Large Scale Calculations of 3D Elastic Wave Propagation in a Complex Geology

V. Pereyra, E. Richardson

Weidlinger Associates
4410 El Camino Real
Los Altos, CA 94022

S. E. Zarantonello

Computational Research Division
Fujitsu America, Inc.
San Jose, CA 95134-2022

Abstract

We present a large scale finite element simulation of fully elastic seismic waves propagating in a complex three dimensional geological environment.

Our results are noteworthy on three accounts:

1. *The complexity of the synthetic model which involves a folded structure and variable velocity of propagation.*
2. *The size and detail of the computational mesh, with over 13 million elements and 1700 time steps.*
3. *A high computational efficiency and speed of execution.*

A finite element explicit-in-time procedure was used to solve the 3D elastic wave equations. The wave train was generated by a 3 Hz surface pressure source over a simulated time interval of 10 seconds. The simulation itself took 4.5 hours to complete on a Fujitsu VP2400 supercomputer, at a sustained rate of about 0.5 GFLOPS.

1 Introduction

Because of its stringent computational demands, 3D fully-elastic wave modeling is still viewed mostly as a research topic and a challenge for today's most powerful computers [1],[3]. In this article we show that "difficult" problems are feasible with the finite element technique, and can be solved today in reasonable amounts of time with the proper combination of hardware and software.

Our study involves the simulation of a single shot experiment propagating in a $10\,km^3$ cubical domain, with a 3D finite element explicit-in-time procedure. About 13 million linear basis elements and 1700 time steps were used to simulate a 10 second time interval.

Although large 3D elastic wave problems on Earth models have been solved with finite differences [4], [5], to the best of our knowledge the simulation reported in this article is the largest of its type ever made with the finite element technique. Our results were calculated on a Fujitsu VP2400/10 supercomputer [9] using Weidlinger Associates's package INTEGRA [7] for integrated forward and inverse modeling of geological regions.

2 Model Representation

Model specification and construction are important issues in three-dimensional geological simulations.

A volume of Earth is described in INTEGRA as an aggregate of irregular sub-volumes where material properties vary smoothly. The subvolumes, in turn, are separated by material interfaces where discontinuities may occur. Both, interface surfaces and volumetric properties, are represented analytically by bicubic and tricubic tensor products of B-splines, either in explicit or parametric form, as required.

While this representation is directly applicable to seismic ray tracing (a simpler procedure to partially simulate wave propagation [7]), the finite element solution of the elastic wave equations requires that material properties be given on a computational mesh, a "voxel representation" consisting of small regular parallelepipeds. In INTEGRA this transition from geometric description to voxel representation is automated, so the same model construction can be used for both, ray tracing or finite element simulations.

1063-9535/92 $3.00 © 1992 IEEE

3 Finite Element Overview

GeoFlex is a finite element program in the INTEGRA package designed to simulate the seismic response of complex geologic structures. It computes the full elastic solution of waves propagating in an isotropic, linear-elastic medium, producing results in the form of nodal quantities (velocities or displacements) and elemental quantities (material stresses and strains).

GeoFlex has 1D, 2D, and 3D modeling functions. Each of these is driven by its own processor for computational efficiency. As mentioned above, material models are limited to the linear elastic case.

Forward time integration is performed with an explicit integration scheme, which avoids the difficulties of manipulating a large global stiffness matrix for the model. With explicit time integration the size of the computational time step is limited by the Courant stability criterion.

In its most general form, GeoFlex uses isoparametric finite elements to represent a continuum. Two-dimensional models use 4-noded quadrilateral elements, while 3-dimensional models use 8-noded hexahedronal elements. The elemental forces due to internal strain energy are computed with single point integration. Single point integration greatly simplifies the calculations, but adversely allows for unresisted modes of deformation (termed hour-glass modes) which must be supressed during the elemental force computation. As a further simplification of the computional scheme, elemental shapes in GeoFlex are limited to rectangles for 2D and rectangular prisms for 3D.

GeoFlex is written in standard Fortran 77 and has been installed on SUN, SGI, and IBM workstations, VAX and PRIME minicomputers, IBM mainframes, CRAY supercomputers, and most recently, FUJITSU supercomputers. Computational models have ranged in size from a few hundred elements on minicomputers and low-end workstations, to the 13 million element 3D model run on the Fujitsu VP2400.

