General Notes for Reports

- Write-ups should be organized into short, labelled sections. You need not use exactly the same labels & sections as in this sample. Lists can be used, but should be in sentence form.
- Correctness, clarity and conciseness of communication is very important.
- Effective communication of your ideas is an important component of the grading. Short bulleted lists, when appropriate, are encouraged.
- Have your reports proofread before submission.
- Computer figures should be annotated to be as standalone as possible. It is alright to repeat results in the text of your report.
- Adhere to page limits. Extra material may be included as an Appendix, but might not be considered in the grading.
- Most important will be how you address the main issues. What was the experiment/simulation about? What were the key results? What mathematical conclusions/observations can be drawn from the lab?

Comparing Numerical Quadratures

1) Introductory Sections

Motivation: In the Matlab demo *w1simp.m*, several test integrations are performed in an attempt to quantify the concepts of convergence, accuracy, and efficiency as discussed in the lecture of 06 January. Three methods are applied: a uniform discretization of Simpson's rule, and the Matlab built-in quadrature routines *quad* and *quadl*.

2) Theory Sections

Simpson's Rule: The first of the numerical quadratures involve the implementation of Simpson's rule. To approximate the integral of $f(t)$ over the interval $a \leq t \leq b$, the interval is sampled at $N+1$ (an odd integer) uniformly spaced points $t_0, t_1 \ldots t_m, \ldots t_N$ where $t_0 = a$ and $t_N = b$. The Simpson's rule approximate quadrature for the integral

$$I = \int_a^b f(t)dt \approx \triangle t \sum_{m=0}^{N} c_m f(t_m) \equiv I_{simp}(N)$$

where $\triangle t \equiv (b-a)/N$ and $t_m = a + m\triangle t$. The Simpson's rule coefficients are given by

$$c_m = \begin{cases} 1/3 & \text{for } m = 0, N \text{ (endpoints)} \\ 4/3 & \text{for odd } m \\ 2/3 & \text{for even } m \end{cases} .$$

For smooth functions $f(t)$, the error of the numerical approximation from the true integral depends on $\triangle t$ as

$$|I - I_{simp}| \propto K \triangle t^4 = K_1 \ N^{-4} \tag{1}$$

where under certain conditions, the constant K can be zero and the error decreases at an even faster power of $\triangle t$. Thus the accuracy, or rate of convergence, is at least $\triangle t^4$. The number of evaluations of the function $f(t)$, $N+1$, is used as our measure of the numerical effort of the Simpson's rule approximation.

3) Methodology Sections

Test Quadratures: A known integral is chosen for our test quadrature

$$I = 3 \int_0^{\sqrt{3}} \frac{dt}{1+t^2} = 3 \arctan t \Big|_0^{\sqrt{3}} = \pi \ .$$

Based upon this exact value, the numerical error $E(N) = |I - I_{simp(N)}|$ is calculated for the values $N = 8, 16, 32, 64, 128$ which corresponds to a decreasing sequence of $\triangle t$-values. Anticipating a power law relation, the error, $E_{simp}(N)$, is plotted on a log-log graph.

The built-in matlab quadrature command *quad* takes an optional tolerance argument which can be indirectly used to control the accuracy of its numerical approximation. The default tolerance, 10^{-6}, is used, in addition to the values 10^{-8} and 10^{-10} – these smaller values of the tolerance should improve the approximation. The *quad* command also allows for an optional output of the number of function evaluations which is used to obtain an effective value of N for graphical comparison with the earlier Simpson's rule quadratures.

A single call to the *quadl* command at the default tolerance (10^{-6}) gives an additional data point $E_{quadl}(N)$ where the effective N is one less than the number of function evaluations.

4) Observations & Results Sections

Error versus Effort: The results are summarized on a log-log plot of the error values, $E(N)$.

- The $\times$ marks show that the Simpson's rule approximation gives decreasing error for decreasing values of $\triangle t = (b-a)/N$. Furthermore, a line of slope -4 has been added to plot to indicate that the rate of error decrease is consistent with the $\triangle t^4$ power law (1).

- The $\circ$ marks for the *quad* command at tolerances 10^{-6} and 10^{-8} lie very close to the Simpson's rule error values. However, at the smallest tolerance, 10^{-10}, the error actually increased from the larger tolerance of 10^{-8}.

- The $\star$ mark records the smallest error value, $E_{quadl} \doteq 1.161 \times 10^{-12}$ obtained from only 47 function evaluations by the *quadl* routine.

5) Conclusions & Discussion Sections

Conclusions: The decreasing error of the Simpson's rule approximation verifies the convergence of the quadrature through the values $N = 8, 16, 32, 64, 128$. In addition, the -4 slope of the log-log plot of $E_{simp}(N)$ gives quantitative support for the $\triangle t^4$ accuracy (1) or rate of convergence. This means that doubling the computational effort (N) roughly decreases the error by a factor of 16.

That the first two values of the *quad* routine are roughly as effective as Simpson's rule reflects the fact that *quad* uses a method which is an adaptive modification of the Simpson's method which seems to converge by a similar fourth-order accuracy law. The increase of the error at the smallest tolerance suggests round-off effects due to the finite-digit limit of matlab (machine $\epsilon = 2.2204 \times 10^{-16}$).

With only one data point, the accuracy of the *quadl* routine cannot be determined. However, the effectiveness based on number of function evaluations is considerably better than the Simpson's rule based results.

Questions & Future Work: Several open issues have been motivated by the results shown here.

- Verify that the uniform discretization of Simpson's rule also displays loss of convergence for larger values of N. Is there anyway to show that round-off effects are to blame?

- Estimate the constant K_1 in the Simpson's rule accuracy law by finding the best-fit line through the error points (use the *polyfit* command).

- Investigate if the *Lobatto* algorithm of the *quadl* routine also displays a power law for the rate of convergence. How is the Lobatto algorithm different from Simpson's rule?

QUADRATURE COMPARISON FOR $I = \int_0^{\sqrt{3}} \frac{3}{1+t^2}\,dt = \pi.$