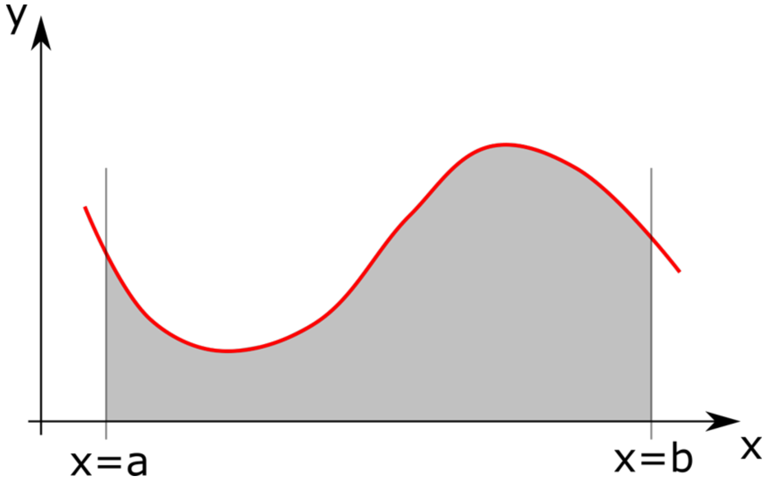
Computer Modelling Techniques

MMME3086 UNUK, 2023/24

Numerical Methods: Numerical integration

Author: Mirco Magnini
Office: Coates B100a
Email: mirco.magnini@nottingham.ac.uk



Contents

1 Introduction

Integrals are employed by engineers to evaluate the total amount or quantity of a given physical variable. The integral may be evaluated over a line, an area, or a volume. For example, let's consider the case of a numerical simulation of a fluid flowing in a pipe, sketched in Fig. 1. At the outlet section of the pipe, the velocity of the fluid is defined at discrete points. The integral of the horizontal velocity component u, evaluated across the outlet area of the pipe, yields the flow rate Q of fluid at the exit:

$$Q = \int_0^H u(y)dy, \tag{1}$$

where H is the distance between the top and bottom walls (in a 2D configuration) of the pipe.

In a one-dimensional case, where $f(x)$ is a function of one variable, integration consists in calculating the integral:

$$I = \int_a^b f(x)dx, \tag{2}$$

The scope of this lecture is to outline the most popular methods to calculate this integral numerically, using a computer. The task of the numerical method is to calculate the integral as accurately as possible, with the smallest number of evaluations of the integrand.

Below, we first outline three different methods to solve integrals of a one-variable function, which are the trapezoidal rule in Section 2, the Simpson's rule in Section 3 and the Gaussian quadrature in Section 4. We introduce very briefly an extension of the Gaussian quadrature to solve multidimensional integrals in Section 5. There is no tutorial planned for this lecture, however some suggestions for potential exercises to be done in Matlab are provided in the final Section 7.

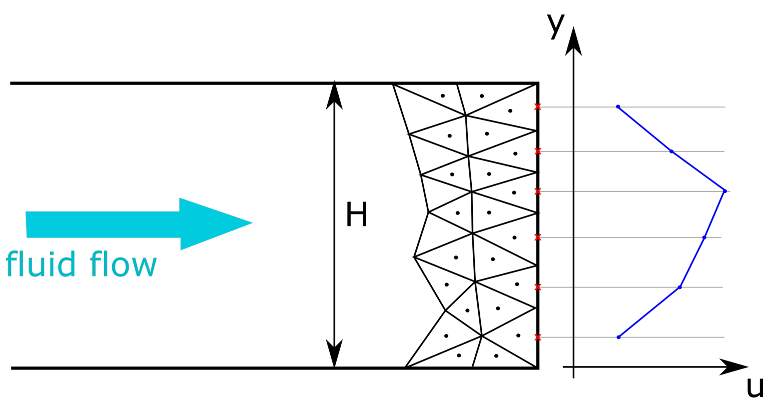


Figure 1: Schematic of the flow of a fluid in a pipe and computational mesh. At the outlet section of the pipe, the velocity of the fluid is defined at discrete points. The integral of the horizontal velocity component evaluated across the outlet area of the pipe yields the flow rate of fluid at the exit.