4 The Finite Element Algorithm

4.1 Equations of Motion

The finite element algorithm is derived from the general equations of motion for a continuum in the form of the Principle of Virtual Work:

$$\int_V \left[\rho \ddot{u}^T \delta u + \sigma^T \delta \epsilon - b^T \delta u\right] dV - \int_S t^T \delta u \, dS = 0 \quad (1)$$

where:

u^T = Displacement Vector
t^T = External Surface Traction Vector
σ^T = Stress Tensor
ϵ^T = Strain Tensor
b^T = Body Force Vector (Gravity)
ρ = Mass Density
δu = Virtual Displacement Vector
$\delta \epsilon$ = Virtual Strain Tensor corresponding to δu
$\cdot$ = Time Derivative
V = Volume of the Continuum
S = Surface of the Continuum

The basis of the finite element method is to transform equation 1 into a system of ordinary differential equations by subdividing the volume, V, into a number of smaller subvolumes, or finite elements, such that the volume integral of equation 1 is now the summation of all the volume integrals of the finite elements. The volume can now be looked at as n subvolumes, V_k, where $k = 1, 2, \ldots, n$. Within each element, the displacement field is represented as a function of the displacements of the nodes that surround the element:

$$u_k = N_k q_k \quad (2)$$

The strain-displacement relations:

$$\epsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \quad (3)$$

(where i, j are the global directions)

are written in terms of equation 2 to produce a relation of strain as a function of nodal displacement at each element k:

$$\epsilon_k = B_k q_k \quad (4)$$

The virtual strain and virtual displacement can be written similarly:

$$\delta u_k = N_k \delta q_k \quad (5)$$

$$\delta \epsilon_k = B_k \delta q_k \quad (6)$$

We assume that the Principle of Virtual Work (equation 1) holds for any arbitrary δq_k. The components of equation 1 can now be written:

$$\begin{aligned} \int_V \rho \ddot{u}^T dV \delta u &= \sum_{k=1}^{n} \int_{V_k} N_k^T \rho N_k \, dV \ddot{q} \delta q \\ &= M \ddot{q} \delta q \end{aligned} \quad (7)$$

$$\int_V \sigma^T \delta\epsilon dV = \sum_{k=1}^{n} \int_{V_k} B_k^T \sigma_k dV \delta q = Q\delta q \quad (8)$$

$$\int_V b^T \delta u dV + \int_S t^T \delta u dS = \sum_{k=1}^{n} \left(\int_{V_k} B_k^T dV + \int_{S_k} N_k^T t_k dS \right) \delta u = P\delta q \quad (9)$$

or:

$$M\ddot{q} + Q = P \quad (10)$$

4.2 Elemental Equations

GeoFlex includes 1D, 2D, and 3D analysis. Here, we will focus on the 3D continuum element which is used for the calculation described in section 5. A trilinear interpolation function is used to represent the displacement field:

$$N_i = \frac{1}{8}(1 + \xi\xi_i)(1 + \eta\eta_i)(1 + \zeta\zeta_i) \quad (11)$$

where: ξ, η, ζ are the natural coordinates of an element, and $i = 1, 2, \ldots, 8$ are the elemental node numbers.

We will use single point quadrature for volume integration, so that the elemental contributions to the equations of motion can be written explicitly. For elemental mass, we use the diagonal, or lumped, mass form:

$$M_k = \frac{1}{8}\rho_k V_k I \quad (12)$$

where V_k is the volume of the k^{th} element and I is the identity matrix.

Elemental forces, as shown in equation 8, are computed in two steps. First, elemental strains are calculated and converted to stresses through some constitutive relationship. The stresses are then used to calculate the elemental forces. We look back at equation 4 to get the relationship between strains and nodal displacements. Since this is a dynamic analysis, we are actually computing nodal velocities rather than displacements, so we use equation 4 to compute a strain increment as follows:

1. **Linearize velocity over a time increment Δt:**

$$\dot{q} = \frac{\partial q}{\partial t} = \frac{\Delta q}{\Delta t} \quad (13)$$

2. **Calculate the strain increment:**

$$\Delta\epsilon_k = \Delta t B_k^T \dot{q}_k \quad (14)$$

We assume x, y, z to be the global coordinate axes and $\Delta x, \Delta y, \Delta z$ to be the lengths of the element sides. Let v_{xi}, v_{yi}, v_{zi} represent the velocities (e.g. $\dot{q}$) at node i. The strain increments can now be written:

$$\Delta\epsilon_{xx} = \frac{\Delta t}{4\Delta x}(\begin{array}{l} -v_{x1} + v_{x2} + v_{x3} - v_{x4} \\ -v_{x5} + v_{x6} + v_{x7} - v_{x8}) \end{array} \quad (15)$$

$$\Delta\epsilon_{yy} = \frac{\Delta t}{4\Delta y}(\begin{array}{l} -v_{y1} - v_{y2} + v_{y3} + v_{y4} \\ -v_{y5} - v_{y6} + v_{y7} + v_{y8}) \end{array} \quad (16)$$

$$\Delta\epsilon_{zz} = \frac{\Delta t}{4\Delta z}(\begin{array}{l} -v_{z1} - v_{z2} + v_{z3} - v_{z4} \\ -v_{z5} + v_{z6} + v_{z7} + v_{z8}) \end{array} \quad (17)$$

