MATH 651: Numerical Analysis II

Instructor: Prof. Wolfgang Bangerth
bangerth@colostate.edu

Homework assignment 3 – due 10/11/2019

Problem 1 (Different methods for numerical integration). Let's go back to the function $f(x) = \sin\frac{1}{x}$ and assume that we want to compute the integral $J = \int_{0.05}^{1} f(x)\ dx$. The value of the integral is about 0.50283962, but you should find a numerical approximation to it.

Using a subdivision of the interval $[0.05, 1]$ into K equally sized intervals I_k, approximate the integral above using the

(i) midpoint rule,

(ii) trapezoidal rule,

(iii) Simpson rule,

(iv) 2-point Gauss rule,

(v) 3-point Gauss rule.

For each of these methods, generate a plot that shows the error between the exact integral value and your numerical approximation as a function of the number of function evaluations you need. (The number of function evaluations will be K, $K+1$, $2K+1$, $2K$, and $3K$ for the methods above.)

For each of the methods, also state how many function evaluations you need to achieve an accuracy of 10^{-6}. **(40 points)**

Problem 2 (Romberg integration Use the Romberg procedure with the trapezoidal rule to obtain more accurate results from function evaluations than the trapezoidal rule would yield by itself. Again show a plot that depicts the error for the numerical approximation of the same integral as in the previous problem, as a function of function evaluations. **(20 points)**

Problem 3 (Adaptive numerical integration In the derivation of the convergence rates for the various methods, always came to a point where we could state the error as

$$|e| \le \sum_{k=1}^{K} \frac{1}{p!} \|f^{(p)}\|_{\infty,I_k} h_k^{p+1}.$$

Here, p is the convergence order of the respective method. (Remember that we lose one power of h in the summation, to obtain $|e| = \mathcal{O}(h^p)$ if the h_k are all roughly of the same size.)

This representation of the total suggests a strategy like the one we explored for interpolation: To choose intervals so that h_k is only small if $f^{(p)}$ is large. To this end, we should start with a relatively coarse mesh and then for each interval compute some sort of criterion η_k that indicates how large the error is – for example

$$\eta_k = \frac{1}{p!} \left| f^{(p)}\left(\frac{x_k - x_{k-1}}{2}\right) \right| h_k^{p+1},$$

where (i) we assume that we can evaluate *derivatives* of f exactly, and (ii) we approximate the ∞-norm of the derivative by just evaluating it at the midpoint of the interval. We would then "refine" the interval with the largest η_k by replacing it with its two halves, recompute our current approximation of the interval, and repeat the process.

Try this strategy with the trapezoidal rule for the integral of Problem 1. This integral should be well suited for this kind of adaptive procedure since the function's smoothness varies substantially in the interval $[0.05, 1]$. In each iteration of the algorithm, evaluate the error between the numerical approximation of the integral and the exact value. Produce a plot that shows the error as a function of the number of function evaluations you need and compare the results with the data obtained in part (a).

How many function evaluations do you now need to achieve an accuracy of 10^{-6}? **(40 points)**