ICM 1998, Berlin, Aug. 18–27

Abstracts of Plenary and Invited Lectures

Section:

0. Plenary Lectures

1991 MS Classification: 65,65F,65N,65R
Hackbusch, Wolfgang, Universität zu Kiel, Germany

From Classical Numerical Mathematics to Scientific Computing

Many of the fundamental numerical methods trace back to the time of Newton and Gauss. However, the discipline of Numerical Mathematics as a field of its own did not appear before the electronic computer came into use. The computer not only marks the beginning of Numerical Mathematics. For the development of this field it is quite essential that the computer technology is steadily improving. This leads to an increasing complexity of the problems to be solved and to several new questions. As a consequence, Numerical Mathematics has changed a lot over the last 50 years. One sign of these changes is the fact that often the name Scientific Computing is used instead of Numerical Mathematics.

It should be noted that Numerical Mathematics or Scientific Computing combine the following parts which mutually dependent: modelling, algorithms, analysis. Modelling is (a) the appropriate mathematical formulation of a problem from a field outside Mathematics and (b) the discretisation process with translates infinitely dimensional problems into finitely dimensional ones. The heart of Numerical Mathematics are the algorithms enabling the solution process. Numerical Analysis comes into play to judge the consequences of the discretisation process, to control the algorithms (reliability, quality, costs, etc.). As pointed out in the end of the lecture, it now seems to be necessary to add a further subfield: the implementation of numerical methods on the computer.

The lecture will, in particular, illustrate the consequences for the treatment of partial differential equations. Since after the discretisation process, the arising systems of equations can be as large as the computer memory allows, the computer technology obviously has an important influence. It may be self-evident that we would like the algorithms to be as efficient as possible, i. e., they should yield the desired results for lowest computational costs. This vague request can be made more precise. We explain why the development of the computer technology directly leads to the need of algorithms with linear complexity, i. e., the computational work for performing an algorithm with n input data must be proportional to n.

The algorithms satisfying this requirement often show a hierarchical structure. An early example is the Fast Fourier Transform (FFT) which is almost of linear complexity and has a typical recursive structure. Wavelets and algorithms exploiting their features are a more recent example. For the

solution of discrete elliptic partial differential equations, multi-grid methods have been developed and turned out to be very flexible. They essentially make use of a hierarchy of discrete problems.
The search for efficient methods has led to a further principle, the adaptivity. Since memory is the limiting factor, one tries to adapt the discretisation in such a way to the problem that the best accuracy can be obtained. There are cases, where the adaptation to the problem can be designed a prior, but, usually, the adaptation process is done a posteriori, more precisely, during the computational process. Examples will be given, where the refinement and coarsening of finite elements is governed by this principle. Adaptation gives again rise to intertwine the design of algorithms with Numerical Analysis. Furthermore, the discretisation process and the solution process are often combined.
It is not only the speed and capacity of the computer which is improving but also its architecture. We discuss the effects to numerical algorithms. Mathematical structures which have been developed by this reason are various decomposition methods (domain decomposition and subspace decomposition methods). However, it seems that decomposition is more helpful as an analysis tool than for the design of efficient algorithms.
We discuss how the principles "hierarchy", "adaptation" and "decomposition" interact. It can be observed that, because of the increased complexity, a new difficulty arises: The implementation process becomes more and more time consuming and needs better scientific support.
The request of linear complexity is very restrictive and seems to exclude, e. g., the treatment of linear systems with a full matrix, since simple operations like a matrix-vector multiplication then are of quadratic complexity. The lecture is concluded by a discussion of this problem. A positive answer are given in the case of integral equations.