$$\Delta\epsilon_{xy} = \frac{\Delta t}{8}[\frac{1}{\Delta y}(\begin{array}{l} -v_{x1} - v_{x2} + v_{x3} + v_{x4} \\ -v_{x5} - v_{x6} + v_{x7} + v_{x8}) \end{array} + \frac{1}{\Delta x}(\begin{array}{l} -v_{y1} + v_{y2} + v_{y3} - v_{y4} \\ -v_{y5} + v_{y6} + v_{y7} - v_{y8})] \end{array} \quad (18)$$

$$\Delta\epsilon_{yz} = \frac{\Delta t}{8}[\frac{1}{\Delta z}(\begin{array}{l} -v_{y1} - v_{y2} - v_{y3} - v_{y4} \\ +v_{y5} + v_{y6} + v_{y7} + v_{y8}) \end{array} + \frac{1}{\Delta y}(\begin{array}{l} -v_{z1} - v_{z2} + v_{z3} + v_{z4} \\ -v_{z5} - v_{z6} + v_{z7} + v_{z8})] \end{array} \quad (19)$$

$$\Delta\epsilon_{xz} = \frac{\Delta t}{8}[\frac{1}{\Delta z}(\begin{array}{l} -v_{x1} - v_{x2} - v_{x3} - v_{x4} \\ +v_{x5} + v_{x6} + v_{x7} + v_{x8}) \end{array} + \frac{1}{\Delta x}(\begin{array}{l} -v_{z1} + v_{z2} + v_{z3} - v_{z4} \\ -v_{z5} + v_{z6} + v_{z7} - v_{z8})] \end{array} \quad (20)$$

Stress increments can be calculated from the strain increments. We use only linear-elastic materials in GeoFlex:

$$\Delta\sigma = D\Delta\epsilon \quad (21)$$

where:

$$D = \begin{bmatrix} D_{11} & D_{12} & D_{13} & 0 & 0 & 0 \\ D_{21} & D_{22} & D_{23} & 0 & 0 & 0 \\ D_{31} & D_{32} & D_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & D_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & D_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & D_{66} \end{bmatrix} \quad (22)$$

$$D_{11} = D_{22} = D_{33} = K + 4/3G \quad (23)$$

$$D_{21} = D_{12} = D_{31} = D_{13} = D_{32} = D_{23} = K - 2/3G \quad (24)$$

and where K is the material bulk modulus and G is the material shear modulus

With increments of stress calculated from equation 21, we are ready to proceed to the calculation of elemental force increments using equation 8. Let $\Delta\sigma_{xx}, \Delta\sigma_{yy}, \ldots$ represent the increments of stress. ΔQ_{ij} is the increment of force in the i-direction at node j. Equation 21, assuming single point integration over a rectangular hexahedron, reduces to:

Elemental Increment of x-Forces

$$\Delta Q_{x1} = \frac{1}{4}(+\Delta\sigma_{xx}\Delta y\Delta z + \Delta\sigma_{xy}\Delta x\Delta z + \Delta\sigma_{xz}\Delta x\Delta y) \quad (25)$$

$$\Delta Q_{x2} = \frac{1}{4}(-\Delta\sigma_{xx}\Delta y\Delta z + \Delta\sigma_{xy}\Delta x\Delta z + \Delta\sigma_{xz}\Delta x\Delta y) \quad (26)$$

$$\Delta Q_{x3} = \frac{1}{4}(-\Delta\sigma_{xx}\Delta y\Delta z - \Delta\sigma_{xy}\Delta x\Delta z + \Delta\sigma_{xz}\Delta x\Delta y) \quad (27)$$

$$\Delta Q_{x4} = \frac{1}{4}(+\Delta\sigma_{xx}\Delta y\Delta z - \Delta\sigma_{xy}\Delta x\Delta z + \Delta\sigma_{xz}\Delta x\Delta y) \quad (28)$$

$$\Delta Q_{x5} = -\Delta Q_{x3} \quad (29)$$

$$\Delta Q_{x6} = -\Delta Q_{x4} \quad (30)$$

$$\Delta Q_{x7} = -\Delta Q_{x1} \quad (31)$$

$$\Delta Q_{x8} = -\Delta Q_{x2} \quad (32)$$

Elemental Increment of y-Forces

$$\Delta Q_{y1} = \frac{1}{4}(+\Delta\sigma_{yy}\Delta x\Delta z + \Delta\sigma_{xy}\Delta y\Delta z + \Delta\sigma_{yz}\Delta x\Delta y) \quad (33)$$

$$\Delta Q_{y2} = \frac{1}{4}(+\Delta\sigma_{yy}\Delta x\Delta z - \Delta\sigma_{xy}\Delta y\Delta z + \Delta\sigma_{yz}\Delta x\Delta y) \quad (34)$$

