University of Nottingham
UK | CHINA | MALAYSIA

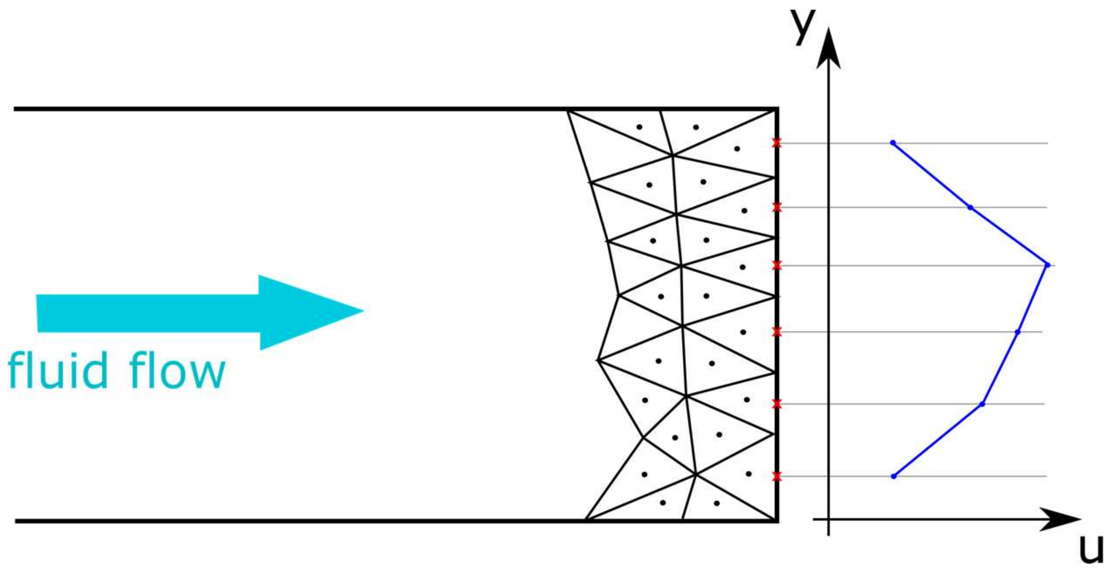
Computer Modelling Techniques

Numerical Methods
Lecture 5: Numerical Integration

Mirco Magnini

Why do we need numerical integration?

For instance, we want to know the volumetric flow rate Q exiting a pipe:



$$Q = \int_{height} u(y)dy$$

The numerical evaluation of the integral has to be **accurate** and **fast**

Most popular methods:

- Trapezoidal rule
- Simpson's rule
- Gaussian quadrature

Today's menu

- Trapezoidal rule
- Simpson's rule
- Gaussian quadrature

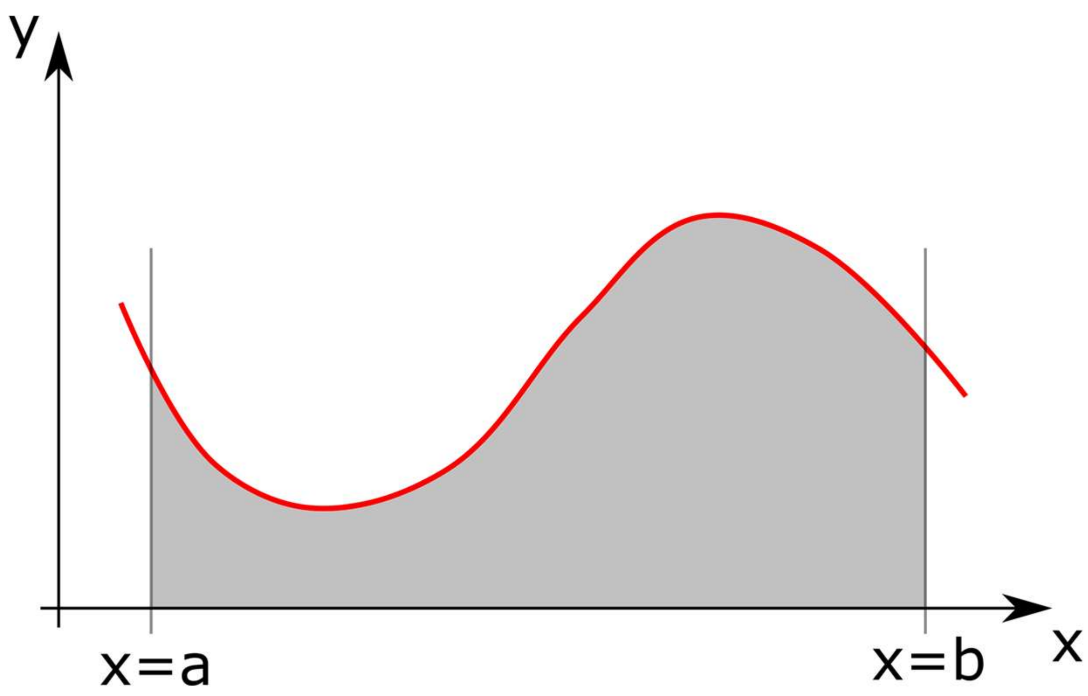
<u>Expected outcome</u>: know how to perform numerical integration of functions; know advantages/limitations of each method; know how to implement each method.

Numerical integration

Our task:

$$I = \int_a^b f(x)dx$$

Calculate the integral as accurately as possible with the smallest number of evaluations of the integrand



Trapezoidal rule

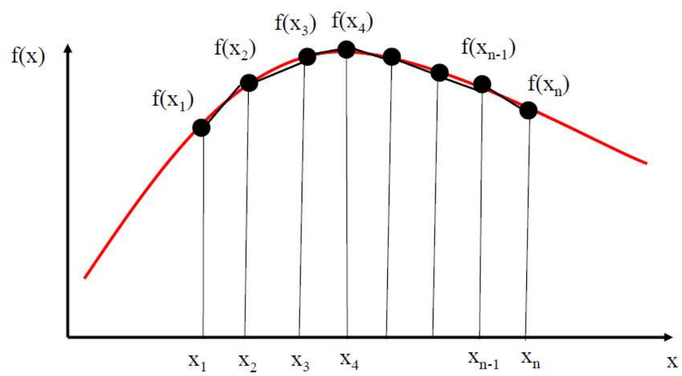
We perform n function evaluations at discrete points $x_1 \equiv a, x_2, x_3, \dots, x_n \equiv b$. We divide [a,b] into $n-1$ segments, each segment containing two consecutive function evaluations. Within each segment, we approximate the function as a straight line. The integral is calculated within each segment as the area of the trapezium bounded between the straight line and the x-axis:

$$I_1 = \int_{x_1}^{x_2} f(x)dx \cong \frac{1}{2}[f(x_1) + f(x_2)](x_2 - x_1)$$

With n function evaluations, we have $n-1$ segments of width $h \equiv (x_2 - x_1) = (b-a)/(n-1)$.

The integral becomes a series of terms:

$$I = \int_{a=x_1}^{b=x_n} f(x)dx \cong h\left[\frac{1}{2}f(x_1) + f(x_2) + \cdots + f(x_{n-1}) + \frac{1}{2}f(x_n)\right] =$$



$$= h\left[\frac{1}{2}f(x_1) + \sum_{i=2}^{n-1} f(x_i) + \frac{1}{2}f(x_n)\right]$$

It can be demonstrated that the error is $O\left[\frac{(b-a)^3}{n^2}f''\right]$

Simpson's rule

The curve is now approximated by a **parabola** evaluated at **3 points**, instead of a straight line.

