Application Brief
Intel Scientific Computers

Solving the Wave Equation on the Intel iPSC™

Cleve Moler & Peter Ross
Intel Scientific Computers

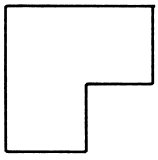
PROBLEM DESCRIPTION

The wave equation provides a mathematical model of a wide range of physical phenomena, including vibrations, resonances, and propagations of disturbances. This document describes one approach to solving a model wave problem using the Intel iPSC™, a hypercube-connected multiprocessor. It is possible to allocate the work to be done among the processors in such a way that the speed of the computation is determined by the speed of the graphics output device.

The differential equation governing waves is:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$

The model problem involves a 2-dimension region, L, formed from three unit squares,



The problem requires $u(x,y,t) = 0$ on the boundary of L. The interest in this particular problem derives from the interior corner, which leads to singularities in the derivatives of u and low accuracy from conventional finite difference and finite element methods.

The mathematical basis of our solution method [1] is an eigenfunction expansion,

$$u(x,y,t) = \sum_{k=1}^{n} a_k \sin(\mu_k t) v_k(x,y)$$

This form of the solution vanishes at $t = 0$, thereby satisfying one initial condition. The second initial condition, a specified initial velocity, determines the coefficients a_k.

The eigenvalues μ_k and eigenfunctions $v_k(x,y)$ are characteristic of the region L. The eigenfunctions can be accurately approximated with an expansion involving polar coordinates (r,θ) and fractional order Bessel functions $J_\alpha(r)$,

$$v_k(x,y) = \sum_{j=1}^{m} c_j^k J_{a_j}(\mu_k r) \sin(a_j \theta)$$

The quantities a_j are taken to be multiples of 2/3, so that the eigenfunctions satisfy the boundary conditions on the two sides of L that form the interior corner and have the correct singular behavior near the corner. The eigenvalues μ_k and the expansion coefficients c_j^k are determined by a least-squares algorithm which specifies that the eigenfunctions nearly vanish on the remainder of the boundary.

---

iPSC is a trademark of Intel Corporation

This approach always produces an analytic function $u(x,y,t)$ which is an exact solution of the wave equation, defined for all values of all its arguments. The only two discretization parameters are the upper summation limits, m and n. They determine the accuracy of the approximations to boundary conditions and initial conditions, respectively.

THE iPSC SOLUTION

The implementation of this method on the Intel iPSC involves four stages and illustrates several ways in which the portions of an algorithm can be allocated among the processors in a multiprocessor system. The output of the demonstration program is a graphical display of a moving projection of the solution.

Stage

1 The Eigenfunctions

The first stage involves only the geometry of the domain L and n of the processors, where n is the number of eigenfunctions in the approximation of the solution. Processor number k computers the eignenvalue μ_k and the coefficients c_j^k

The computation uses subroutines DQRDC and DTRSL from [2] and subroutine FMIN from [3] to find null vectors of rank deficient matrices whose entries involve Bessel functions evaluated on the boundary of L. Each processor saves its individual results for use in the later stages.

2 The Initial Conditions

A particular point in L is specified as the origin of an impulse that determines the initial velocity. Processor k computes the coefficient a_k determined by that impulse. These coefficients are assembled into a vector of coefficients which is passed to all the processors.

3 The Display Grid

Each processor is responsible for a portion of a display grid covering the region. The portions are allocated so that all processors have roughly the same number of display grid points. In the third stage, each of the processors evaluates all n eigenfunctions at each of its grid points. The processors communicate with each other to obtain the required coefficients. Each processor saves the values associated with its display grid points.

4 The Time Steps

The host computer sends a value of t to all processors. Each processor computes a linear combination of the eigenfunction values saved in Stage 3. The coefficients in the linear combination depend upon t. The result is the set of values of the solution $u(x,y,t)$ for the grid points owned by that processor. These results are then assembled into a single list of values for the entire domain which is sent back to the host to be displayed. This fourth stage is repeated for successive values of t to produce the successive frames for the moving display.

References:

[1] L. Fox, P. Henrici and C. Moler, *Approximations and bounds for eigenvalues of elliptic operators*, SIAM J. Num. Anal., pp. 89-102, 1967.

[2] J. J. Dongarra, J. R. Bunch, C. B. Moler and G. W. Stewart, *LINPACK Users' Guide*, SIAM Publications, 368 pp., 1979.

[3] G. E. Forsythe, M. A. Malcolm and C. B. Moler, *Computer Methods for Mathematical Computations*, Prentice-Hall, 288 pp., 1977.