$$\Delta Q_{y3} = \frac{1}{4}(-\Delta\sigma_{yy}\Delta x\Delta z - \Delta\sigma_{xy}\Delta y\Delta z + \Delta\sigma_{yz}\Delta x\Delta y) \quad (35)$$

$$\Delta Q_{y4} = \frac{1}{4}(-\Delta\sigma_{yy}\Delta x\Delta z + \Delta\sigma_{xy}\Delta y\Delta z + \Delta\sigma_{yz}\Delta x\Delta y) \quad (36)$$

$$\Delta Q_{y5} = -\Delta Q_{y3} \quad (37)$$

$$\Delta Q_{y6} = -\Delta Q_{y4} \quad (38)$$

$$\Delta Q_{y7} = -\Delta Q_{y1} \quad (39)$$

$$\Delta Q_{y8} = -\Delta Q_{y2} \quad (40)$$

Elemental Increment of z-Forces

$$\Delta Q_{z1} = \frac{1}{4}(\Delta\sigma_{zz}\Delta x\Delta y + \Delta\sigma_{xz}\Delta x\Delta z + \Delta\sigma_{yz}\Delta y\Delta z) \quad (41)$$

$$\Delta Q_{z2} = \frac{1}{4}(\Delta\sigma_{zz}\Delta x\Delta y + \Delta\sigma_{xz}\Delta x\Delta z - \Delta\sigma_{yz}\Delta y\Delta z) \quad (42)$$

$$\Delta Q_{z3} = \frac{1}{4}(\Delta\sigma_{zz}\Delta x\Delta y - \Delta\sigma_{xz}\Delta x\Delta z - \Delta\sigma_{yz}\Delta y\Delta z) \quad (43)$$

$$\Delta Q_{z4} = \frac{1}{4}(\Delta\sigma_{zz}\Delta x\Delta y - \Delta\sigma_{xz}\Delta x\Delta z + \Delta\sigma_{yz}\Delta y\Delta z) \quad (44)$$

$$\Delta Q_{z5} = -\Delta Q_{z3} \quad (45)$$

$$\Delta Q_{z6} = -\Delta Q_{z4} \quad (46)$$

$$\Delta Q_{z7} = -\Delta Q_{z1} \quad (47)$$

$$\Delta Q_{z8} = -\Delta Q_{z2} \quad (48)$$

As stated earlier, the use of single point quadrature allows for certain modes of deformation in an element that produce no resistance. These modes, called hourglass modes, can in some cases dominate a solution and produce totally erroneous results. For this reason, we have included in the element force calculation a set of hourglass resisting forces, following the method of Flanagan and Belytschko [2]. Essentially, the hourglass modes of an element are isolated and an artificial stiffness is applied to resist the modes. The contribution of the hourglass resistance is of the form:

$$f_{iI}^{HG} = \frac{1}{\sqrt{8}} H_{i\alpha}\gamma_{\alpha I} \quad (49)$$

where $\gamma_{\alpha I}$ are the hourglass shape vectors, which are a function of element geometry. The term, $H_{i\alpha}$, is the hourglass stiffness and is a function of element geometry, elastic stiffness, and nodal velocities. For our 3D hexahedron element, we calculate an hourglass stiffness term:

$$H = \frac{\kappa\Delta t}{24}\left(K + \frac{4}{3}G\right)\left(\frac{\Delta y\Delta z}{\Delta x} + \frac{\Delta x\Delta z}{\Delta y} + \frac{\Delta x\Delta y}{\Delta z}\right) \quad (50)$$

where κ is an elastic control coefficient. We use a value of 0.01. The hourglass forces per node are as follows:

$$\Delta f_{i1}^{HG} = H(-g_i - h_i - k_i + l_i) \quad (51)$$
$$\Delta f_{i2}^{HG} = H(-g_i + h_i + k_i - l_i) \quad (52)$$
$$\Delta f_{i3}^{HG} = H(+g_i + h_i - k_i + l_i) \quad (53)$$
$$\Delta f_{i4}^{HG} = H(+g_i - h_i + k_i - l_i) \quad (54)$$
$$\Delta f_{i5}^{HG} = H(+g_i + h_i - k_i - l_i) \quad (55)$$
$$\Delta f_{i6}^{HG} = H(+g_i - h_i + k_i + l_i) \quad (56)$$
$$\Delta f_{i7}^{HG} = H(-g_i - h_i - k_i - l_i) \quad (57)$$
$$\Delta f_{i8}^{HG} = H(-g_i + h_i + k_i + l_i) \quad (58)$$

where i is the direction x, y, or z, and:

$$g_i = v_{i1} + v_{i2} - v_{i3} - v_{i4} - v_{i5} - v_{i6} + v_{i7} + v_{i8} \quad (59)$$
$$h_i = v_{i1} - v_{i2} - v_{i3} + v_{i4} - v_{i5} + v_{i6} + v_{i7} - v_{i8} \quad (60)$$
$$k_i = v_{i1} - v_{i2} + v_{i3} - v_{i4} + v_{i5} - v_{i6} + v_{i7} - v_{i8} \quad (61)$$
$$l_i = -v_{i1} + v_{i2} - v_{i3} + v_{i4} + v_{i5} - v_{i6} + v_{i7} - v_{i8} \quad (62)$$

with v_{ij} the velocity (e.g. $\dot{q}$) in the i-direction at node j.

