CS412: INTRODUCTION TO NUMERICAL ANALYSIS 11/18/10

Lecture 20: Numerical Integration III

Instructor: Professor Amos Ron Scribes: Mark Cowlishaw, Yunpeng Li, Nathanael Fillmore

For the last few lectures we have discussed numerical approximations of the definite integral

$$I(f) = \int_a^b f(t)dt$$

Our general approach is to perform polynomial interpolation of the function over $[a, b]$, then integrate the polynomial. We showed that, if we use the Lagrange representation, the approximation of the definite integral can be written as a simple weighted sum of the function values.

$$I(f) \approx \sum_{i=0}^{n} f(t_i) \cdot w_i$$

Where $w(i)$ is the integral of the Lagrange polynomial for t_i:

$$w_i = \int_a^b \ell_i(t)dt$$

Note that the weight w_i is distinct from the nth Newton polynomial $\omega(t)$, which comes from the error formula for approximation of definite integrals:

$$\int_a^b f(t)dt - \int_a^b p(t)dt = \int_a^b \frac{f^{(n+1)}(c)}{(n+1)!} \underbrace{\prod_{j=0}^{n}(t - t_j)}_{\omega(t)} dt$$

It is also important to note that the quantity $f^{(n+1)}(c)$, cannot simply be taken outside the integral, as c may be a different value for every t. For this reason, we split up the error analysis into cases. The error analysis for case 1, in which we put restrictions on $\omega(t)|_{[a,b]}$ allows us to calculate the error as

$$\int_a^b f(t)dt - \int_a^b p(t)dt = \frac{f^{(n+1)}(c)}{(n+1)!} \int_a^b \prod_{j=0}^{n}(t - t_j)dt \quad \text{(case 1)}$$

However, the analysis that allows us to express the error this way is beyond the scope of this lecture.

1 Composite Rules

If we examine the error formula for any of the simple rules, for example, the error for Simpson's rule:

$$E_S(f) \quad = \quad -\frac{f^{(4)}(c)}{2880}(b-a)^5$$

we see that the error becomes smaller as the size of the interval $[a,b]$ shrinks. As we discussed last time, this leads to the idea of *composite rules*. In a composite rule, we split up the interval $[a,b]$ into equidistant partitions, estimate the integral over each partition using a simple rule, then add the estimates together to reach our final approximation. As we saw last time, a simple rule with an error formula of the form:

$$E_{\text{rule}} \quad = \quad \# f^{(j)}(c) \cdot (b-a)^{j+1}$$

where # is some constant, has a corresponding composite rule with error formula

$$E_{\text{composite-rule}} \quad = \quad \# f^{(j)}(c) \cdot h^j \cdot (b-a)$$

where $h = (b-a)/N$ is the size of each subinterval. To illustrate, here are the error formulas for the composite rules corresponding to the simple rules we have studied.

Rectangle Rule

$$E_{CR}(f,h) \quad = \quad \frac{1}{2} f'(c) \cdot h \cdot (b-a) \tag{1}$$

Midpoint Rule

$$E_{CM}(f,h) \quad = \quad \frac{1}{24} f''(c) \cdot h^2 \cdot (b-a) \tag{2}$$

Trapezoid Rule

$$E_{CT}(f,h) \quad = \quad -\frac{1}{12} f''(c) \cdot h^2 \cdot (b-a) \tag{3}$$

Simpson's Rule

$$E_{CS}(f,h) \quad = \quad -\frac{1}{2880} f^{(4)}(c) \cdot h^4 \cdot (b-a) \tag{4}$$

Note that, in each case, a high power of h, corresponding to an algorithm that quickly converges to small error, requires higher fidelity of the function f.

Note that the composite rules are to basic rules as spline interpolation is to polynomial interpolation. Indeed a composite rule essentially fits a spline to the given function and then interpolates the spline.

To illustrate the use of these error formulas, consider the following example:

EXAMPLE 1.1. *Compute the number of function evaluations required to approximate* $\log(2)$ *with error* $\leq 10^{-8}$

Solution: Recall that we can represent the natural logarithm of 2 as an integral.

$$\log(2) = \int_1^2 \frac{1}{t} dt$$

To approximate this integral, we should first consider which algorithm to use. Since $1/t$ is infinitely differentiable on $[1, 2]$, we are free to use any of the rules, so we choose composite Simpson's rule, since it converges at a rate of h^4. We know that $f^{(4)}(t) = 24/t^5$, substituting into equation 1 yields:

$$\begin{aligned}
10^{-8} &\geq \frac{1}{2880} \left\| \frac{24}{c^5} \right\|_{\infty,[1,2]} \cdot h^4 \cdot (2-1) \\
10^{-8} &\geq \frac{1}{2880} 24 \cdot h^4 \\
0.0331 &\geq h \\
N &\approxeq 30
\end{aligned}$$

