VALIDATED NUMERICS AND THE ART OF DIVIDING BY ZERO

WARWICK TUCKER

We will discuss a modern approach to numerical computations, based on interval analysis. Although the theory has been well known since the mid 60's, it is not until recently that fast and efficient implementations of interval algorithms have appeared. Today there exist many good interval packages for e.g. Maple, Matlab, C++, and Fortran, much due to the fact that there now is one globally accepted standard for floating point computations.

Interval analysis is the mathematical foundation of so called auto-validating algorithms. By computing with intervals instead of single points, important properties of the real line can be captured and used in the algorithms. This leads to very robust methods, well suited for ill-conditioned problems.

Auto-validating algorithms produce mathematically correct results, incorporating not only the computer's internal representation of the floating point numbers and its rounding procedures, but also all discretisation errors of the underlying numerical method. Thus the result comes equipped with guaranteed error bounds. With today's computing speeds, this appears to be the only reasonable way to keep track of error propagation.

There are many situations in which the interval algorithm returns a guaranteed result faster than the floating point version delivers an "approximation" (which can be very wrong indeed). Throughout the talk, several such applications will be presented, e.g. root-finding, implicit curve generation, and chaos theory.