

Project 2 Report

Engineering Analysis and Computation I

Ahmed Jaheen - 900212943

Mina Yasser - 900214039

Mohamed ElFahla - 900192563

*Submitted to Dr. Mohamed Rashed
This report is prepared for ENGR3202, section 1*



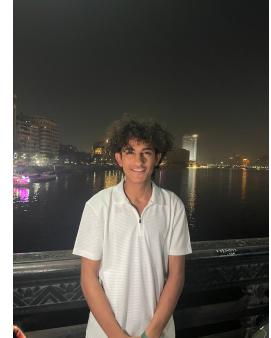
SCHOOL OF SCIENCE AND ENGINEERING
THE AMERICAN UNIVERSITY IN CAIRO

21/07/2023

List of contributors

1. Ahmed Jaheen

Contributions: Wrote `main.m`, `newtonRaphson.m`, `CalculateF.m`, `CalculateJacobian.m`. Also, the latex code for this report and the report.



2. Mina Yasser

Contributions: Wrote `gaussElimination.m`, `predictPlateSize.m`. Also, the report.



3. Mohamed ElFahla

Contributions: Wrote `gaussJordan.m`, `plotRelativeErrors.m`, `plotPressureVsRadius.m`. Also, the report.



Problem Definition

The problem involves developing a **MATLAB** code to calculate the average depth of a lake based on discrete depth measurements taken at specific points within the lake. The lake's plan is represented as a set of data points, each comprising 3-dimensional coordinates (x, y, z) representing the location and depth of the lake at that point.

The task is to employ numerical integration methods, specifically the Composite 1/3 **Simpson's rule**, **Single Trapezoidal rule**, and **Single 3/8 Simpson's rule**, to approximate the lake's average depth. The depth measurements are treated as discrete values, and the numerical integration methods are applied to compute the lake's total volume.

Additionally, **MATLAB** will use numerical triple integration to calculate the volume of each small element within the lake based on discrete depth measurements. Furthermore, numerical double integration will be used to compute the area of the lake's boundary from the discrete coordinate data.

The final output will be the lake's average depth, obtained by dividing the total volume (computed using numerical integration) by the lake's total surface area (computed using numerical double integration). The project requires verifying and validating the developed computer program and comparing its results with known datasets or analytical solutions.

Solution algorithm

1. Input:

- (a) The depth measurements and corresponding coordinates (x, y, z) of various points within the lake.
- (b) The coordinates (x, y) of the lake's boundary points for calculating the area.

2. Average Depth Calculation:

- (a) Composite 1/3 Simpson's Rule:

- i. Divide the depth measurements into equal intervals (if the number of data points is odd, the last break will use the 3/8 Simpson's rule).
- ii. For each interval, apply the 1/3 Simpson's rule to compute the average depth.
- iii. Sum up the average depths from all intervals to get the total volume.

- (b) Single 3/8 Simpson's Rule:

- i. If the number of data points is odd, use the 3/8 Simpson's rule for the last interval (which is not considered in the Composite 1/3 Simpson's rule).
- ii. Compute the average depth for this interval using the 3/8 Simpson's rule.
- iii. Add this average depth to the total volume obtained from the Composite 1/3 Simpson's rule.

- (c) Total Volume:

- i. The sum of volumes obtained from the Composite 1/3 Simpson's rule and the Single 3/8 Simpson's rule will give the lakes total volume.

3. Numerical Triple Integration (x, y, z) for the Depth:

- (a) For each depth measurement (x, y, z) , apply numerical triple integration to obtain the volume of the corresponding small element in the lake.
- (b) Sum up the volumes from all the elements to calculate the total volume earlier.

4. Numerical Double Integration (x, y) for the Area:

- (a) Use numerical double integration (x, y) to calculate the area of the lake's boundary. This will give the surface area of the lake.
- (b) If the lake is open (no boundary provided), assume a limit based on the data points' extreme coordinates.

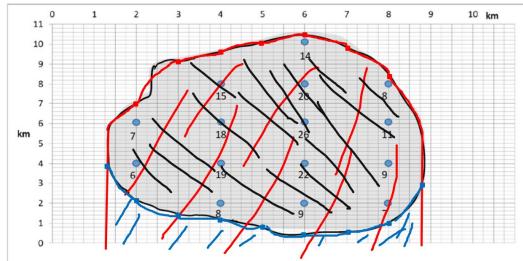
5. Average Depth Calculation and Final Output:

- (a) Divide the total volume obtained from the numerical triple integration by the total surface area obtained from the numerical double integration to calculate the lakes average depth.
- (b) Display the computed average depth as the final output.