2 The trapezoidal rule

Our objective is to calculate an approximate solution of the integral displayed in Eq. (2). In order to do this, we perform n function evaluations at the discrete points $x_1, x_2, x_3, \ldots, x_n$, with $x_1 \equiv a$ and $x_n \equiv b$, being equally spaced. With the trapezoidal method, see the schematic in Fig. 2, we divide the integration interval $[a, b]$ into $n-1$ segments, each segment containing two consecutive function evaluations. Within each segment, we approximate the function as a straight line and the approximated integral of the function within the segment is calculated as the area of the trapezium bounded between the straight line and the $x-$axis. Therefore, the approximated integral within one segment is calculated as:

$$I_1 = \int_{x_1}^{x_2} f(x)dx \simeq \frac{1}{2}\left[f(x_1) + f(x_2)\right](x_2 - x_1). \tag{3}$$

Note that, with n function evaluations, we have $n-1$ intervals of width $h \equiv (x_2 - x_1) = (b-a)/(n-1)$. Therefore, the integral in Eq. (2) is approximated by a series of terms:

$$I = \int_{a\equiv x_1}^{b\equiv x_n} f(x)dx \simeq h\left[\frac{1}{2}f(x_1) + f(x_2) + \ldots + f(x_{n-1}) + \frac{1}{2}f(x_n)\right] = h\left[\frac{1}{2}f(x_1) + \sum_{i=2}^{n} f(x_i) + \frac{1}{2}f(x_n)\right]. \tag{4}$$

Note that h does not have to be constant, but can be varied along the interval to follow steep changes of $f(x)$. The accuracy of the method improves as the size of the segments decreases. It can be demonstrated that the numerical error of the integral calculated with the trapezoidal rule is (see Chapra and Canale [1], Sec. 21.1.1):

$$\mathcal{O}\left[\frac{(b-a)^3}{n^2}f''\right],$$

which means that if we double the number of function evaluations, the numerical error reduces by a factor 4; also, the trapezoidal method is exact if $f(x)$ is a linear function, such that $f''(x) = 0$ and

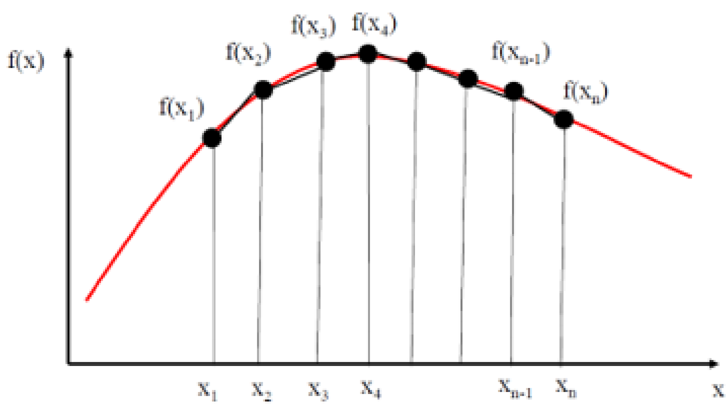


Figure 2: Trapezoidal rule to evaluate the approximate integral under the function $f(x)$.

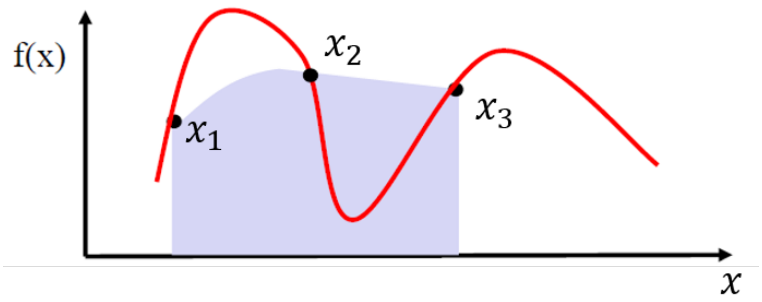


Figure 3: Simpson's rule rule to evaluate the approximate integral under the function $f(x)$.

thus the truncation error is zero.

Further reading: Chapra and Canale [1], Sec. 21.1; Press et al. [2], Sec. 4.1.1.