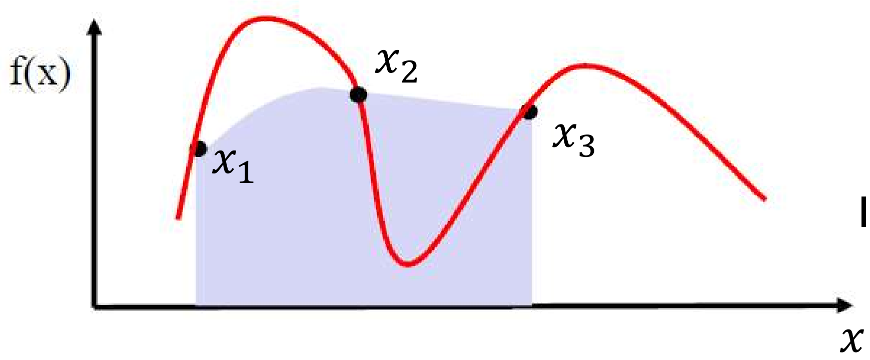
The integral of the function between three consecutive points is:

$$I_1 = \int_{x_1}^{x_3} f(x)dx \cong \frac{h}{3}[f(x_1) + 4f(x_2) + f(x_3)]$$

where $h \equiv (b-a)/(n-1)$. With n function evaluations, the integral becomes a series of terms:

$$I = \int_{a=x_1}^{b=x_n} f(x)dx \cong \frac{h}{3}[f(x_1) + 4f(x_2) + 2f(x_3) + 4f(x_4) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)] =$$

$$= \frac{h}{3}\left[f(x_1) + \sum_{\substack{i=2,\\ i:even}}^{n-1} 4f(x_i) + \sum_{\substack{i=3,\\ i:odd}}^{n-2} 2f(x_i) + f(x_n)\right]$$



It can be demonstrated that the error is $O\left[\frac{(b-a)^5}{n^4} f^{(iv)}\right]$

Gaussian quadrature

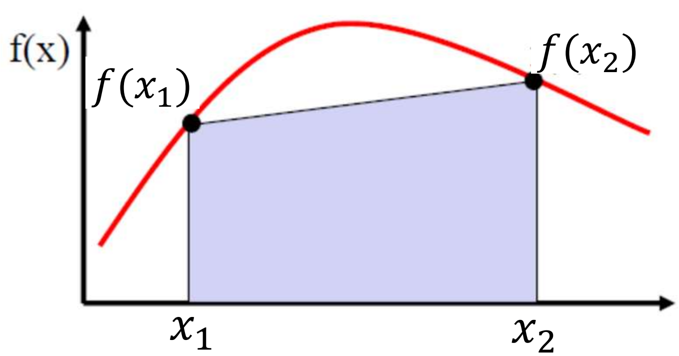
We have seen that the numerical calculation of an integral can be generalised as a **series** of **function evaluations**:

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

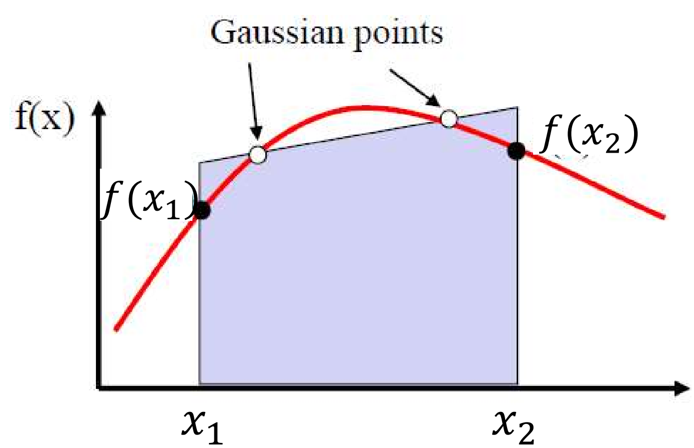
where w_i is the weight coefficient that multiplies the value of the function at a given point x_i.

