Nature of numerical computations

Responsible teacher: Anatoliy Malyarenko

November 14, 2003

Abstract

Contents of the lecture:

- Absolute and relative errors.
- Number representation.

Absolute and relative errors

Let x_0 be some real number than we need to compute. Let x denotes the result of computations.

Absolute error: $|x - x_0|$.

Relative error: $\dfrac{|x - x_0|}{|x|}$.

Relative error: example

Consider the expression

$$\frac{\left|e^x - 1 - x - \frac{x^2}{2!} - \cdots - \frac{x^n}{n!}\right|}{|e^x|},$$

i.e., the *relative error* of the Taylor approximation. The theory says that for every x this expression goes to 0 an n increases.

Consider the next script:

```
nTerms = 50;
for x = [10 5 1 -1 -5 -10]
    figure;
    term = 1; s = 1;
    f = exp(x)*ones(nTerms, 1);
    for k = 1:nTerms
        term = x.*term/k;
        s = s + term;
        err(k)=abs(f(k)-s);
    end
    relerr = err/exp(x);
    semilogy(1:nTerms, relerr);
    ylabel('Relative error in partial sum');
    xlabel('Order of partial sum');
    title(sprintf('x = %5.2f', x));
end
```

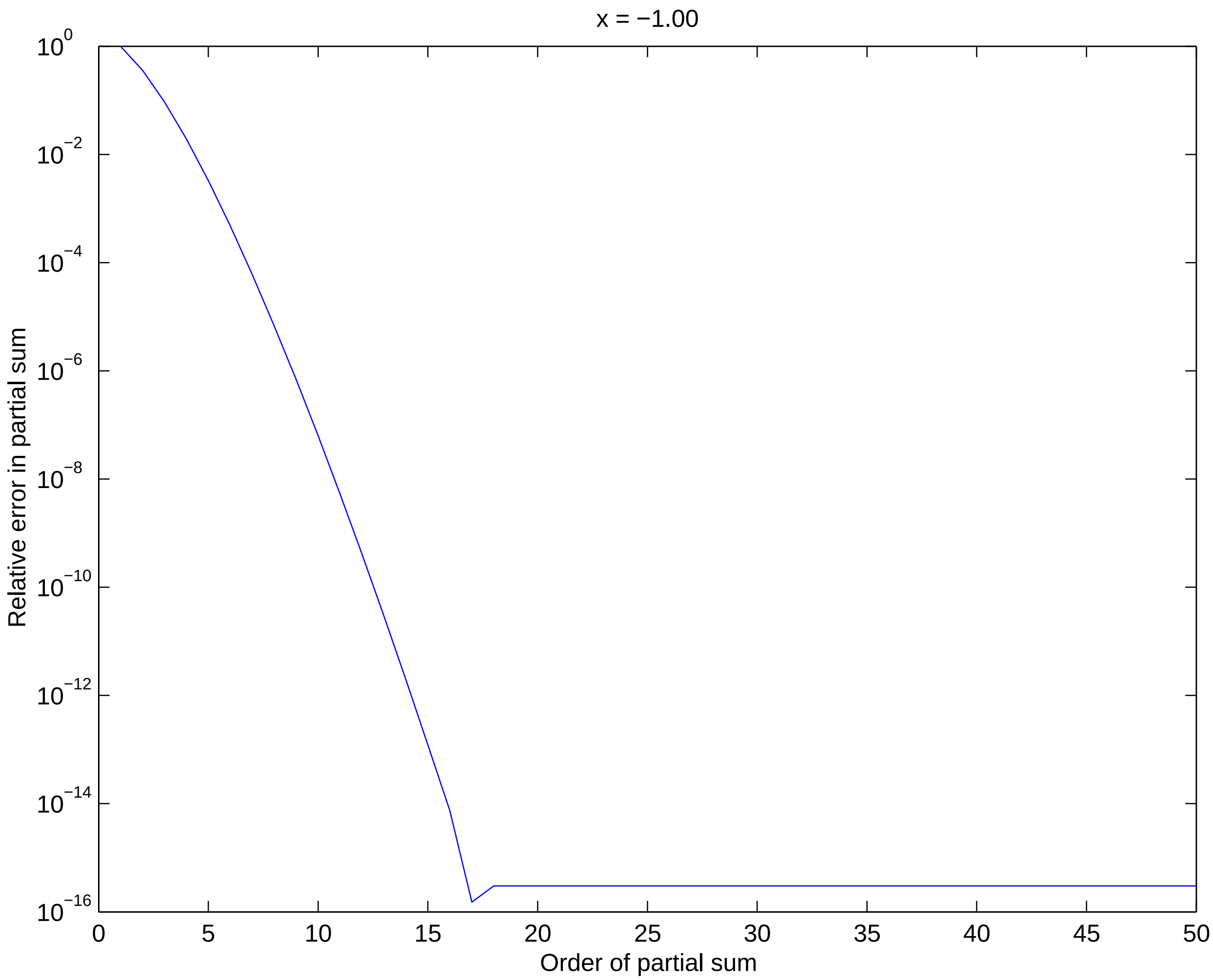

Figure 1: Error in Taylor approximations to e^x, $x = -1$

The script proves that the relative error *does not go to* 0 as the number of terms in the series increases. How to explain this phenomenon?

Number presentation

Usually we represent numbers on a decimal base, for example

$$1957 = 1 \times 10^3 + 9 \times 10^2 + 5 \times 10^1 + 7 \times 10^0.$$

The fractional part is presented by using negative powers, for example

$$0.14 = 1 \times 10^{-1} + 4 \times 10^{-2}.$$

The elements of hardware can not have 10 possible positions, but only 2 positions. On a computer we must use *binary base*, for example

$$0.5 = 2^{-1}.$$

Since only a *finite* memory space is available to store the number, we will have a *rounding error*.

Number presentation in memory

Sixty four bits are used to represent a real number in MATLAB:

- 52 bits are used to represent the fractional part, `f`;
- 11 bits are used to represent the biased exponent, `e`;
- 1 bit is used to represent the sign, `s`.

