Applied Numerical Methods I

Floating Point Systems

Machine Numbers

A *machine* is a calculator or computer that does arithmetic with a subset of real numbers called *machine numbers*. These numbers are formed by writing them in *normalized form* (scientific notation) with a fixed number of digits $t$ in the mantissa and a fixed number of digits $m$ in the exponent. This truncation leads to a few computational issues not seen with real numbers, to be discussed. Machine numbers have 4 components: a *sign bit* for the mantissa (plus or minus), the mantissa, a sign bit for the exponent, and the exponent. The format can be found on wikipedia under ‘floating point numbers’. We don’t need all the details, but here is a typical representation.

\[
\pm \, t_1 \, t_2 \, t_3 \, t_4 \, t_5 \, \pm \, m_1 \, m_2 \, m_3 
\]

Examples of Machine Numbers

Numbers are stored on machine in binary (base 2), so there are only 2 digits, 0 and 1. For class purposes, however, we often look at machine numbers in decimal because we are more accustomed to thinking in decimal.

Decimal examples. $1313.12$ with $t = 6$ and $m = 4$ is

\[
+ \, 1 \, 3 \, 1 \, 3 \, 1 \, 2 \, + \, 0 \, 0 \, 0 \, 3 
\]

If the same number is put into a different floating point system, say $t = 4, m = 2$, the mantissa must be truncated. The result is

\[
+ \, 1 \, 3 \, 1 \, 3 \, + \, 0 \, 3 
\]

The truncation causes *representation error* in computations using this number. The important idea is that not every real number can be stored on a machine as a machine number.