4.3 Computational Sequence

In GeoFlex, several quantities are computed and stored prior to the solution of the equations of motion. After the geometry and grid have been defined, the elemental mass matrices are calculated using equation 12 and assembled into a global lumped mass vector M. Also, elemental spacings $\Delta x, \Delta y, \Delta z$ and volumes, V_k, are precalculated and saved. We are assuming a condition of small deformations and small rotations, so the mass and geometry specific terms will remain unchanged during the calculation.

GeoFlex uses an explicit, central difference procedure for solving the equations of motion. The following steps describe the sequence for advancing one time increment, Δt. Here, the superscript, n, denotes the time $t = n \cdot \Delta t$. The subscript, i, refers to node number i.

1. **The quantities $\dot{q}_i^n$ and $F_i^{n-1/2}$ are known at time $t = n \cdot \Delta t$.**
2. **Loop through all elements in the grid.**
   (a) Calculate strain increments, $\Delta\epsilon^n$, from equations 15-20
   (b) Calculate stress increments, $\Delta\sigma^n$, from equation 21
   (c) Calculate internal force contribution to nodal forces from equations 25-48
   (d) Calculate hourglass resisting forces from equations 49-62
3. **Apply external nodal forces P_i.**
4. **Loop through all nodes in the grid.**
   (a) Update nodal forces:
   $$F_i^{n+1/2} = F_i^{n-1/2} + \Delta Q_i^n + \Delta f_i^{HG^n} + \Delta P_i$$
   where ΔQ and Δf^{HG} were calculated in step 2, and ΔP was calculated in step 3.
   (b) Update nodal velocities:
   $$\dot{q}_i^{n+1} = \dot{q}_i^n + \frac{F_i^{n+1/2}}{M_i}\Delta t$$
5. **Apply velocity boundary conditions to the grid, possibly modifying $\dot{q}_i^{n+1}$ calculated in step 4.**
6. **Go to step 1 to begin the next time step.**

5 The Large Scale Calculation

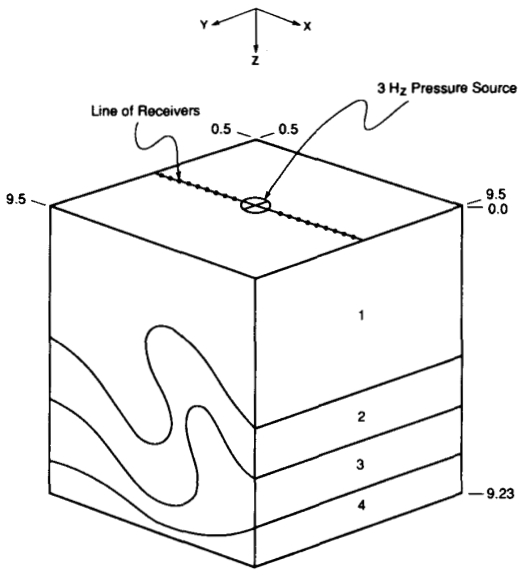
5.1 Model Description

The geologic model selected for this study is that of a recumbent fold. A diagram of the geology is shown in figure 1. Within the bounds of the model, there are four material types, or regions, that are defined by their elastic wave speeds and mass density. Note from figure 1 that the wave speeds, c_p, c_s, and density, ρ, vary linearly with depth in region 1.

The model definition for GeoFlex, for this type of geologic model, consists of a grid definition and three arrays containing a value for c_p, c_s, and ρ for each element in the grid. The grid definition is simply the overall dimensions of the model in the x,y,z directions and the number of elements in each direction, since we are using constant grid spacing in all directions. The elemental quantities, c_p, c_s, and ρ are derived from a ray-tracing model [7] and completely define the elastic constants of the element:

$$K = \rho\left(c_p^2 - 4/3\, c_s^2\right)$$
$$G = \rho c_s^2$$

where K and G are the material bulk and shear moduli.



Region	cp	cs	ρ
1	2 + 0.4z	1.2 + .24z	2 + 0.4z
2	3.5	2.1	2.3
3	4.5	2.1	2.3
4	5.5	2.1	2.3

Figure 1. Diagram of Fold finite element model.

