Scientific computing: Some good advice

Carl Christian Kjelgaard Mikkelsen

October 31, 2012

When programming a computer, you must always be aware that floating point arithmetic is radically different from exact arithmetic. If you ignore the difference, then your programs will frequently either crash or return results which are entirely wrong. Below is a list of some of the things that you should look for. The list is not complete, but it is a good place to start. Many of the entries on list are due to Ole Østerby from the DAIMI in Aarhus, Denmark and are copied almost ad verbatim from his notes.

1. **A good approximation plus a small correction term**

   In many calculations, typically iterations, we shall compute better and (hopefully) better approximations to the solution. If the calculations can be arranged as above, this will always be advantageous. For example, when computing an average of two real numbers

   \[
   a + \frac{b - a}{2} \text{ is better than } \frac{a + b}{2},
   \]

   and when executing Newton’s method for computing the square root of a real number \(s\), then

   \[
   x - \frac{x^2 - s}{2x} \text{ is better than } \frac{1}{2} \left( x + \frac{s}{x} \right).
   \]

2. **Add the small terms first**

   When adding a large number of positive terms you must sum the smallest terms first, otherwise their contribution, which can be significant, is never felt. If the number of terms is extremely large, then you should consider tree-wise addition. If you have both negative and positive terms, then you need to be extra careful, as you might experience catastrophic cancellation, see item 3.

3. **Be very careful when subtracting two real numbers which are almost equal**

   This point is frequently misunderstood. By design, the computer can subtract two floating point numbers \(x\) and \(y\) with a small relative error and

   \[
   \text{fl}(x - y) = (x - y)(1 + \delta), \quad |\delta| \leq u.
   \]
where $fl(x - y)$ is the computed (floating point representation) of $x - y$. So where is the problem? Given a pair of real numbers, then machine must first create a floating point representation of $x$ and $y$ and only then can it subtract the representations. It is the combination of these two operations which can result in disaster. Specifically, we have

$$\frac{|(x - y) - fl(fl(x) - fl(y))|}{|x - y|} \leq \frac{2u |x| + |y|}{1 - 2u |x - y|}$$

In short, if $x$ and $y$ are almost equal, your relative error bound can be very large and you can not necessarily trust the computed value for $x - y$.

4. Avoid large intermediate results on the road to a small final result

A good example is the naive computation of $x \rightarrow e^x$. The Taylor series expansion of $\exp(x)$ at $x_0 = 0$, works fine for positive values of $x$, but we get negative results for many negative values of $x$.

5. Use mathematical reformulations to avoid 3. and 4.

For small values of $x$

$$2 \sin^2 \frac{x}{2} \text{ is better than } 1 - \cos(x)$$

and

$$\frac{x^4}{\sqrt{x^4 + 2} + 2} \text{ is better than } \sqrt{x^4 + 4} - 2.$$  

For negative values of $x$

$$\frac{1}{\sum_{j=0}^{n} (-x)^j j!} \text{ is a better approximation of } e^x = \frac{1}{e^{-x}} \text{ than } \sum_{j=0}^{n} \frac{x^j}{j!},$$  

and it completely eliminates the specific problem mentioned in item 4.

6. Use series expansions to supplement 5

For small values of $x$

$$\frac{1}{2} - \frac{x^2}{24} + \frac{x^4}{720} - \frac{x^6}{40320} + \ldots \text{ is better than } \frac{1 - \cos(x)}{x^2}.$$  

7. Use integer calculations whenever possible

The following MATLAB command will never terminate

```
>> b=0; a=1/1000; while (b<1) b=b+a; end;
```

although we would expect it to terminate after 1000 iterations! It is far safer to use an integer to control the loop, as in
\begin{verbatim}
>>b=0; a=1/1000; for i=1:1000 b=b+a; end;
\end{verbatim}

Similarly, it you should pass integers to subroutines whenever possible.

Historical note: This is the type of mistake which prevented the Patriot missiles from hitting the Scud missiles fired against Israel during the war with Iraq in early 1991.

8. **Be wary of very small number or very large numbers**

   In exact arithmetic \( \sqrt{ab} = \sqrt{a} \sqrt{b} \),

   when \( a \) and \( b \) are positive real numbers. However, if \( a \) and \( b \) are too small (too large) then \( \sqrt{ab} \) underflows (overflows), while \( \sqrt{a} \sqrt{b} \) will succeed. This is due to the fact that there is a smallest (largest) positive floating point number.

   Historical note: This exact problem caused crashes in the HSL software package around 2007.

9. **Look at the numbers once in a while**

   It is a good idea to (write and) check intermediate results, e.g. while testing the program, and judge whether they make sense. Sound judgment and common sense are invaluable helpers.