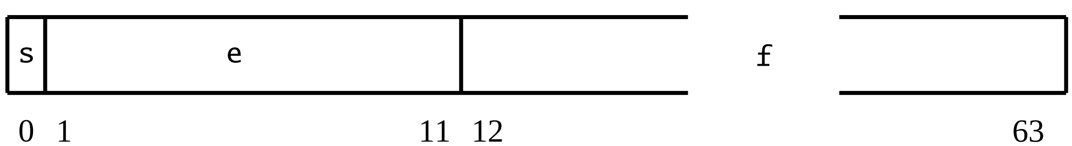

Figure 2: MATLAB real number format

The real number x presents as the *machine number*

$$x = \pm 2^{p} \sum_{k=1}^{52} \alpha_k \cdot 2^{-k}, \tag{1}$$

where every α_k is equal to 0 or 1 and lies in the fractional part. The number p is presented according to the next rules:

e	Sense
00000000000	Presents $+0$ or -0
00000000001	Presents $p = -1022$
...	...
11111111110	Presents $p = 1023$
11111111111	Presents $+\infty$, $-\infty$ and NaN

Table 1: Presentation of exponent

If x is the machine number (1), then the next larger machine number is

$$x_+ = \pm 2^p \left(\sum_{k=1}^{52} \alpha_k \cdot 2^{-k} + 2^{-52} \right).$$

The spacing of these numbers, or the absolute rounding error, is

$$x_+ - x = 2^{-52} \cdot 2^p = 2^{p-52}.$$

The relative spacing, or the relative rounding error, is defined by the ratio

$$\frac{x_+ - x}{|x|} \approx 2^{-52} \approx 2.2204 \times 10^{-16}.$$

If $x = 1$, then the absolute rounding error is equal to the relative rounding error, and both of them are equal to the distance from 1 to the next machine number. This is what we call `eps` in MATLAB.

Conclusions

- ☞ The set of numbers in a computer is finite, since rounding errors.
- ☞ The spacing of computer numbers is *not* uniform, but the relative roundoff error is constant.
- ☞ Algorithms that are equivalent mathematically may behave very different numerically.

Problems

1. The approximation for the cosine function is given by

$$T_n(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \cdots + (-1)^n \frac{x^{2n}}{(2n)!}.$$

   Write M-file that explores the relative error in the partial sums. Use the program at p. 1 as a pattern.

2. The next two functions

$$y_1(x) = \sqrt{x^2+1} - 1 \quad \text{and} \quad y_2(x) = \frac{x^2}{\sqrt{x^2+1}+1}$$

   are mathematically equivalent. Plot the function $|y_1(x) - y_2(x)|$ over increasingly smaller neighbourhoods around $x = 0$. Use `x=linspace(-delta,delta,n)` for $n = 100$ and several values of delta in the interval $[10^{-4}, 10^{-3}]$.

3. (For pass with distinction). Explain the results obtained in Problem 2.