Numerical Analysis

Math 370 MWF 2:30-3:25
Spring 1997 201 Fowler

(04.07.97) Newton-Cotes Formulae; Composite Numerical Integration

Newton-Cotes Formulas

The official name for the quadrature rules that Trapezoid and Simpson fall under is called the $(n+1)$**-point closed Newton-Cotes formula**. In this case n is the degree of the Lagrange Polynomial used to do the interpolation for $f(x)$ in the integral.

In general, the **closed Newton-Cotes formulae** look like:
If n **is even** and $f \in C^{n+2}[a,b]$

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+3} f^{(n+2)}(\xi)}{(n+2)!} \int_0^n t^2(t-1)\dots(t-n)\,dt$$

and if n **is odd** and $f \in C^{n+1}[a,b]$

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+2} f^{(n+1)}(\xi)}{(n+2)!} \int_0^n t(t-1)\dots(t-n)\,dt$$

where $\xi \in [a,b]$ and $h = (b-a)/n$ and $a_i = \int_{x_0}^{x_n} L_{n,i}(x)\,dx$
What is the **precision** of an $(n+1)$-point closed Newton-Cotes formula?

The Midpoint Rule is the first example of an $(n+1)$**-point open Newton-Cotes formula**
In the open Newton-Cotes formulas the nodes x_k do not necessarily fall within the interval of integration one is interested in. In fact, they are defined saying that $h = (b-a)/(n+2)$ and $x_k = x_0 + kh$ for $k = 0, 1, \dots n$ and $x_0 = a + h$

EXERCISE: Show that $x_n = b - h = x_0 + nh$

In general, **the open Newton-Cotes formulae** look like:
If n **is even** and $f \in C^{n+2}[a,b]$

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+3} f^{(n+2)}(\xi)}{(n+2)!} \int_{-1}^{n+1} t^2(t-1)\dots(t-n)\,dt$$

and if n **is odd** and $f \in C^{n+1}[a,b]$

$$\int_a^b f(x)\,dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+2} f^{(n+1)}(\xi)}{(n+2)!} \int_{-1}^{n+1} t(t-1)\dots(t-n)\,dt$$

where $\xi \in [a,b]$ and $h = (b-a)/(n+2)$, $x_{-1} = a$, $x_{n+1} = b$ and $a_i = \int_{x_{-1}}^{x_{n+1}} L_{n,i}(x)\,dx$
What is the precision of an $(n+1)$-point open Newton-Cotes formula?

What's the difference between the open and closed Newton-Cotes formulae?

EXERCISE: Use the general form of the open Newton-Cotes formula to write down the formula for Midpoint Rule

Here are the definitions of the standard numerical quadrature rules that you should know:
Midpoint (1-point open Newton-Cotes)

$$M = (b-a) f\left(\frac{a+b}{2}\right) + \frac{h^3}{3} f''(\xi), \qquad \text{where } h = \frac{b-a}{2}$$

Trapezoid (2-point closed Newton-Cotes)

$$T = \frac{(b-a)}{2}\left[f(a) + f(b)\right] - \frac{h^3}{12} f''(\xi), \qquad \text{where } h = b-a$$

Simpson (3-point closed Newton-Cotes)

$$S = \frac{(b-a)}{6}\left[f(a) + 4f(\frac{a+b}{2}) + f(b)\right] - \frac{h^5}{90} f^{(4)}(\xi) \qquad \text{where } h = \frac{b-a}{2}$$