3 The Simpson's rule

With the Simpson's rule, the curve is now approximated by a parabola evaluated at 3 points, instead of a straight line, see Fig. 3. The approximated integral of the function between three consecutive points is calculated as:

$$I_1 = \int_{x_1}^{x_3} f(x)dx \simeq \frac{h}{3}\left[f(x_1) + 4f(x_2) + f(x_3)\right], \tag{5}$$

where $h = (b-a)/(n-1)$, so that $x_3 - x_1 = 2h$. Therefore, for n function evaluations at n equidistant points, the integral in Eq. (2) is approximated by a series of terms:

$$\begin{aligned} I = \int_{a\equiv x_1}^{b\equiv x_n} f(x)dx &\simeq \frac{h}{3}\left[f(x_1) + 4f(x_2) + 2f(x_3) + 4f(x_4) + \ldots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)\right] = \\ &= \frac{h}{3}\left[f(x_1) + \sum_{\substack{i=2,\\ i:even}}^{n-1} 4f(x_i) + \sum_{\substack{i=3,\\ i:odd}}^{n-2} 2f(x_i) + f(x_n)\right]. \end{aligned} \tag{6}$$

It can be demonstrated that, with the Simpson's rule, the numerical error in the evaluation of the integral is:

$$\mathcal{O}\left[\frac{(b-a)^5}{n^4} f^(iv)\right],$$

and therefore the rule is exact for polynomials up to third-order. The Worked example at the end of this document shows how to implement the trapezoidal and Simpson's rule in Matlab.

Further reading: Chapra and Canale [1], Sec. 21.2; Press et al. [2], Sec. 4.1.1.

4 Gaussian quadrature

We have seen that the numerical calculation of an integral can be generalised as a series of function evaluations:

$$I = \int_a^b f(x)dx \simeq h \sum_{i=1}^{n} f(x_i) w_i, \tag{7}$$

where w_i is the weight coefficient that multiplies the value of the function at a given point x_i. Rather than using fixed points on the curve, the Gaussian quadrature evaluates the function at specific positions (see Fig. 4), so that when the function evaluations are multiplied by carefully chosen weight coefficients, it results in the most accurate evaluation of the integral. The points on the curve are carefully chosen so that the area above the curve "balances" the area below the curve.

With the Gaussian quadrature method, the integral in Eq. (2) is approximated using the formula:

$$I = \int_{-1}^{1} f(x)dx \simeq \sum_{g=1}^{G} f(x_g) w_g, \tag{8}$$

where the range of integration is from -1 to $+1$. If the integral has different limits (e.g. a and b), a linear transformation of the independent variable is required (see section below). The function is evaluated at the gaussian points x_g; the number of gaussian points used, G, coincides with the number of function evaluations, and it is decided by the user; their coordinates x_g, however, must be at specific locations, as indicated in the table in Fig. 5. At these coordinates, $f(x_g)$ is multiplied by a specific weight coefficient w_g (see table) and the products added together to calculate the integral.

To demonstrate how the Gaussian quadrature scheme works, Fig. 6 shows a typical function integrated with 4 Gaussian points. Using Eq. (8) with $G = 4$ yields:

$$I = \int_{-1}^{1} f(x)dx \simeq \sum_{g=1}^{G} f(x_g) w_g = f(x_1)w_1 + f(x_2)w_2 + f(x_3)w_3 + f(x_4)w_4, \tag{9}$$

where G is the total number of Gaussian points, x_g is the "Gaussian coordinate", and w_g is the associated weight function. Note that the choice of the number of points to use G is purely arbitrary,

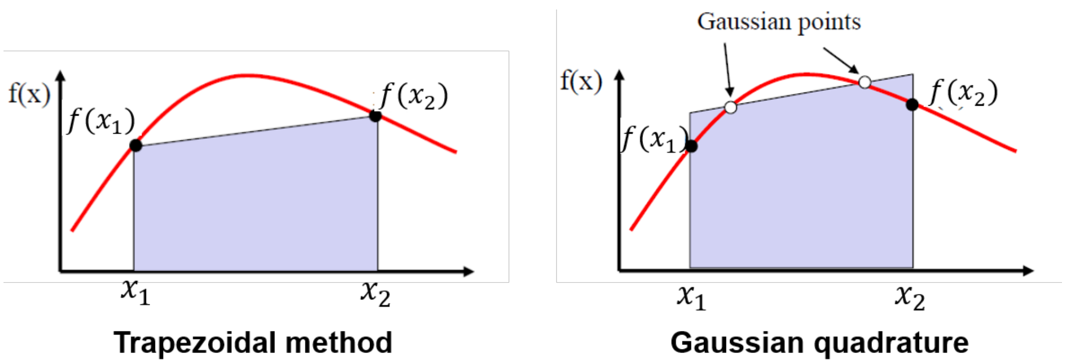


Figure 4: Difference between trapezoidal method and Gaussian quadrature.

although it defines the accuracy of the method: given G gaussian points, the Gaussian quadrature has error $\mathcal{O}(f^{(2G)})$, and therefore it is exact for polynomials of order up to $2G-1$. Accuracy can be improved increasing G, at a higher computational cost owing to the increased number of function evaluations. Therefore, for a fixed number of function evaluations, the Gaussian quadrature is the most accurate integration scheme.

