EM314 - NUMERICAL METHODS ASSIGNMENT - INTEGRATION

Hisni Mohammed M.H.

E/15/131

2. **MATLAB code ( trapezoidal.m )**

function [ I, Iexact, RE] = trapezoidal( f, a, b, numSeg );

Step = ( b-a )/numSeg;

X = a:Step:b;

I = 0;

for i = 1:length( X )-1

I = I + f( X( i ) ) + f( X( i+1 ) );

end

I = I\*Step\*.5;

Iexact = integral(f,a,b);

RE = 100\*( Iexact – I )/Iexact;

End

1. **MATLAB code ( f.m )**

function res = f(x)

res = (1-x-4.\*x.^3+2.\*x.^5);

end

**MATLAB code ( Q1b.m )**

for i=2:4

[I,Iexact,RE] = trapezoidal(@f,0,4,i);

fprintf('\nSegments = %d \nEstimateValue = %f \nExactValue = %f

\nPRE = %f %%\n',i,I,Iexact,RE);

End

**Output of the code**

Segments = 2

EstimateValue = 1852.000000

ExactValue = 1105.333333

PRE = -67.551267 %

Segments = 3

EstimateValue = 1447.720165

ExactValue = 1105.333333

PRE = -30.975889 %

Segments = 4

EstimateValue = 1300.000000

ExactValue = 1105.333333

PRE = -17.611580 %

1. **MATLAB code ( simpsonsOneThree.m )**

function [I,Iexact,RE] = simpsonsOneThree(f, a, b,numSeg);

Step = (b-a)/numSeg;

X = a:Step:b;

I = 0;

for i = 1:2:length(X)-2

I = I + f( X( i ) ) + 4\*f( X( i+1 ) ) + f( X( i+2 ));

end

I = Step\*I/3;

Iexact = integral(f,a,b);

RE = 100\*(Iexact - I)/Iexact;

end

1. **MATLAB code ( Q1d.m )**

for i=2:2:6

[I,Iexact,RE] = simpsonsOneThree(@f,0,4,i);

fprintf('\nSegments = %d \nEstimateValue = %f \nExactValue = %f

\nPRE = %f %%\n',i,I,Iexact,RE)

end

**Output of the code**

Segments = 2

EstimateValue = 1276.000000

ExactValue = 1105.333333

PRE = -15.440290 %

Segments = 4

EstimateValue = 1116.000000

ExactValue = 1105.333333

PRE = -0.965018 %

Segments = 6

EstimateValue = 1107.440329

ExactValue = 1105.333333

PRE = -0.190621 %

1. **MATLAB code ( compositeSimpsons.m )**

For odd number of segments combination of Simpsons 1/3rd and 3/8th rule is used.

function [PRE] = compositeSimpsons(f, a, b,numSeg);

Step = (b-a)/numSeg;

X = a:Step:b;

IOneThree = 0;

for i = 1:2:length(X)- 5

IOneThree = IOneThree + f( X( i ) ) + 4\*f( X( i+1 ) ) + f( X( i+2 ));

end

IOneThree = Step\*IOneThree/3;

i = length(X)- 3;

IThreeEight = f( X( i ) ) + 3\*f( X( i+1 ) ) + 3\*f( X( i+2 ) ) + f( X( i+3 ) );

IThreeEight = ( 3\*Step\*IThreeEight ) / 8;

I = IOneThree + IThreeEight;

Iexact = integral(f,a,b);

PRE = 100\*(Iexact - I )/Iexact;

end

**MATLAB code ( Q1e.m )**

fprintf('\nSegments\tTrapezoidalRule\tSimpsonsRule\n');

for i=2:15

[I,Iexact,TRPRE] = trapezoidal(@f,0,4,i);

if ( rem(i,2) == 0 )

[I,Iexact,SRPRE] = simpsonsOneThree(@f,0,4,i);

else

SRPRE = compositeSimpsons(@f,0,4,i);

end

fprintf('\t%d\t\t%.2f\t\t\t%.2f\n',i,TRPRE,SRPRE);

end

**Output of the code**

Segments TrapezoidalRule SimpsonsRule

2 -67.55 -15.44

3 -30.98 -6.86

4 -17.61 -0.97

5 -11.33 -0.81

6 -7.89 -0.19

7 -5.80 -0.19

8 -4.45 -0.06

9 -3.52 -0.06

10 -2.85 -0.02

11 -2.36 -0.03

12 -1.98 -0.01

13 -1.69 -0.01

14 -1.46 -0.01

15 -1.27 -0.01

Step size reduces when segment size is increasing. With the step size convergence rate is higher when Multiple Application of Simpsons 1/3rd rule used than when Composite Trapezoidal rule is used.

1. **MATLAB code ( f2.m )**

function res = f2(x)

m=1;

%m=1.5;

%m=2;

n=2;

%n=2.5;

%n=3;

res = ( x.^(m-1) ).\*( (1-x).^(n-1) );

end

**MATLAB code ( Q2a.m )**

fprintf('Exact Value = %f\n', integral( @f2,0,1 ));

fprintf('\t\t\tTrapezoidal\t\t\tSimpsons\n')

fprintf('Segments\tEstimate\tPRE(%%)\tEstimate\tPRE(%%)\n');

for i=2:6

[ TRI, Iexact, TRPRE ] = trapezoidal( @f2, 0, 1, i );

if ( rem(i,2) == 0 )

[ SRI, Iexact, SRPRE ] = simpsonsOneThree(@f2,0,1,i);

else

[ SRI, SRPRE ] = compositeSimpsons(@f2,0,1,i);

End

fprintf('%4d\t\t%f\t%.2f\t%f\t%.2f\n',i,TRI,TRPRE,SRI,SRPRE);

end

**Output of the code**

**β( 1, 2 )**

Exact Value = 0.500000

Trapezoidal Simpsons

Segments Estimate PRE(%) Estimate PRE(%)

2 0.500000 0.00 0.500000 0.00

3 0.500000 0.00 0.500000 0.00

4 0.500000 0.00 0.500000 0.00

5 0.500000 0.00 0.500000 0.00

6 0.500000 0.00 0.500000 0.00

**β( 1.5, 2.5 )**

Exact Value = 0.196350

Trapezoidal Simpsons

Segments Estimate PRE(%) Estimate PRE(%)

2 0.125000 36.34 0.166667 15.12

3 0.157135 19.97 0.176777 9.97

4 0.170753 13.04 0.186004 5.27

5 0.177980 9.36 0.189065 3.71

6 0.182347 7.13 0.190751 2.85

7 0.185222 5.67 0.191965 2.23

8 0.187232 4.64 0.192725 1.85

9 0.188702 3.89 0.193343 1.53

10 0.189816 3.33 0.193761 1.32

**β( 2, 3 )**

Exact Value = 0.083333

Trapezoidal Simpsons

Segments Estimate PRE(%) Estimate PRE(%)

2 0.062500 25.00 0.083333 0.00

3 0.074074 11.11 0.083333 0.00

4 0.078125 6.25 0.083333 0.00

5 0.080000 4.00 0.083333 0.00

6 0.081019 2.78 0.083333 0.00

7 0.081633 2.04 0.083333 0.00

8 0.082031 1.56 0.083333 0.00

9 0.082305 1.23 0.083333 0.00

10 0.082500 1.00 0.083333 0.00

1. From above table, for same number of segments error is relatively less, when Simpsons Rules are used than Composite trapezoidal rule. And in Simpsons rule convergence is higher than the trapezoidal rule.