6. Error Analysis and Validation:

- (a) Perform error analysis to evaluate the accuracy of the integration methods used.
- (b) Validate the computational tool's results by comparing them with analytical solutions or known datasets for test cases.

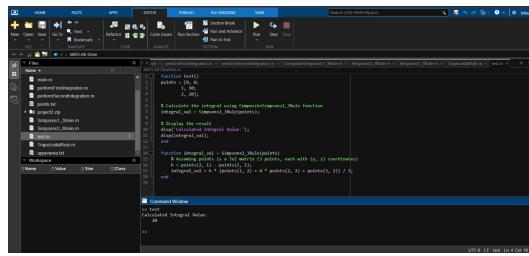
7. Area Calculation:



Validation of the computer program

Firstly, we will test our function for Single Simpson's 1/3 rule in compare with the solved example in the lecture notes:

1. Single Simpson's 1/3 rule function:

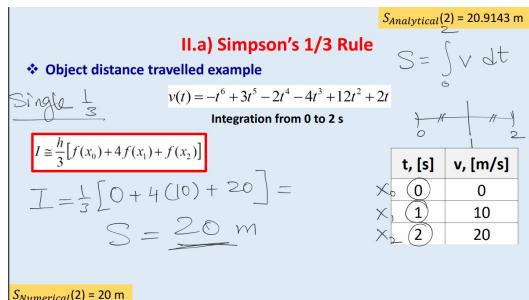


```

function result = simpson_1_3_rule()
    points = [x0, x1, x2, x3];
    v = [v0, v1, v2, v3];
    % calculate the integral value CompositeSimpson_1_3rule function
    integral_val = Simpson_1_3rule(points);
    disp('Calculated Integral Value:');
    disp(integral_val);
end

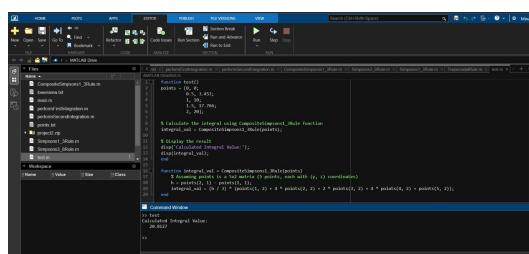
```

2. Solved example from lecture notes:



Next, we will test our function for Composite Simpson's 1/3 rule in compare with the solved example in the lecture notes:

1. Composite Simpson's 1/3 rule function:

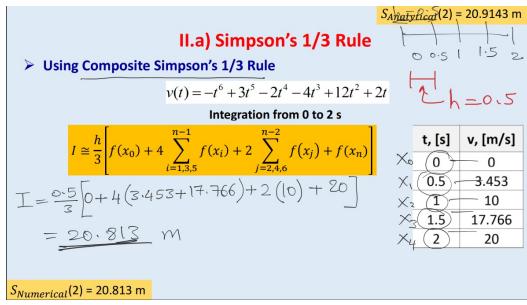


```

function result = composite_simpson_1_3_rule()
    points = [x0, x1, x2, x3, x4, x5];
    v = [v0, v1, v2, v3, v4, v5];
    % calculate the integral value CompositeSimpson_1_3rule function
    integral_val = CompositeSimpson_1_3rule(points);
    disp('Calculated Integral Value:');
    disp(integral_val);
end

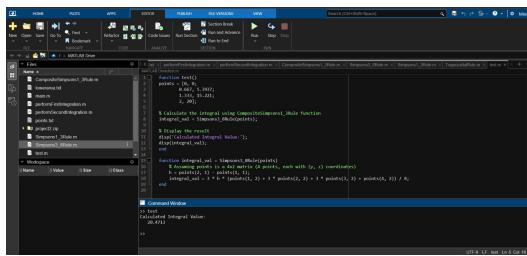
```