Further reading: Chapra and Canale [1], Sec. 22.4; Press et al. [2], Sec. 4.6.

Example. Use Gaussian quadrature with 4 points to evaluate the integral:

$$I = \int_{-1}^{1} \frac{x}{\sqrt{2x+3}} dx. \tag{10}$$

The analytical solution is:

$$I = \int_{-1}^{1} \frac{x}{\sqrt{2x+3}} dx = \left[\frac{(x-3)\sqrt{2x+3}}{3} \right]_{-1}^{1} = -0.157379. \tag{11}$$

	Gaussian Coordinate	Weight Function
n = 2		
	-0.5773502691896257	1.0
	0.5773502691896257	1.0
n = 3		
	0.0	0.8888888888888888
	-0.7745966692414834	0.5555555555555556
	0.7745966692414834	0.5555555555555556
n = 4		
	-0.3399810435848563	0.6521451548625461
	0.3399810435848563	0.6521451548625461
	-0.8611363115940526	0.3478548451374538
	0.8611363115940526	0.3478548451374538
n = 5		
	0.0	0.5688888888888889
	-0.5384693101056831	0.4786286704993665
	0.5384693101056831	0.4786286704993665
	-0.9061798459386640	0.2369268850561891
	0.9061798459386640	0.2369268850561891

Figure 5: Coordinates of the gaussian points according to the number of function evaluations G, here denoted as n.

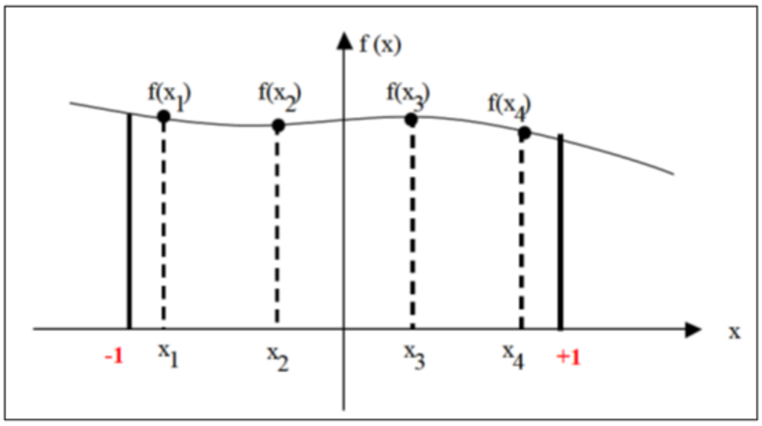


Figure 6: Example of a 4-point Gaussian quadrature scheme.

Using Eq. (9):

$$
\begin{aligned}
I &= \int_{-1}^{1} \frac{x}{\sqrt{2x+3}} dx \simeq f(x_1)w_1 + f(x_2)w_2 + f(x_3)w_3 + f(x_4)w_4 = \\
&= f(-0.8611) \cdot 0.3478 + f(-0.3399) \cdot 0.6521 + f(0.3399) \cdot 0.6521 + f(0.8611) \cdot 0.3478 = \\
&= -0.7618 \cdot 0.3478 + (-0.2231) \cdot 0.6521 + 0.1772 \cdot 0.6521 + 0.3963 \cdot 0.3478 = \\
&= -0.2650 - 0.1455 + 0.1156 + 0.1378 = -0.1571, \qquad (12)
\end{aligned}
$$

which is quite close to the exact solution.