Thus our approximation will have about 30 subintervals. This will require 30 evaluations at the midpoints and 31 evaluations at the endpoints, for a grand total of 61 function evaluations. ⊠

2 Extrapolation

If we are given an error formula for a numerical method that is an *exact* measure of the error, we can sometimes use that formula to derive better numerical methods. This process is called *extrapolation.* For example, consider the error formula for the composite midpoint rule.

$$E_{CM}(f, h) = \underbrace{\frac{1}{24} f''(c) \cdot (b-a)}_{(*)} \cdot h^2 \qquad (5)$$

This formula is *almost* good enough to use for extrapolation. However, the constant $(*)$ depends both on the function f and the size of the interval h. For extrapolation, we would like a constant that depends only on the function f. As it turns out, it is possible to rewrite equation 5 as follows:

$$E_{CM}(f, h) = c_f \cdot h^2 + \tilde{c_f}(h) \cdot h^4$$

where c_f is a constant dependent *only* on f, and $\tilde{c_f}(h)$ is a constant dependent on both f and h. We can rewrite the error formula for the composite trapezoid rule (equation 1) similarly:

$$E_{CT}(f, h) = k_f \cdot h^2 + \tilde{k_f}(h) \cdot h^4$$

It also turns out that, similar to the standard form of the error functions, $k_f = -2 \cdot c_f$. Now, consider the definition of the error functions for the composite midpoint and composite trapezoid rules.

$$I(f) - I_{CM}(f,h) = E_{CM}(f,h) \tag{6}$$
$$I(f) - I_{CT}(f,h) = E_{CT}(f,h) \tag{7}$$

If we multiply the top equation by 2 and add it to the bottom equation, this yields:

$$3 \cdot I(f) - I_{CT}(f,h) - 2 \cdot I_{CM}(f,h) = E_{CT}(f,h) + E_{CM}(f,h)$$

Now, since $k_f = -2 \cdot c_f$, the first term in each error formula cancels, leaving only some constant times h^4, thus we have:

$$3 \cdot I(f) - I_{CT}(f,h) - 2 \cdot I_{CM}(f,h) = O(h^4)$$

How does this translate into a new method? We simply estimate the definite integral by applying the composite midpoint rule with subinterval size h, multiplying the result by two and adding this to the result of one application of the trapezoid rule with identical subinterval size h, then dividing the entire sum by three. This method has error $O(h^4)$, since:

$$I(f) - \frac{I_{CT}(f,h) - 2 \cdot I_{CM}(f,h)}{3} = O(h^4)$$

How good is this new method? If we look at the weights applied to each interpolation point, we see that the composite midpoint rule applies weights:

$$\frac{2}{3}\left[f\left(a+\frac{h}{2}\right) + f\left(a+\frac{3h}{2}\right) + \cdots + f\left(b-\frac{h}{2}\right)\right]$$

while the trapezoid rule applies weights:

$$\frac{1}{3}\left[\frac{1}{2} \cdot f(a) + f(a+h) + f(a+2h) + \cdots + f(b-h) + \frac{1}{2} \cdot f(b)\right]$$

Combining these yields:

$$\frac{1}{6}\left[f(a) + 4 \cdot f\left(a+\frac{h}{2}\right) + 2 \cdot f(a+h) + 4 \cdot f\left(a+\frac{3h}{2}\right) + 2 \cdot f(a+2h) + \ldots \right.$$
$$\left. \cdots + 2 \cdot f(b-h) + 4 \cdot f\left(b-\frac{h}{2}\right) + f(b)\right]$$

Close inspection reveals that this new rule is identical to composite Simpson's rule. Can we use extrapolation to produce rules that are better than composite Simpson's rule? The answer is yes, we can do this by mixing composite Simpson's rule with itself.

As before, we can write the error formula for composite Simpson's rule as a lower order term with a constant independent of h, and a higher order term with a constant dependent on h.

$$E_{CS}(f,h) \quad = c_f \cdot h^4 + \tilde{c_f}(h) \cdot h^6 \tag{8}$$

Furthermore, we can calculate the error formula for composite Simpson's rule with a subinterval width twice as large:

$$E_{CS}(f,2h) \quad = c_f \cdot (2h)^4 + \tilde{c_f}(h) \cdot h^6 \tag{9}$$

Note that the first term in equation 9 will be exactly sixteen times the first term in equation 8, so that combining the two rules produces error proportional to the sixth power of h.

$$16 \cdot I_{CS}(f,2h) - I_{CS}(f,2h) \quad = \quad O(h^6)$$

Thus, we can produce a rule with error that converges as the sixth power of h using:

$$I(f) - \frac{16 \cdot I_{CS}(f,2h) - I_{CS}(f,2h)}{15} \quad = \quad O(h^6)$$