The grid chosen for this study had dimensions 234 x 234 x 240 representing a total of 13,141,440 elements with 13,309,225 nodes. The maximum grid resolution was controlled by the available memory on the Fujitsu VP 2400. GeoFlex requires approximately seven 32-bit floating point numbers per grid node and three floating point numbers per element for this type of calculation. We had slightly over 800 MBytes of vector memory available.

Our objective was to propagate both dilatational and shear waves from a point pressure source on the top center of the grid. To properly model the propagation of a 3 Hz signal through the slowest material we required a minimum of 10 elements per wave length.

The 3 Hz pressure source was a Ricker wavelet produced by:

$$p = Ae^{-\left(2\pi f \frac{t}{4}\right)^2} \cos(2\pi f t)$$

where p is the resulting pressure, A is the peak amplitude, f is the frequency in Hz, and t is time. The pressure was applied with a Gaussian spatial decay about a source point on the grid.

A simple transmitting boundary was used on all external surfaces of the grid, except for the surface on which the pressure was applied, which was left free.

A cut-away contour plot of the material wavespeeds is shown in figure 2.

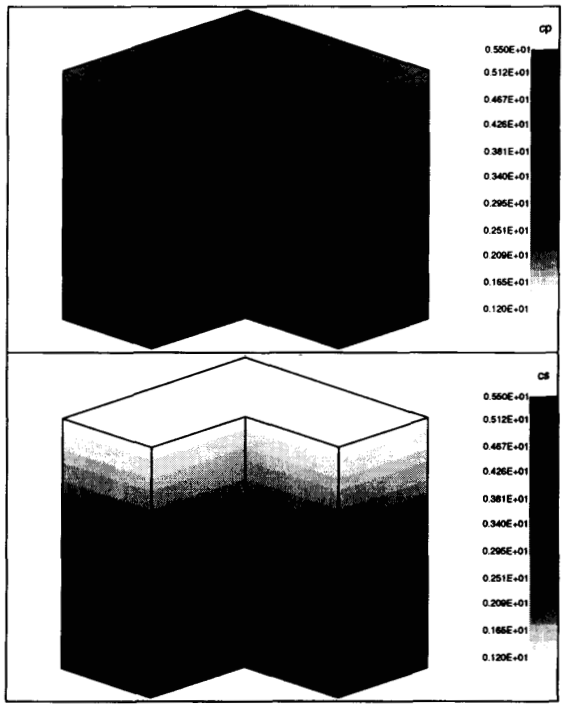


Figure 2. Fold model grid showing c_p and c_s.

(For color plate see page 845)

The top picture represents c_p, the p-wave speed, while the bottom picture shows c_s, the s-wave speed. The grid depicted is coarser than the computational grid used in the calculation on the Fujitsu VP2400, due to memory limitations on the graphics workstation used to produce these illustrations.

5.2 Calculation Results

A total of 1700 time steps were computed for the Fold model on the Fujitsu VP2400. Using a time increment of 0.006 seconds, a total simulation time of 10.2 seconds was achieved.

Figure 3 is a synthetic seismogram produced by plotting the computed nodal velocities along the line of surface receivers shown in figure 1.

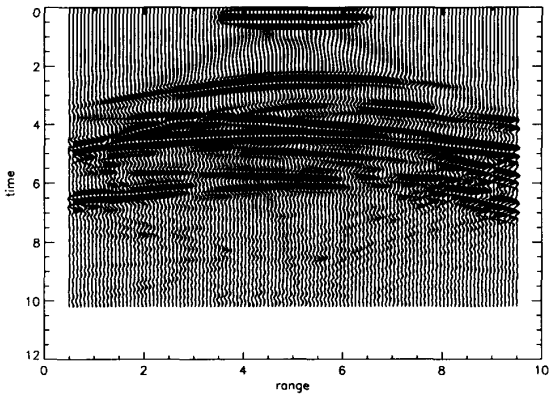


Figure 3. Fold model seismogram.

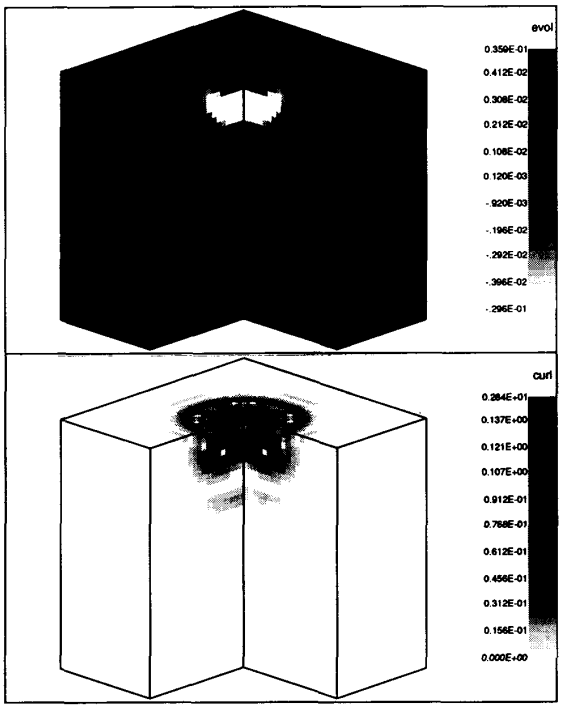


Figure 4. Fold model; time=1.5 seconds.