Example. Use Gaussian quadrature with 4 points to evaluate the integral:

$$I = \int_{-1}^{1} \frac{1}{(3x+5)^2} dx. \qquad (13)$$

The analytical solution is:

$$I = \int_{-1}^{1} \frac{1}{(3x+5)^2} dx = \left[\frac{-1}{3(3x+5)} \right]_{-1}^{1} = 0.125. \qquad (14)$$

Using Eq. (9):

$$
\begin{aligned}
I &= \int_{-1}^{1} \frac{1}{(3x+5)^2} \simeq f(x_1)w_1 + f(x_2)w_2 + f(x_3)w_3 + f(x_4)w_4 = \\
&= f(-0.8611) \cdot 0.3478 + f(-0.3399) \cdot 0.6521 + f(0.3399) \cdot 0.6521 + f(0.8611) \cdot 0.3478 = \\
&= 0.1712 \cdot 0.3478 + 0.0631 \cdot 0.6521 + 0.0276 \cdot 0.6521 + 0.0174 \cdot 0.3478 = \\
&= 0.0595 + 0.0411 + 0.0180 + 0.0061 = 0.1247, \qquad (15)
\end{aligned}
$$

which is quite close to the exact solution.

Change of integration limits from -1 to $+1$

Let's say that we want to solve the integral:

$$I = \int_a^b f(x)dx, \tag{16}$$

using Gaussian quadrature. However, the integration limits in Gaussian quadrature are -1 and $+1$. To be able to use Eq. (8), we need to do a change of variables. Let's define a new variable $\widetilde{x}$, such that:

$$x = A_1 + A_2\widetilde{x}. \tag{17}$$

A_1 and A_2 can be found by requiring that when $x = a$, $\widetilde{x} = -1$ and when $x = b$, $\widetilde{x} = 1$, so that:

$$A_1 = \frac{b+a}{2}, \quad A_2 = \frac{b-a}{2} \Rightarrow x = \frac{b+a}{2} + \frac{b-a}{2}\widetilde{x}, \tag{18}$$

from which it follows that:

$$dx = \frac{b-a}{2}d\widetilde{x}. \tag{19}$$

Therefore, we can rewrite the integral as a function of $\widetilde{x}$:

$$I = \int_a^b f(x)dx = \frac{b-a}{2}\int_{-1}^1 f\left(\frac{b+a}{2} + \frac{b-a}{2}\widetilde{x}\right)d\widetilde{x}, \tag{20}$$

which is now bounded by -1 and $+1$, and use Gaussian quadrature to solve it.

5 Gaussian quadrature for multidimensional integrals

Gaussian quadrature can be easily extended to evaluate integrals in 2D or 3D by employing nested summations, for instance in 2D:

$$\int_{-1}^1 \left(\int_{-1}^1 f(x,y)dx\right)dy \simeq \sum_{g2=1}^{G_2}\left(\sum_{g1=1}^{G_1} f(x_{g1}, y_{g2})\, w_{g1}\right) w_{g2}, \tag{21}$$

where the number of points G_1 and G_2 for each summation loop may be different. Similarly, the scheme can be extended to functions of three variables.

Further reading: Press et al. [2], Sec. 4.8.

6 Worked example

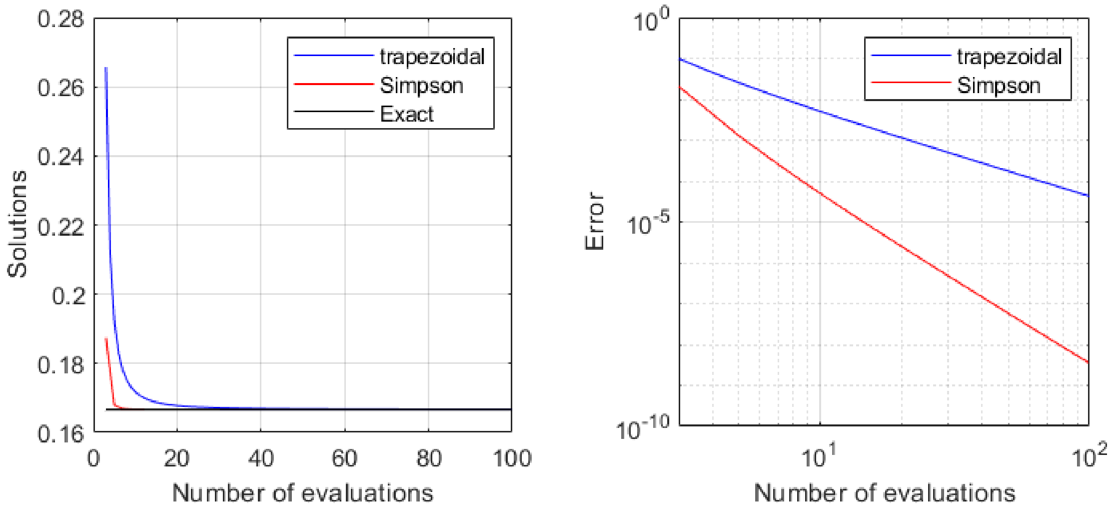
We want to evaluate the following integral:

