Aschaffenburg University of Applied Sciences
Faculty for Engineering
Prof. Dr. Michael Möckel

Numerical Methods and Simulation
Module MT 03
Winter term 2022/23

Email: Michael.Moeckel@th-ab.de
Telephone: +49 (0)6021 / 4206 - 507

`https://moodle.th-ab.de/course/view.php?id=624`

Problem Set 9

Numerical integration (Quadrature)

Coursework

Exercise 1: From interpolation to numerical integration

Numerical integration, also known as *quadrature*, is about integrating a function f which is given not by an analytical expression but by a *finite set of numbers* $(x_i, f(x_i)$ with $1 \leq i \leq n$. Therefore, many interpolation strategies can be extended to methods for numerical integration.

(0) **General quadrature.** In general, the integral of a function f which is only known at some sample locations $x_i \in \mathcal{D}$ is given as the weighted sum

$$I[f;\mathcal{D}] \equiv \sum_{x_i \in \mathcal{D}} w_i f(x_i)$$

w_i are suitable *weight factors*. These weight factors can be specified by a *quadrature rule* which assumes a certain interpolation of the data points for calculating the integral. If piecewise polynomials of a given order are used for this interpolation the *Newton - Cotes* quadrature formulas can be obtained:

(1) The simplest quadrature rules are those for standard **Riemann integration**, i.e. the decomposition of the integral into a set of rectangles. This corresponds to a fitting of piecewise constant functions (resulting in a non-continuous function) to the data points. A constant function is a polynomial of order 0. Write down the weight factors w_i if $f(x_i)$ represents the

- (i) left hand point of a suitable rectangle
- (ii) right hand point of a suitable rectangle
- (iii) midpoint of a suitable rectangle

(2) A linear interpolation between two function values $f(x_i)$ and $f(x_{i+1})$ leads to the **trapezoidal rule**. What are the corresponding weight factors w_i?

(3) Implement the above quadrature rules in a MATLAB routine which receives two sets of points $x = [x_i]$ and $y = [f(x_i)]$.

(4) Run the routine for the function $f(x) = x^2$ with 10 values $x_i \in [0, 2]$.

(5) Compare with the built-in MATLAB routine `trapz(x,y)`.

Exercise 2: Simpson's rule

The most famous and most popular among the Newton-Cotes quadrature rules is known as **Simpson's rule** (despite the fact that it was earlier used by Kepler). It assumes piecewise fitting of quadratic polynomials to sections of three neighbouring locations x_i, x_{i+1} and x_{i+2}. Then the weights of the quadrature formula $q_2[f] = w_0 f(x_i) + w_1 f(x_{i+1}) + w_2 f(x_{i+2})$ are determined such that the exact integral is reproduced for three simple polynomial functions:

1.) $f_1(x) = 1$, such that $\int_{x_i}^{x_{i+2}} f_1(x)dx = x_{i+2} - x_i \stackrel{!}{=} w_0 + w_1 + w_2$

2.) $f_2(x) = x$, such that $\int_{x_i}^{x_{i+2}} f_2(x)dx = 1/2(x_{i+2}^2 - x_i^2) \stackrel{!}{=} w_0 x_i + w_1 x_{i+1} + w_2 x_{i+2}$

3.) $f_3(x) = x^2$, such that $\int_{x_i}^{x_{i+2}} f_3(x)dx = 1/3(x_{i+2}^3 - x_i^3) \stackrel{!}{=} w_0 x_i^2 + w_1 x_{i+1}^2 + w_2 x_{i+2}^2$

(1) If the quadrature rule is exact for these three polynomials it is also correct for any interpolating polynomial of the form $g(x) = a_2 x^2 + a_1 x + a_0$. Why?

(2) Re-write the conditions in (3) for equidistant locations with $h = x_{i+1} - x_i$ and calculate the weight factors w_i for a single section of the integral.

(3) For evaluating an integral at many locations x_i various sections need to be glued together. Then points at the boundaries contribute to two sections. Rewrite the full quadrature formula for N sections.

(4) Implement the Simpson rule in a MATLAB routine which receives two sets of points $x = [x_i]$ and $y = [f(x_i)]$.

Exercise 3: Recycling in numerics - iterative trapezoidal rule

If a function is known analytically its integral can be calculated numerically by using different numbers of locations x_i and function calls $f(x_i)$. Increasing the resolution of the independent variable x_i increases the precision of the integral but also requires increased numerical effort. To keep the additional effort as small as possible it is advisable to recycle previously calculated results and to improve a result until a certain level of precision has been reached.
Consider a scheme where the number of equidistant locations $x_i \in [a, b]$ is doubled in every iteration n. For $n = 0$ the trapezoidal rule results in an integral $I_0[f; a, b] = (a - b)\ 1/2(f(a) + f(b))$. With every iteration the basis length $h = x_{i+1} - x_i$ is halved and more inner points are added to the integral.

(1) Consider on paper the first two iterations and write down the contributions to the integrals I_1 and I_2. How is I_1 contained in I_2?

(2) Write down a general scheme for an arbitrary number of iterations. Consider (i) the number of inner points `ip`, (ii) the current basis length h_i and (iii) how the previous integral I_{i-1} contributes to the next iteration I_i.

(3) Program a MATLAB function `TrapIteration(f, a,b,n)` which accepts as parameters an anonymous function f, an interval given by a and b and the number of iterations n. It should loop over all iterations from `i=1:n` and print the current number of the iteration as well as the current value of the integral.

(4) Run the MATLAB routine for the function $f(x) = x^2$ in the interval set by $a = 0$ and $b = 1$. What is the expected result for the value of the integral? Start with n = 10, 15, 20 iterations and carefully increase n to slightly larger values. What do you observe? Note: You can interrupt MATLAB with the keys Ctrl + C.

Exercise 4: MATLAB commands for quadrature

MATLAB provides a couple of routines for integrating a function f numerically over the interval $[a, b]$ including

- `trapz(x,y)`
- `quad(f,a,b)` which implements the Simpson rule. Note that this and all similar commands called `quad...` will be removed from a future MATLAB release, so use instead
- `integral(f, a, b)` which uses an adaptive choice of intermediate points

Calculate the integrals for the following functions numerically

(1) $\int_0^\pi \sin(x) dx$

(2) $\int_0^{\pi/2} \sqrt{1 + \cos^2(x)} dx$

(3) $\int_0^1 e^{-u^2} du$

(4) Plot the integral of the Gauss error function $\int_0^x e^{-u^2} du$ in the interval $[0, 5]$.

(5) Calculate the integral of the function $f(x) = x^3 - 30x + 30$ between its two smallest roots