**COMPUTER SCIENCE 349A**
**Handout Number 30**

# ROUNDOFF ERROR ANALYSIS FOR COMPOSITE NEWTON-COTES FORMULAS

For simplicity consider the composite Simpson's rule

$$\int_a^b f(x)dx \approx \frac{h}{3}\left[ f_0 + 2\sum_{j=1}^{m-1} f_{2j} + 4\sum_{j=1}^{m} f_{2j-1} + f_{2m} \right],$$

where $f_i = f(x_i)$ and $h = \frac{b-a}{2m}$. When this formula is evaluated using floating-point arithmetic, there are two main sources of roundoff error: in the evaluation of the values $f_i$, and in calculating the summations. For now, we'll consider only the effect on the computed approximation to $\int_a^b f(x)dx$ of roundoff errors that occur in evaluating the values $f_i$.

Suppose $\tilde{f}_i$ denotes the computed approximation to $f_i$ (that is, $\tilde{f}_i$ is inexact due to roundoff error).

Denote the roundoff error by $e_i = \tilde{f}_i - f_i$ and suppose that $|e_i| \le \varepsilon$ for all $i$ (that is, we are assuming that $\varepsilon$ is the largest possible roundoff error when evaluating $f(x_i)$ regardless of the value of $x_i$).

Then the accumulated effect of these roundoff errors in the composite Simpson's rule is

$$\begin{aligned}
\text{ERR} &= \left| \frac{h}{3}\left( \tilde{f}_0 + 2\sum_{j=1}^{m-1} \tilde{f}_{2j} + 4\sum_{j=1}^{m} \tilde{f}_{2j-1} + \tilde{f}_{2m} \right) - \frac{h}{3}\left( f_0 + 2\sum_{j=1}^{m-1} f_{2j} + 4\sum_{j=1}^{m} f_{2j-1} + f_{2m} \right) \right| \\
&= \left| \frac{h}{3}\left[ e_0 + 2\sum_{j=1}^{m-1} e_{2j} + 4\sum_{j=1}^{m} e_{2j-1} + e_{2m} \right] \right|
\end{aligned}$$

Thus,

$$
\begin{aligned}
\text{ERR} &\le \frac{h}{3}\left[\ |e_0| + 2\sum_{j=1}^{m-1}|e_{2j}| + 4\sum_{j=1}^{m}|e_{2j-1}| + |e_{2m}|\ \right] \\
&\le \frac{h}{3}\left[\varepsilon + 2(m-1)\varepsilon + 4m\varepsilon + \varepsilon\right] \\
&= \frac{h}{3}(6m\varepsilon) \\
&= 2mh\varepsilon \\
&= (b-a)\varepsilon, \ \ \text{since } \ h = \frac{b-a}{2m}\ .
\end{aligned}
$$

Thus the composite Simpson's rule is **very stable**: roundoff errors do not disastrously accumulate as $m \to \infty$ (or $h \to 0$)

<u>Notes</u>

1. The above analysis is not complete: it ignores the roundoff errors made in doing the approximately $2m$ additions (in the composite Simpson's rule). As $m \to \infty$, the accumulated effect of these errors will eventually make the total roundoff error large (for extremely large values of *m*, for example if *m* were several million or more). However, sufficient accuracy can usually be obtained without using such large values of *m*, so algorithms for quadrature are stable

2. The roundoff error analysis for other quadrature formulas is similar.

<u>Overall conclusion</u>: quadrature is stable. Sufficient accuracy can be obtained before the negative effects of the roundoff error due to the $2m$ additions in the quadrature formula become apparent. That is, the magnitude of the accumulated roundoff error when using a quadrature formula is usually just the magnitude of the roundoff due to evaluating the $f_i$, and the above analysis shows that this is not large.