Rather than using fixed points on the curve, the **Gaussian quadrature** evaluates the function at **specific positions,** 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**.



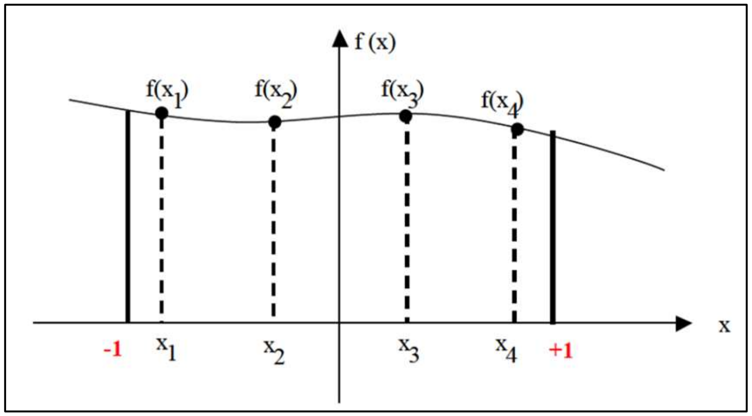
Trapezoidal method



Gaussian quadrature

Gaussian quadrature

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



Example: 4-points Gaussian quadrature scheme

- The **range of integration** is from -1 to +1. If the integral has different limits, a linear transformation of the independent variable is required (see notes).

- The function is evaluated at the **gaussian points** x_g. At these coordinates, $f(x_g)$ is multiplied by a specific **weight coefficient** w_g and the products added together to calculate the integral.

- For a fixed number of function evaluations, the Gaussian quadrature is the **most accurate** integration scheme. Given G gaussian points, the Gaussian quadrature has error $O\left(f^{(2G)}\right)$, and therefore it is exact for polynomials of order up to $(2G-1)$.

- **Accuracy** can be improved increasing G, at a higher cost of computational time.

Gaussian quadrature

Gaussian points coordinates and related weights for G up to 5

	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

Gaussian quadrature - Example

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

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

Example: 4-points Gaussian quadrature scheme

Gaussian Coordinate x_g	Weight Function w_g	$f(x) = 1 / (3x+5)^2$		$f(x) = x / (2x+3)^{0.5}$	
		$f(x_g)$	$f(x_g)\,w_g$	$f(x_g)$	$f(x_g)\,w_g$
-0.8611363	0.3478548	0.1712354	0.0595651	-0.7618207	-0.2650030
-0.3399810	0.6521452	0.0631279	0.0411686	-0.2232066	-0.1455631
0.8611363	0.3478548	0.0173889	0.0060488	0.3962747	0.1378461
0.3399810	0.6521452	0.0275940	0.0179953	0.1772283	0.1155786
		$\Sigma f(x_g)\,w_g = 0.1247778$		$\Sigma f(x_g)\,w_g = -0.1571415$	

$$I = \int_{-1}^{1} f(x)dx = f(x_1)w_1 + f(x_2)w_2 + f(x_3)w_3 + f(x_4)w_4 =$$

$$= f(-0.86111363)(0.3478548) + f(-0.3399810)(0.6521452) + f(0.8611363)(0.3478548) + f(0.3399810)(0.6521452)$$

Gaussian quadrature – Multidimensional integrals

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

$$I = \int_{-1}^{+1} \left(\int_{-1}^{+1} f(x,y)dx \right) dy \cong \sum_{g2=1}^{G2} \left(\sum_{g1=1}^{G1} f\left(x_{g1}, y_{g2}\right) w_{g1} \right) w_{g2}$$

where the number of points $G1$ and $G2$ for each summation loop may be different.

Similarly, the scheme can be extended to functions of 3 variables.

What to take home from today's lecture

- Working principles of trapezoidal, Simpson's and Gaussian quadrature methods
- Discretised version of the integral based on each method
- Order of convergence of each method and its implications
- Advantages/limitations of each method