Numerical Methods I – Lab 7

Fall Semester 2005

November 22, 2005

1. Write an Octave function for computing the weights w_i in the Newton–Cotes quadrature formula on $[0, 1]$, where

$$\int_0^1 f(x)\,dx \approx \sum_{i=0}^{n} w_i\,f(i/n)\,.$$

   Use the fact that Newton–Cotes quadrature is exact at least for all polynomials of degree $\leq n$. This yields $n+1$ linearly independent equations for the $n+1$ unknown weights w_i which can be solved numerically; you are allowed to use the built-in Octave linear algebra routines. Compute the result up to $n = 10$. For which n do negative weights occur?

   Optional. The weights are rational numbers. Write a program for computing the rational expressions of the w_i. You may use a symbolic software package.

2. The computations in Part 1 might become problematic when $n > 10$ depending on the method used for solving the linear system $Aw = b$ for the weights. Using `w=inv(A)*b` or LU-factorization, you can document this by checking the error of the quadrature rule for the monomials $f(x) = x^m$, $m = 0, \ldots n$; if you have used these $n+1$ linearly independent polynomials to derive the linear system in Part 1, then this is equivalent to computing the residual $b - Aw$. What is the reason for the observed instability? How can you improve the result? Think of what you have learned about solving linear systems.

   Note: There seems to be a lesser problem with using `w=A\b`. What could be the reason?

(Credits to P.O.)