Dear colleagues,[1]

For many applications a verified solution of a numerical problem with narrow bounds is needed. A fusion reactor is planned to be built in France a few kilometers to our west. To keep the reaction going and stable large systems of equations have to be solved. Computations which we have been asked to verify have sometimes had surprising results. So I want to verify all the computations and simulations essential for this reactor before they are applied. This requires fast hardware support of the necessary arithmetic.

Surprisingly there are problems for which a verified solution can be obtained faster than an approximate solution. For systems of equations, however, a verified solution requires a particular verification step to follow an approximate solution. The verification step needs a different kind of arithmetic (defect correction techniques and interval arithmetic). On existing computers this arithmetic has to be simulated by the floating-point or integer arithmetic provided. Thus the verification step is usually slower by orders of magnitude than the computation of an approximate solution. A consequence of this is that for large problems result verification is simply not possible or it is not applied because of an unrealistic waste of computing power. Quadruple precision arithmetic suffers from the same problem if it is simulated by software.

This need not be so. Verification techniques are not inherently slow. It is the software simulation of the necessary arithmetic which makes them slow. Hardware support for it would drastically change the situation. The two kinds of arithmetic must be brought into balance.

Interval operations can be made as fast as simple floating-point operations. On Intel or AMD processors about 30 additional gates would almost suffice to do this. It cannot be done by putting extra floating-point units on a chip. A hardware implementation of the exact dot product is at least as fast as a (possibly wrong) conventional computation of the dot product in floating-point arithmetic. No clever floating-point simulation of it can come anywhere near this speed.

With respect to the exact dot product Nick Maclaren writes in his mail of Oct. 4, 2006:

Again, this is exactly like mandating 'exact' results for the special functions. 32-bit cos is easy; 64-bit is feasible; 128-bit is not in cases where time and memory are critical; and what about extended? And, if cos must be exact, why not erf?

This seems to me to point out the right way to go. The dot product of two floating-point vectors is the most important and the most frequently used elementary function. Hardware support for the exact dot product for (single and) double precision would be enough for many computational methods to be

[1] Oct. 10, 2006.

greatly enhanced. It also would suffice to correctly compute the vast majority of dot products for higher precision data formats. This is the least that should be done. If a very large exponent range is needed software aids (perhaps sorting or other methods) could be used.

Experience has shown that for numerical computing, manufacturers only implement what the arithmetic standard requires. It would be tragic if a certain lack of familiarity with verification techniques were to prevent a future arithmetic standard from providing the arithmetic needed to allow results to be verified.

A more formal specification of the *Basic Requirements ...* is being prepared by the GAMM-Fachausschuss on Computer Arithmetic and Scientific Computing.

Ulrich Kulisch