computational mathematics

Prof. Dr. S. Sauter
Institut für Mathematik
Universität Zürich

Numerik I – Homework 6

Deadline: 5.4.2019 12:00

Exercice 1 (7 P.)(Theoretical task)
Let $I = [a, b]$. We want to approximate

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

for a function $f \in C^0(I)$.

(a) Derive a quadrature formula $Q_M(f)$ based on the constant interpolation in the midpoint $x_M = \frac{a+b}{2}$.

(b) Derive a quadrature formula $Q_H(f)$ based on the linear Hermite Interpolation in the midpoint $x_M = \frac{a+b}{2}$, i.e. a linear interpolation $p(f)$ of f that satisfies

$$p(f)(x_M) = f(x_M), \qquad p(f)'(x_M) = f'(x_M).$$

(c) Derive for the quadrature formula in (a) an estimate of the error

$$|\mathcal{I}(f) - Q_M(f)|.$$

Hint: Use (b) and proceed as in the error estimate of the Simpson's rule in the script.

(d) Let $a = x_0 < x_1 < \ldots < x_n = b$ be an equidistant partition of the interval I. Define by Q_M^n the composed quadrature formula of Q_M: in each subinterval $[x_{i-1}, x_i]$, $i = 1, \ldots, n$ apply the quadrature Q_M. Derive an estimate of the error

$$|\mathcal{I}(f) - Q_M^n(f)|.$$

Exercise 2 (9 P.) (Computational task)
Let $I := [0, 1]$ and divide I into n subintervals $\cup_{i=1}^n [x_{2i-2}, x_{2i}]$ (not necessarily equidistant). We want to approximate the integral of a given function $f : I \to \mathbb{R}$

$$\mathcal{I}(f) := \int_0^1 f(x)dx$$

using a composed quadrature rule $Q_{\tilde{S}}^n$, based on the 3-point interpolation on $\{x_{2i-2}, x_{2i-1}, x_{2i}\}$, $1 \le i \le n$.

(a) Code the composed quadrature rule described above.
Input: vector of nodes $(x_i)_{i=0}^{2n}$, vector $(f_i)_{i=0}^{2n}$ where $f_i = f(x_i)$.
Output: $Q_{\tilde{S}}^n(f)$.
Note that n subintervals correspond to $2n + 1$ points.
Remark: If $x_{2i-1} = \frac{x_{2i-2}+x_{2i}}{2}$, this is the composed Simpson's rule.

(b) Test the program with the function $f(x) := \pi x \sin(\pi x^2)$ for n equidistant subintervals and $x_{2i-1} = \frac{x_{2i-2}+x_{2i}}{2}$ (i.e. composed Simpson's rule). Compute analytically $\mathcal{I}(f)$. Store in a table the values $e_n(f) := |Q_S^n(f) - \mathcal{I}(f)|$ for $n = 2^i$, $1 \le i \le 10$. Plot the logarithm of the error with respect to $\log(n)$. What can you say about the convergence rate?

(c) From the lecture we know that the error of the composed Simpson's quadrature for $f \in \mathcal{C}^\infty(I)$ can be estimated by:

$$e_n(f) = |\mathcal{I}(f) - Q_S^n(f)| \le Ch^4,$$

where $h = \dfrac{1}{n}$ is the length of each subinterval. Using the plot/data produced in (b), estimate the constant C.

(d) Code the composed trapezoidal rule Q_T^m, which for each subinterval $[x_{j-1}, x_j]$ for $j = 1, \ldots, m$ is defined by

$$\int_{x_{j-1}}^{x_j} f(x)dx \approx \frac{x_j - x_{j-1}}{2} \left(f(x_{j-1}) + f(x_j)\right),$$

Input: $(x_i)_{i=0}^m$ and $(f_i)_{i=0}^m$, where $f_i = f(x_i)$.
Output: $Q_T^m(f)$.

(e) Compare the error and the order of convergence produced by the two methods of (b) and (c) with respect to same number of function evaluations (i.e. same number of nodes).

(f) Test your program for the same function $f(x) := \pi x \sin(\pi x^2)$ with nodes

$$x_i = \begin{cases} \frac{i}{2n} & \text{for } i \text{ even,} \\ \frac{i+\delta_i}{2n} & \text{for } i \text{ odd.} \end{cases}$$

for $|\delta_i| < \frac{1}{2}\delta$, $\delta = 1$ and different values of n. Plot the logarithm of the error with respect to $\log(n)$. For δ_i use a random number (e.g. with `rand` in Matlab). What is the convergence rate of the method?
Repeat the test for smaller values of $\delta \to 0$ and comment on your observations.