1.8 PROBLEM: SUMMING SERIES

A classic numerical problem is the summation of a series to evaluate a function. As an example, consider the infinite series for $\sin x$:

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots \qquad \text{(exact)}.$$

Your **problem** is to use this series to calculate $\sin x$ for $x < 2\pi$ and $x > 2\pi$, with an absolute error in each case of less than 1 part in 10^8. While an infinite series is exact in a mathematical sense, it is not a good algorithm because we must stop summing at some point. An algorithm would be the finite sum

$$\sin x \simeq \sum_{n=1}^{N} \frac{(-1)^{n-1} x^{2n-1}}{(2n-1)!} \qquad \text{(algorithm)}. \tag{1.15}$$

But how do we decide when to stop summing? (Do not even think of saying, "When the answer agrees with a table or with the built-in library function.")

1.8.1 Numerical Summation (Method)

Never mind that the algorithm (1.15) indicates that we should calculate $(-1)^{n-1}x^{2n-1}$ and then divide it by $(2n-1)!$ This is not a good way to compute. On the one hand, both $(2n-1)!$ and x^{2n-1} can get very large and cause overflows, even though their quotient may not. On the other hand, powers and factorials are very expensive (time-consuming) to evaluate on the computer. Consequently, a better approach is to use a single multiplication to relate the next term in the series to the previous one:

$$\frac{(-1)^{n-1}x^{2n-1}}{(2n-1)!} = \frac{-x^2}{(2n-1)(2n-2)} \frac{(-1)^{n-2}x^{2n-3}}{(2n-3)!}$$

$$\Rightarrow \qquad n\text{th term} = \frac{-x^2}{(2n-1)(2n-2)} \times (n-1)\text{th term}. \tag{1.16}$$

While we want to ensure definite accuracy for $\sin x$, that is not so easy to do. What is easy to do is to assume that the error in the summation is approximately the last term summed (this assumes no round-off error, a subject we talk about in Chapter 2, "Errors & Uncertainties in Computations"). To obtain an absolute error of 1 part in 10^8, we then stop the calculation when

$$\left| \frac{n\text{th term}}{\text{sum}} \right| < 10^{-8}, \tag{1.17}$$

where "term" is the last term kept in the series (1.15) and "sum" is the accumulated sum of all the terms. In general, you are free to pick any tolerance level you desire, although if it is too close to, or smaller than, machine precision, your calculation may not be able to attain it. A pseudocode for performing the summation is

```
term = x, sum = x, eps = 10^(-8)                    # Initialize do
  do term = -term*x*x/(2n+1)/(2*n-2);                  # New wrt old
  sum = sum + term                                       # Add term
  while abs(term/sum) > eps                        # Break iteration
end do
```

1.8.2 Implementation and Assessment

i) Write a program that implements this pseudocode for the indicated x values. Present the results as a table with headings x imax sum $|\text{sum} - \sin(x)|/\sin(x)$ where **`sin(x)`** is the value obtained from the built-in function. The last column here is the relative error in your computation. Modify the code that sums the series in a "good way" (no factorials) to one that calculates the sum in a "bad way" (explicit factorials).

ii) Produce a table as above.

iii) Start with a tolerance of 10^{-8} as in (1.17).

iv) Show that for sufficiently small values of x, your algorithm converges (the changes are smaller than your tolerance level) and that it converges to the correct answer.

v) Compare the number of decimal places of precision obtained with that expected from (1.17).

vi) Without using the identity $\sin(x+2n\pi) = \sin(x)$, show that there is a range of somewhat large values of x for which the algorithm converges, but that it converges to the wrong answer.

vii) Show that as you keep increasing x, you will reach a regime where the algorithm does not even converge.

viii) Now make use of the identity $\sin(x + 2n\pi) = \sin(x)$ to compute $\sin x$ for large x values where the series otherwise would diverge.

ix) Repeat the calculation using the "bad" version of the algorithm (the one that calculates factorials) and compare the answers.

x) Set your tolerance level to a number smaller than machine precision and see how this affects your conclusions.

Beginnings are hard.

—*Chaim Potok*