For visualization purposes, two extra quantities were computed along with nodal velocities: volumetric strain and the magnitude of the curl of the velocities. Figures 4, 5, and 6 are cut-away contour plots of the grid displaying volumetric strain and curl amplitude at 3 times during the calculation. In the top picture of each figure the volumetric strain indicates the propagating p-wave. In the lower picture the curl amplitude shows the propagating s-wave.

The total computation time for 1700 time steps was 4.5 CPU-hours, equivalent to a rate of about 1.4 x 10^6 elements processed per second. At each time step GeoFlex calculated 335 floating point operations per 3D element in the force update, and 3 floating point operations per node in the velocity update. There was also boundary processing done at each time step, but these calculations were of minor significance compared to the element force calculations. The sustained rate of floating point performance was about 0.5 GFLOPS.

The calculations were run in-core, with negligible I/O overhead. As already mentioned, 800 MBytes of main memory were required for these computations.

(For color plates see page 845)

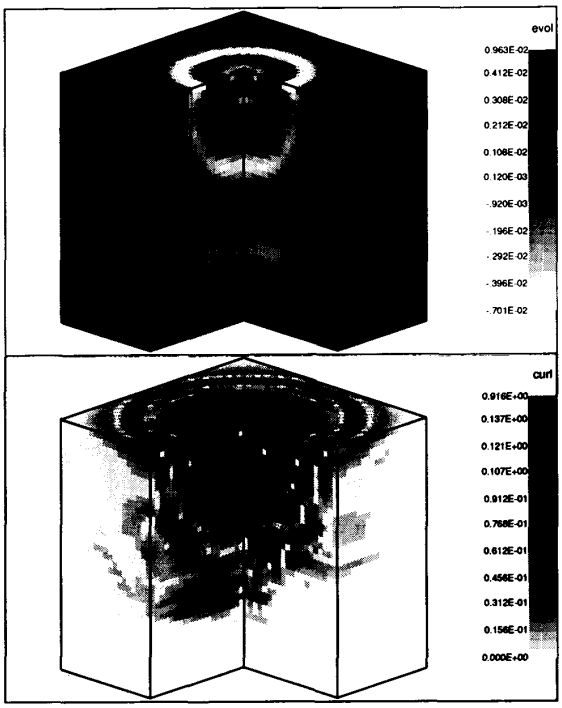


Figure 5. Fold model; time=3.0 seconds.

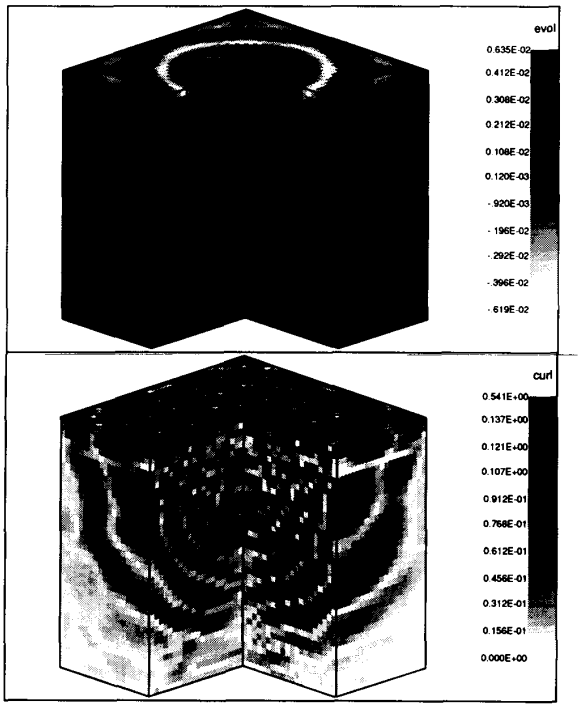


Figure 6. Fold model; time=5.25 seconds.

(For color plate see page 845)

6 The Fujitsu VP2400/10

The machine used in this study, the VP2400/10 at Fujitsu America's Open Systems Center, is a mid-range uniprocessor in the VP2000 series, Fujitsu's second generation of supercomputer systems [9].

