Back-tracking $Ax = b$

Uri M. Ascher

Dept. of Computer Science, Univ. of British Columbia

Vancouver, B.C., V6T 1Z4, Canada

ascher@cs.ubc.ca

A physical problem begets a mathematical model. A mathematical model begets a numerical method for an approximate, computable solution. A numerical method begets large, sparse systems of linear algebraic equations $Ax = b$.

Ideally, one would like to design a solver that automatically and efficiently computes the solution x for any such system. But this ideal may require too much to be feasible in such generality. In some situations, specific features of the underlying discretized problem may be crucial for the design of efficient algorithms for the linear algebra. Furthermore, there are instances where the best approach to design fast algorithms to find x is not to solve $Ax = b$ at all. Rather, start by *back-tracking* away from the algebraic system, to the numerical method that gives rise to it. A reinterpretation of the numerical method may then yield an embedding of $Ax = b$ in a larger linear algebra system which is easier to solve.

We will consider two instances of this process: One is the numerical simulation of multibody systems, such as robot manipulators, where the composite rigid body method is replaced by the articulated body method, yielding a solution algorithm with optimal complexity in the number of bodies. The other instance involves the numerical solution of nodal and mixed finite element methods, where a reformulation in terms of hybrid methods allows the design of highly efficient multilevel preconditioners for the larger system.