Program 4 Foundations of Computational Math 1 Fall 2020

Due date: 11:59PM on November 23, 2020

Programming Exercise

Overview

In this assignment you will implement two numerical quadrature methods based on composite methods with global mesh refinement for error estimation and termination. The notes contain all of the required expressions for one of the methods – Composite Trapezoidal Rule – but you must derive some expressions as part of the assignment for the other one – Composite Midpoint Rule. You will compare each method's predicted behavior to observed behavior and compare the methods efficiency (accuracy vs function evaluations), i.e., what takes least work to achieve a required accuracy.

All methods will assume uniform interval size, $H_m = (b-a)/m$, and will be written in terms of H_m as will their error expressions rather than their local step size h_i, in order to facilitate comparisons.

Composite Trapezoidal Rule

For the Composite Trapezoidal Rule all of the expressions needed are given in the notes. The basic for is given by

$$I_m^{ctr} = \frac{H_m}{2}\left[f_0 + f_m + 2\sum_{i=1}^{m-1} f_i\right].$$

where $0 \le i \le m$, $x_i = a + iH_m$ and $f_j = f(x_j)$.

Assuming $\alpha = 1/2$ there is complete reuse of previous function evaluations to generate the fine grid ($2m$ intervals) quadrature, I_{2m}, from the coarse grid (m intervals) quadrature, I_m^{ctr}:

$$I_{2m}^{ctr} = \frac{1}{2}\left[I_m^{ctr} + H_m \sum_{\text{m new pts}} f_i\right]$$

where the new points are the midpoints of the m coarse grid intervals.

The composite error expression for m intervals is

$$E_m^{ctr} = I(f) - I_m^{ctr}(f) = -(b-a)\frac{H_m^2}{12}f'' + O(H_m^3)$$

and the error estimate from the coarse/fine combination, $\alpha = 1/2$, is easily seen to be from the expression in the class notes of the form:

$$E_m^{ctr} \approx \frac{2^r}{(2^r - 1)}(I_{2m}^{ctr} - I_m^{ctr})$$

$$E_{2m}^{ctr} \approx \frac{1}{(2^r - 1)}(I_{2m}^{ctr} - I_m^{ctr})$$

where $r = 2$ for the Composite Trapezoidal Rule.

So for this method you must implement, validate, and compare behavior to theory and the other methods.

Composite Midpoint Rule

The Composite Midpoint Rule with m intervals is given by

$$I_m^{cmp} = \sum_{i=1}^{m} H_m f(x_i) = H_m \sum_{i=1}^{m} f_i, \quad x_i = \frac{a_i + b_i}{2}$$

and the error expression is known to be

$$E_m^{cmp} = I(f) - I_m^{cmp}(f) = (b-a)\frac{H_m^2}{24}f'' + O(H_m^3).$$

Note that it is order 2 just like the Composite Trapezoidal Rule.

The error estimate from the coarse/fine combination, $\alpha = 1/3$, is easily seen to be from the expression in the class notes of the form

$$E_m^{cmp} \approx \frac{3^r}{(3^r - 1)}(I_{3m}^{cmp} - I_m^{cmp})$$

$$E_{3m}^{cmp} \approx \frac{1}{(3^r - 1)}(I_{3m}^{cmp} - I_m^{cmp})$$

where $r = 2$ for the Composite Midpoint Rule.

Expression to be derived: For the Composite Midpoint Rule global mesh refinement with $\alpha = 1/3$ allows complete reuse of function evaluations. Show that the efficient expression for I_{3m}^{cmp} in terms of I_m^{cmp} and the new function evaluations required for $3m$ intervals has the form

$$I_{3m}^{cmp} = \frac{1}{3}\, I_m^{cmp} + H_{3m} \sum_{2m \text{ new pts}} f(x_i).$$

Of course, your answer must include where the new points are relative to the points used for I_m^{cmp} and how you generate them.

So for this method you must implement, validate, and compare behavior to theory and the other methods after deriving the efficient implementation form.

The Integrals:

You must validate the correctness of your implementations in addition to investigating the behavioral comparisons mentioned above. Appropriate basic integration problems can be used for this task. However, your evaluation and comparisons of the methods must include the following problems (whose analytical solutions are given).

$$\int_0^3 e^x dx = e^3 - 1 \tag{1}$$

$$\int_0^{\frac{\pi}{3}} e^{\sin(2x)} \cos(2x) dx = \frac{1}{2}\left(-1 + e^{\frac{\sqrt{3}}{2}}\right) \tag{2}$$

$$\int_{-2}^{1} \tanh(x) dx = \ln\left(\frac{\cosh(1)}{\cosh(2)}\right) \tag{3}$$

$$\int_0^{3.5} x\cos(2\pi x) dx = -\frac{1}{2\pi^2} \tag{4}$$

$$\int_{0.1}^{2.5} \left(x + \frac{1}{x}\right) dx = \frac{2.5^2 - 0.1^2}{2} + \ln(2.5/0.1) \tag{5}$$

Performance Tradeoffs Comment

Note that the Composite Midpoint and Composite Trapezoidal Rules are both order 2. But for complete reuse in the global mesh refinement the Composite Midpoint Rule adds intervals more quickly than the Composite Trapezoidal Rule and therefore has the potential to do more work than knowledge of the true error might require.

The Tasks:

1.1. Provide systematic evidence of the correct execution of your codes.

1.2. For each integral in the problems list, estimate the subinterval size needed to satisfy various error demands using the simple composite quadrature methods and a simple a priori error bound based on the composite methods error expression. The relative error requires knowledge of the size of the answer I. You may use your knowledge of the exact answer for each to make these predictions. Compare these interval sizes to those produced by the global mesh refinement algorithms.

1.3. Run the codes for various error requirements and terminate the global mesh refinement when the predicted error satisfies those requirements. Summarize appropriately and concisely your observations, compare the observed performance to predictions and explain based on knowledge of the methods and the particular problems. Comment on any behavioral differences observed between the problems and explain them based on differences in the functions being integrated. As mentioned above, discuss and compare the number of function evaluations for a particular level of accuracy. Your discussions should include comparing the accuracy actually achieved using the true error computed from your analytical solutions to the integrals.

Other Test Problems

After you have submitted you solutions you should make an appointment with the TA Joshua Gonzalez. He will ask you to demonstrate your code on some test problems as a final evaluation of your codes correctness. You should also be prepared to explain the design and operation of your code to him.

Submission of Results

Expected results comprise:

- A document describing your solutions as prescribed in the notes on writing up a programming solution posted on the class website.
- The source code, makefiles, and instructions on how to compile and execute your code including the Math Department machine used, if applicable.
- Code documentation should be included in each routine.
- All text files that do not contain code or makefiles must be PDF files. **Do not send Microsoft word files of any type**.

These results should be submitted by 11:59 PM on the due date.