2. Solved example from lecture notes:

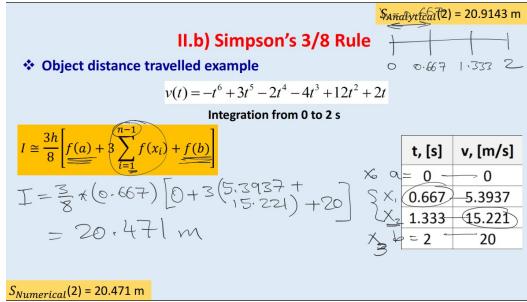


Then, we will test our function for Single Simpson's 3/8 rule in compare with the solved example in the lecture notes:

1. Single Simpson's 3/8 rule function:



2. Solved example from lecture notes:



After that, we will test our function for Single Trapezoidal rule in compare with the solved example in the lecture notes:

1. Single Trapezoidal rule function:

```

function I = trapezoidal_rule(f, a, b, n)
% This function calculates the integral of a function f from a to b using the trapezoidal rule with n subintervals.
% Input: f - function handle, a - lower limit, b - upper limit, n - number of subintervals
% Output: I - calculated integral value

% Calculate the width of each subinterval
h = (b - a) / n;

% Initialize the result
I = 0;

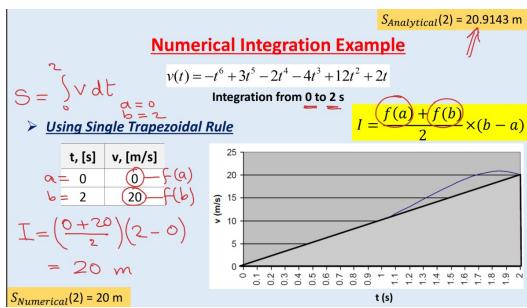
% Loop through the subintervals
for i = 1:n-1
    % Evaluate the function at the endpoints
    x0 = a + i * h;
    x1 = a + (i + 1) * h;
    y0 = feval(f, x0);
    y1 = feval(f, x1);

    % Add the trapezoidal area
    I = I + 0.5 * (y0 + y1) * h;
end

% Add the area under the last segment
I = I + 0.5 * feval(f, b) * h;
end

```

2. Solved example from lecture notes:



After that, we will test our function for Multiple integration in compare with the solved example in the lecture notes:

1. Multiple integration function:

```

function I = multiple_integration(f, a, b, c, d)
% This function performs multiple integration of a function f over the range [a, b] and [c, d].
% Input: f - function handle, a - lower limit for x, b - upper limit for x, c - lower limit for y, d - upper limit for y
% Output: I - calculated integral value

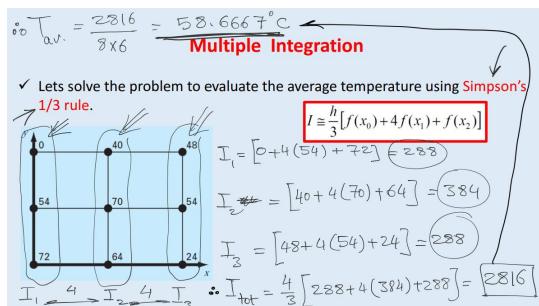
% The first integration
I = quad(f, a, b);

% The second integration
I = quad(I, c, d);

end

```

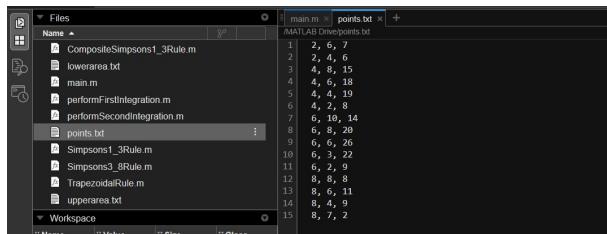
2. Solved example from lecture notes:



User Guide

The project is written in **MATLAB**, allowing you to compile it using **MATHWORKS** online or offline platform. Once the compilation process is complete, the program will be initiated as following:

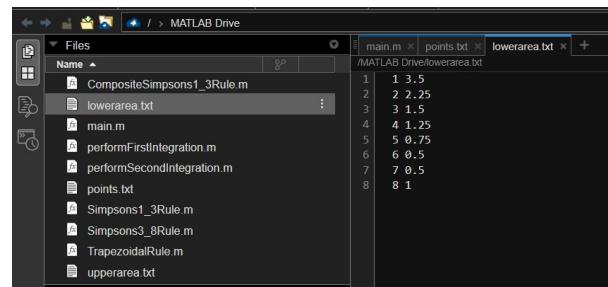
1. Firstly, the user has to write the points in the following format (x, y, z) in a text file as following:



A screenshot of the MATLAB Drive interface. The left sidebar shows files like 'CompositeSimpsons1_3Rule.m', 'lowerarea.txt', 'main.m', 'performFirstIntegration.m', 'performSecondIntegration.m', 'points.txt', 'Simpsons1_3Rule.m', 'Simpsons3_8Rule.m', 'TrapezoidalRule.m', and 'upperarea.txt'. The right pane shows the contents of 'points.txt':

1	2, 6, 7
2	2, 4, 6
3	4, 8, 15
4	6, 18
5	4, 2, 19
6	4, 2, 8
7	6, 10, 14
8	6, 8, 28
9	6, 6, 26
10	6, 3, 22
11	6, 2, 0
12	8, 8, 8
13	8, 6, 11
14	8, 4, 9
15	8, 7, 2

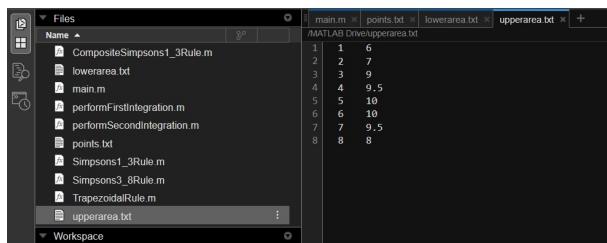
2. Secondly, the user has to write the points in the lower surface of the irregular shape he wants the average in in the following format (x, y) in a text file as following:



A screenshot of the MATLAB Drive interface. The left sidebar shows files like 'CompositeSimpsons1_3Rule.m', 'lowerarea.txt', 'main.m', 'performFirstIntegration.m', 'performSecondIntegration.m', 'points.txt', 'Simpsons1_3Rule.m', 'Simpsons3_8Rule.m', 'TrapezoidalRule.m', and 'upperarea.txt'. The right pane shows the contents of 'lowerarea.txt':

1	3.5
2	2.25
3	1.5
4	1.25
5	0.75
6	0.5
7	0.5
8	1

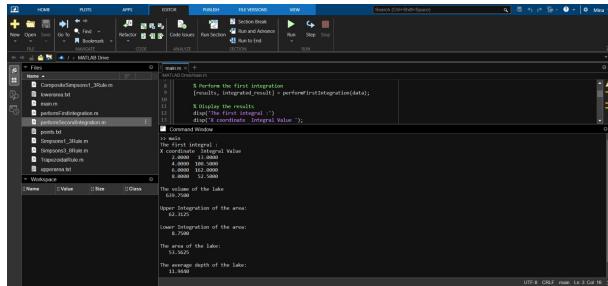
3. Thirdly, the user has to write the points in the upper surface of the irregular shape he wants the average in in the following format (x, y) in a text file as following:



A screenshot of the MATLAB Drive interface. The left sidebar shows files like 'CompositeSimpsons1_3Rule.m', 'lowerarea.txt', 'main.m', 'performFirstIntegration.m', 'performSecondIntegration.m', 'points.txt', 'Simpsons1_3Rule.m', 'Simpsons3_8Rule.m', 'TrapezoidalRule.m', and 'upperarea.txt'. The right pane shows the contents of 'upperarea.txt':

1	1 6
2	2 7
3	3 9
4	4 9.5
5	5 10
6	6 10
7	7 9.5
8	8 8

4. Finally, the user will receive the volume and the area of the lake in addition to the average depth, and then the program will terminate.



A screenshot of the MATLAB IDE. The current file is 'Integration.m'. The code contains a function definition and several comments. The command window shows the output of the code execution, including the volume, area, and average depth of the lake.

```
% This script performs the first integration
% Results: composite Simpson's rule + midpoint rule integration(data);
%
% Display the results
% The first integral:
% Volume of the lake:
% Average depth of the lake:
%
% The second integral:
% Upper Integration of the area:
% Lower Integration of the area:
%
% The area of the lake:
% The average depth of the lake:
```

The volume of the lake
40.7500

Upper Integration of the area:
42.3125

Lower Integration of the area:
41.6875

The area of the lake:
53.5625

The average depth of the lake:
11.8400

Bibliography

- [1] Dr. Mohamed Rashed. (2023). *Lecture Notes*.
- [2] Dr. Mostafa Youssef. (2023). *Lecture Notes*.