$$\int_0^1 x^5 dx = \frac{1}{6}, \tag{22}$$

using both the trapezoidal and Simpson's rule. Also, we want to use a variable number of function evaluations and demonstrate the order of convergence of each method. The code to do this is below:

```
1  %%%%% Computational Modelling Techniques - Part 1: Numerical Methods
2  %%%%% Lecture 4, Worked Example 1 - Numerical integration of 1D function
3
4  clear all; close all; clc
5
6  a=0; b=1; % Integral limits
7  Nt=3:100; % Number of points for trapezoidal method
8  Ns=3:2:99; % Number of points for Simpson's. Must be an odd number!
9
10 %%% Function y(x)=x^5 --> Int(x^5 dx)_0^1=1/6
11 I=1/6; % Analytical solution
12
13 for i=1:numel(Nt) % Integral with trapezoidal method
14     x=linspace(a,b,Nt(i)); y=x.^5;
15     h=(b-a)/(Nt(i)-1);
16     It(i)=h*(0.5*y(1)+sum(y(2:end-1))+0.5*y(end));
17 end
18
19 for i=1:numel(Ns) % Integral with Simpson's
20     x=linspace(a,b,Ns(i)); y=x.^5;
21     h=(b-a)/(Ns(i)-1);
22     Is(i)=h/3*(y(1)+sum(4*y(2:2:end-1))+sum(2*y(3:2:end-1))+y(end));
23 end
24
25 figure('color','w','units','Centimeters','position',[5 5 7.5 7]);
26 plot(Nt,It,'b'); hold on; plot(Ns,Is,'r'); plot([Nt(1) Nt(end)],[I I],'k-')
27 box on; grid on; xlabel('Number of evaluations'); ylabel('Solutions');
28 legend('trapezoidal','Simpson','Exact')
29
30 figure('color','w','units','Centimeters','position',[5 5 7.5 7]);
31 errt=abs(It-I); errs=abs(Is-I); % Error
32 loglog(Nt,errt,'b'); hold on; loglog(Ns,errs,'r')
33 box on; grid on; xlabel('Number of evaluations'); ylabel('Error');
34 legend('trapezoidal','Simpson')
35
36 P=polyfit(log10(Nt),log10(errt),1); -P(1) % Order of trapezoidal
37 P=polyfit(log10(Ns),log10(errs),1); -P(1) % Order of Simpson's
```

where we choose $n = 3, 4, 5, ..., 100$ for the trapezoidal method and $n = 3, 5, 7, ..., 99$ for the Simpson's method, as the latter wants an odd number of function evaluations. Lines 16 and 22 implement Eqs. (4) and (6), respectively. The error is calculated as deviation from the analytical solution, and the last two lines calculate the slop of the error curve. Figure 7 plots the solutions obtained with the two methods for increasing number of function evaluations and the error with respect to the analytical solution in loglog coordinates. The slope of the error curves becomes rather constant at larger n. The slopes, calculated based on the last 10 points of each curve, is 2.02 for the trapezoidal method and 4.05 for the Simpson's rule, thus confirming the convergence orders of the truncation errors of, respectively, $\mathcal{O}(1/n^2)$ and $\mathcal{O}(1/n^4)$.



(a) (b)

Figure 7: (a) Convergence of the numerical integration of $f(x) = x^5$ between 0 and 1 using trapezoidal and Simpson's rules and (b) related errors.

7 Suggested exercises

There is no tutorial planned for this lecture. However, you can try to implement the trapezoidal rule, the Simpson's method as well as the Gaussian quadrature method in Matlab to solve the two examples in Section 4. For the trapezoidal rule and the Simpson's method, you can try different numbers of intervals and see how the error changes depending on h. As a benchmark for your implementation, you can use Matlab's built-in numerical integration methods: `trapz` integrates a function using the trapezoidal method (type `doc trapz` in the command window); the function `integral` performs integration using global adaptive quadrature and default error tolerances.

References

[1] S. C. Chapra and R. P. Canale. *Numerical Methods for Engineers, 7th edition.* McGraw-Hill Education, New York, USA, 2015. NUsearch; Download(may not work).

[2] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes.* Cambridge University Press, Cambridge, England, 2007. NUsearch; Download(may not work).