 iterations
for i=1:n
   vec2(i)=x^(i-1) / factorial(i-1);
end
vec2_fin= cumsum(vec2);
toc

%% way - 03 (vectorized)
tic

p= 0:19;          % a vector with the powers of x for vectorized operation
vec3= x.^p ./ factorial(p);   % using element-wise operation
vec3_fin= cumsum(vec3);
toc

%% Calculating the errors

check_equality= isequal(vec1_fin,vec2_fin,vec3_fin)  % checking whether all three approaches provided the same
result

true_error= true_value - vec_fin(end)
relative_true_error = (true_error/true_value) *100

approx_error= diff(vec3_fin);
relative_approx_error= (approx_error./ vec3_fin(2:end))*100


%% plotting
% plotting is a very useful way of representing your findings and results. It's more of an art than coding.
% let’s look at different ways of plotting and understand the difference

subplot(2,2,1)
plot(1:n,vec3_fin)

subplot(2,2,2)
plot(1:n, vec3_fin,'Marker','*', 'MarkerEdgeColor','r')

subplot(2,2,3)
plot(1:n, vec3_fin, 'Marker','o', 'MarkerEdgeColor','r')
xlabel('no. of iteration')
hold on
yline(true_value,'--g')
hold off
```

Prepared by

ASIF NEWAZ
Lecturer, Department of EEE, IUT

```
subplot(2,2,4)
plot(2:n, approx_error, 'Marker','p')
xlabel('no. of iteration')
ylabel('approximation error')

% check matlab documentation of the 'plot' function for specifics
```

Assignment

- Calculate different types of numerical errors while approximating the Maclaurin series for $\sin(x)$. Use visualization to present the results. Also create a table to display the results of each iteration in tabular form.

Prepared by

ASIF NEWAZ
Lecturer, Department of EEE, IUT