MS-C1650 Numeerinen analyysi, Exercise 5, Guidelines

Lauri Perkkiö

May 17, 2019

Problem 1

Binary floating point numbers (3 bits + 1 for sign + 1 for exponent + 1 for exponent's sign). Choose first +, $m = 0$. Then, $2^m = 1$, and the possible floating point binary numbers are

$$\begin{aligned}
(0.000)_2 &= 0 \\
(0.100)_2 &= 2^{-1} = 1/2 \\
(0.010)_2 &= 2^{-2} = 1/4 \\
(0.001)_2 &= 2^{-3} = 1/8 \\
(0.110)_2 &= 2^{-1} + 2^{-2} = 3/4 \\
... &= ...
\end{aligned}$$

And so on, repeat with $m = 1$, $m = -1$, and also with negative sign. In this case, the machine epsilon ϵ means the smallest number above zero.

Normalized numbers: fix $b_1 = 1$. (Compare to the "scientific notation" $0.d_1 d_2 d_3 ... d_r \times 10^n$, where d_1 is non-zero, r is the number of significant digits). Then, the smallest number is...

Problem 2

Let us assume that a root x_* of $f(x)$ is in the interval $(16, 17)$, or by using binary numbers,

$$17 = (10001.000...)_2 > x_* = (10000.b_1 b_2 b_3 ...)_2 > 16,$$

Of course, assume also that $f(16)$ and $f(17)$ differ in sign. Using the bisection method with the initial interval $[16, 17]$, we know immediately that the first ... bits are surely correct. After the second iteration (say, interval $[16.5, 17]$), the first ... bits are surely correct. Etc.

Problem 3

a) Idea: uncertainty in the data $x = 100 \pm 1$, and how it propagates into the output y. Replace $y(x)$ by its linear approximation at $x = 100$, and compute the maximal possible error when $x \in [99, 101]$

b) The error in x starts to dominate the error in the square root evaluation after ... digits are computed precisely.

Morale of the story: In real life, there's usually some error in the data, and "solving the equations up to 100 digits" is pointless. (And one point of "numerical analysis" is to give useful estimates for the numerical error.)

Problem 4

Everything that is needed is in bezier.pdf in MyCourses -¿ Materials.

MATLAB

The method is not obvious:

`http://cr.yp.to/bib/1976/brent-elementary.pdf`