The characteristics of the VP2400/10 model are as follows:

1. The CPU consists of a *main memory unit*, a *channel processor*, and a *processing element* comprising two distinct entities: one *scalar unit* and one *vector unit* which can run simultaneously. The vector unit, in turn, includes an ensemble of vector registers and replicated pipelines (each delivering two results per vector clock cycle).

2. The main main memory unit contains up to 128 MWords (or 1 GByte) of 512-way *interleaved* memory composed of 1 Mbit static RAM chips with 35 ns access time. This main memory is backed up by up to 32 GBytes of additional solid-state memory referred to as the *system storage unit*.

All vector floating-point calculations are performed on 64 bit words. However, when running in *single-precision* mode (as in this study) the floating point operands are held in memory as 32 bit words, and expanded or truncated when fetched or stored from memory.

The configuration of the particular VP2400/10 system used in our study are listed in Tables 1 and 2. The operating environment consisted of Fujitsu's implementation of UNIX System V, Release 4. An advanced Fortran vectorizing compiler was used.

Table 1: General specifications.

Operating System	UXP/M + VPO
Compiler	Fortran77 EX/VP
Scalar Clock Cycle	8.0 ns
Vector Clock Cycle	4.0 ns
Peak Performance	2 GFLOPS
Main Memory	1 GBytes
Memory Bandwidth	8 GBytes/s
I/O Bandwidth	1 GBytes/s

Table 2: Vector unit specifications.

Load-Store Pipelines	2
Multiply-Add Pipelines	2
Divide Pipelines	1
Mask Pipelines	2
Vector Registers	64 KBytes
Mask Registers	1024 Bytes

Out of seven pipelines, six can operate simultaneously: two load-store, two mask, and two arithmetic pipelines. As mentioned above, these pipelines are in essence double units. For example, the load-store pipelines can deliver two 64 bit words per vector unit clock cycle each. The multiply-add pipelines can execute logical, vector add, vector multiply, or compound multiply-add instructions. One multiply-add pipeline, therefore, calculates up to two "multiply-adds" per vector clock cycle, when it executes the compound multiply-add instruction on contiguous data elements.

Peak performance on the VP2400/10 is approached when the calculations are load-balanced, and when two multiply-add pipelines concurrently execute compound multiply-add instructions (thus delivering 8 floating-point results per vector unit clock cycle). This is the case when computing convolutions, solving linear equations, or performing standard matrix-matrix or matrix-vector operations.

7 Conclusions

To the best of our knowledge, the elastic wave simulation reported in this study is the largest ever made using a 3D finite element technique for a complex geological region. It is certainly the largest ever attempted with GeoFlex.

As more powerful computers become available more complex simulations will become possible, and elastic wave modeling will become more widely used in industry [8]. Our study demonstrates the feasibility of modeling one-shot events in complex geologies, and provides a benchmark for estimating the computational cost in the pre-stack simulation of full seismic surveys that usually involve many shots.

Acknowledgements

We wish to thank Dr. Kenichi Miura of Fujitsu America Inc. for his support, and the technical staffs of Weidlinger Associates and Fujitsu America's Open Systems Center for their assistance.

References

[1] S. W. Fagin, *Seismic Modeling of Geologic Structures*, Geophysical Development Series, Society of Exploration Geophysicists. 1991.

[2] D. P. Flanagan and T. Belytschko, *A Uniform Strain Hexahedron and Quadralateral with Orthogonal Hourglass Control*, International Journal for Numerical Methods in Engineering, Vol. 17, pp. 679-706, 1981.

[3] O. G. Johnson and E. L. Leiss, *Computational Techniques of Exploration Geophysics*, Supercomputer Review, pp. 34-44, September 1990.

[4] P. Mora, *Modeling Three-Dimensional Seismic Waves*, Proceedings of the "Geophysics and Computers - A Look at the Future" workshop, Houston Area Research Center, pp. 139-163, April 1989.

[5] I. R. Mufti, *Large-Scale Three-Dimensional Seismic Models and their Interpretive Significance*, Geophysics, Vol. 55, No. 9, September 1990.

[6] V. Pereyra, *Two point ray tracing in general 3D media*, To appear in Geophysical Prospecting, 1992.

[7] V. Pereyra, *Weidlinger Associates Inversion Project, Vol. IV*, Weidlinger Associates, Los Altos, CA, December 1991.

[8] E. Piault, A. Sei, P. Duclos, *GigaFLOPS Performance for 2D/3D Heterogeneous Acoustic Modeling on a Massively Parallel Computer*, Proceedings of the First International Conference on Mathematical and Numerical Aspects of Wave Propagation Phenomena. Edited by G. Cohen,L. Halpern, P. Joly. SIAM, pp. 136-144, April 1991.

[9] N. Uchida, *Fujitsu VP2000 Series Supercomputers*, International Journal of High Speed Computing, Vol. 3, Nos. 3 and 4, pp. 169-185,September and December 1991.