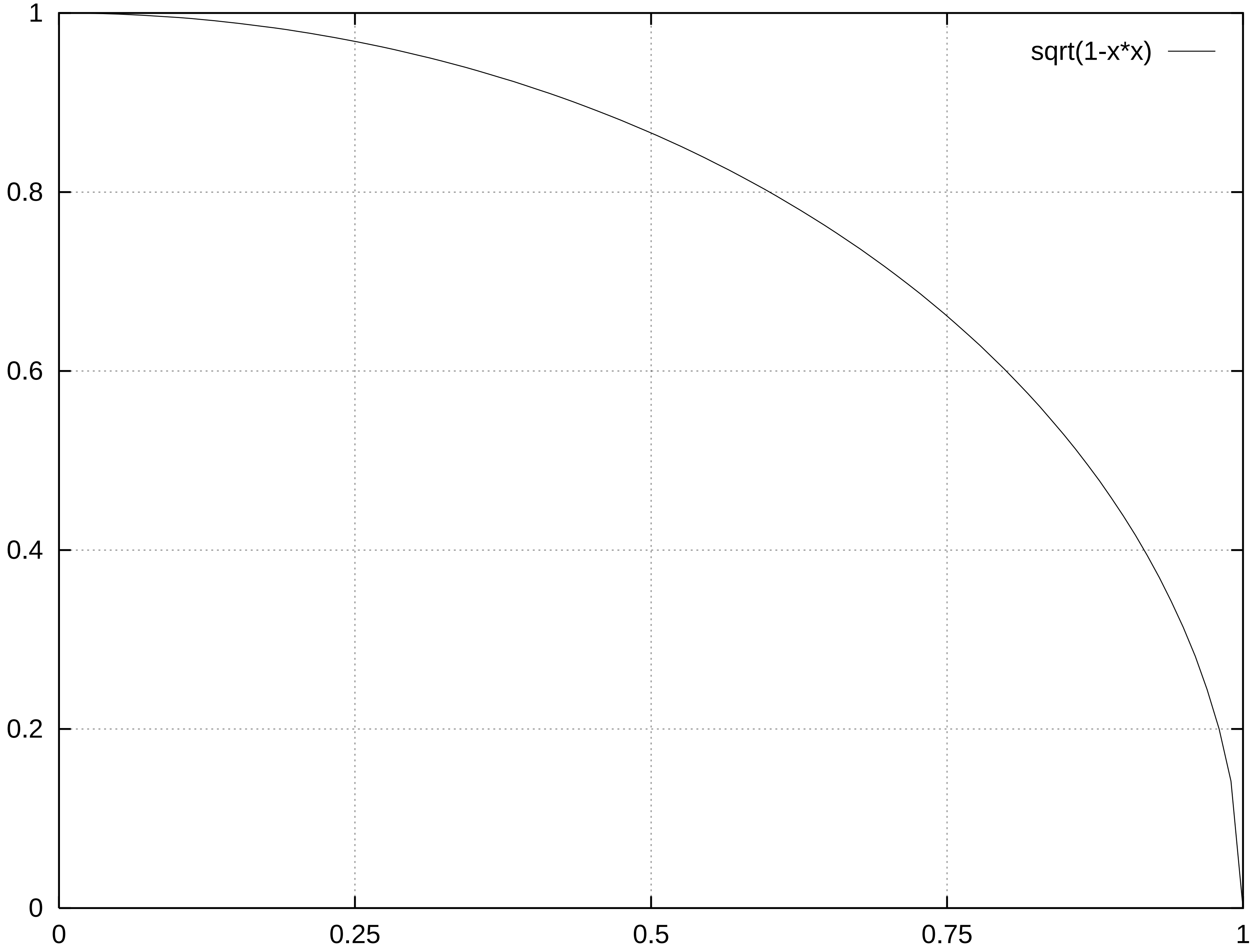
Composite Numerical Integration
Of course, usually when these rules are used in practice, the interval of integration is split up into N subintervals and the rule is applied repeatedly on each subinterval in order to compute a more accurate approximation to the value of the integral as N increases and the size of each subinterval decreases.

We are interested in looking at what happens to the error formula in each of these cases if we apply them repeatedly on N subintervals.

EXERCISE
Let's consider the Trapezoid Rule. Sketch the approximation that using the Trapezoid Rule with $N = 2$ would look like and write down the approximation T to I the exact value of the integral $\int_0^1 \sqrt{1-x^2}\, dx$. We'll denote this by $T(2, \sqrt{1-x^2}, 0, 1)$. Find $|I - T(2, f, a, b)|$

Repeat the exercise with $N = 4$ to compute $T(4, \sqrt{1-x^2}, -1, 1)$. Find $|I - T(4, f, a, b)|$

Does your picture remind you of any topic from **interpolation**?


sqrt(1-x*x)
1
0.8
0.6
0.4
0.2
0
0
0.25
0.5
0.75
1

Error Term of Composite Trapezoid
Starting with the formula for the Trapezoid Rule, we can show that the error formula for the Composite Trapezoid Rule is $O(\Delta x^2)$ and has precision 1.

$$\text{CT} = \text{T}(N, f, a, b) = \frac{\Delta x}{2}\left[f(a) + 2\sum_{j=1}^{N-1} f(a + j\Delta x) + f(b)\right] - \frac{(b-a)}{12}(\Delta x)^2 f''(\mu)$$

where $\mu \in (a, b)$ and $\Delta x = (b-a)/N$

Similarly,
Composite Midpoint Method

$$\text{CM} = \text{M}(N, f, a, b) = \sum_{k=1}^{n} f(x_k)\Delta x + \frac{(b-a)}{24}(\Delta x)^2 f''(\mu)$$

where $\mu \in (a, b)$, $x_k = a + (k-1/2)\Delta_x$ and $\Delta x = (b-a)/n$ and n is the number of subintervals in $[a, b]$
Composite Simpson's Method

$$\text{CS} = \frac{2}{3}CM + \frac{1}{3}CT - \frac{(b-a)}{2880}(\Delta x)^4 f^{(4)}(\mu)$$

where $\mu \in (a, b)$, $\Delta x = (b-a)/n$ and n is the number of subintervals in $[a, b]$
Example
Now that we know expressions for the error terms of these methods we can use this information for **error control**. Consider $\int_1^3 e^{x^2}\,dx$.
How many N subintervals would we have to split [1,3] into in order to be *confident* that you have computed the integral to with 10^{-4} of its exact value?
Compare how many subsintervals you need using Midpoint versus using Simpson's.

Effect of Round-Off Error on Composite Numerical Integration
Clearly when we do numerical integration we do quite a bit of function evaluation. As we saw in numerical differentiation, everytime we evaluate a function $f(x)$ the computer is using $\tilde{f}(x) + e(x)$. Lets consider the Trapezoid Method and try and compute $E(\Delta x)$ and see what happens as $\Delta x \to 0$