NASA Contractor Report 178158

ICASE REPORT NO. 86-51

ICASE

A THREE DIMENSIONAL CALCULATION OF ELASTIC EQUILIBRIUM FOR COMPOSITE MATERIALS

Liviu R. Lustman

Milton E. Rose

Contract Nos. NAS1-17070, NAS1-18107

July 1986

INSTITUTE FOR COMPUTER APPLICATIONS IN SCIENCE AND ENGINEERING
NASA Langley Research Center, Hampton, Virginia 23665

Operated by the Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

A THREE DIMENSIONAL CALCULATION OF ELASTIC EQUILIBRIUM FOR COMPOSITE MATERIALS

Liviu R. Lustman

Institute for Computer Applications in Science and Engineering

Milton E. Rose

Old Dominion University

ABSTRACT

A compact scheme is applied to three dimensional elasticity problems for composite materials, involving simple geometries. The mathematical aspects of this approach are discussed, in particular the iteration method. A vector processor code implementing the compact scheme is presented, and several numerical experiments are summarized.

Research was supported by the National Aeronautics and Space Administration under NASA contracts NAS1-17070 and NAS1-18107, while the authors were in residence at ICASE, NASA Langley Research Center, Hampton, VA 23666-5225.

INTRODUCTION

This paper presents numerical solutions of static elasticity problems, using a compact finite element scheme. Its main aim is to illustrate the various computational questions raised by such a technique, as well as to test it by a rather complex problem : three dimensional edge stress equilibrium of a composite material for simple geometries.

As described in [5], the compact scheme expresses, to second order accuracy, the relationship between the average values of the displacements and the normal tractions on the faces of each volume element which is in isolated equilibrium. Global equilibrium obtains when the displacements are continuous and the traction forces balance across element interfaces. For brick elements standard finite difference notations can be used and result in the compact finite difference procedure used here.

The same algorithm has been presented in [1], where it was applied to two dimensional problems in rectangular domains. Its mathematical properties - a variational formulation resulting in symmetry and natural boundary conditions - are even more important and advantageous in the full three dimensional context. Another interesting feature is the parallelism inherent in this method.

Several engineering codes are available for treating laminates. These usually average material properties across the layers and then perform two dimensional computations - even though the typical behavior under strong stress is delamination, showing that large displacements and stresses must occur between the layers. Sometimes other simplifying *ad hoc* assumptions are made - see section 4. Thus it seemed important to perform a three dimensional calculation based on the basic elasticity equations with no simplifications at all, if only to provide a check for the standard, thin plate methods of solution. Even the simple problems considered here involve $10^4 - 10^5$ variables, and a central part of the study concerns iterative methods for their solution.

The physical problem itself presents serious difficulties because of the singularities present at layer interfaces. Reference [2] summarizes several calculations which do not agree, even qualitatively, on the behavior near the singularity. The compact scheme has the peculiarity of never using the singular point itself, since only average values on element faces are involved. Although there was no special singularity treatment, our results match reasonably well those of [2], where mesh refinement was used. Further research is necessary to clarify the behavior of the compact scheme near singularities.

1. THE COMPACT SCHEME

The numerical discretization that we employ is based on subdivision of the computational domain into brick cells of size $\Delta x_1 \times \Delta x_2 \times \Delta x_3$. The variables, in our case the displacements u and stresses τ, are associated with the cell faces. These quantities are the unknowns in the discrete problem, where they represent averages over the faces of the continuous values. For the time being, numerical stresses and displacements are taken to be independent, although their differential analogs are, of course, connected by the static elasticity equations:

$$
\begin{aligned}
a. \quad & \sum_j \frac{\partial \tau_{ij}}{\partial x_j} = 0 \\
b. \quad & \tau_{ij} = \sum_{k,l} c_{ijkl}\, \varepsilon_{kl} \\
c. \quad & \varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \\
d. \quad & \omega_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right) \quad ,
\end{aligned}
\tag{1.1}
$$

The tensor ε_{ij} is the strain and formula (1.1b) is the stress-strain relation, involving the stiffness coefficients of the material, c_{ijkl}. The antisymmetric (rotational) tensor ω will be needed in the sequel.

In order to approximate the differential equation (1.1), we introduce the differencing and averaging operators (defined on the numerical variables):

$$
\begin{aligned}
(\delta_1 w)_{klm} &= \frac{w^1{}_{klm} - w^1{}_{k-1lm}}{\Delta x_1} \\
(\mu_1 w)_{klm} &= \frac{w^1{}_{klm} + w^1{}_{k-1lm}}{2}
\end{aligned}
\tag{1.2}
$$

where

$$
w^1{}_{klm} = w\left(\left(k+\frac{1}{2}\right)\Delta x_1 \,,\, l\Delta x_2 \,,\, m\Delta x_3\right)
$$

i.e. the variable defined on the face $x_1 = const$ of the klm-cell. Similar definitions hold for indices 2, 3. The results of these operators may be assigned to the center of the cell $(k\Delta x_1 \,,\, l\Delta x_2 \,,\, m\Delta x_3)$, and then they will be second order approximations to the partial derivative and evaluation operators, respectively.

We then replace (1.1) by the following equations, containing two parameters κ and γ (see [1]):

$$a. \quad \delta_1\tau_{1j} + \delta_2\tau_{2j} + \delta_3\tau_{3j} = 0$$

$$b. \quad \mu_i\tau_{ij} = \tau_{ij}(u) + \gamma^2\Delta x_i^2\omega_{ij}(u) \tag{1.3}$$

$$c. \quad \mu_i u_j - \kappa^2\Delta x_i^2\delta_i\tau_{ij} = \lambda_j$$

In the above system, j runs from 1 to 3, and there is no summation on i. The expressions $\tau_{ij}(u)$ and $\omega_{ij}(u)$ are computed from the discrete variables according to formulas (1.2c-d), using the divided difference δ instead of the derivatives.

The vector λ is an average of u in the cell, determined in such a way that (1.3a) is satisfied:

$$\lambda = \frac{\sum_{i=3}^{3}\rho_i\mu_i u}{\sum_{i=3}^{3}\rho_i} \tag{1.4}$$

where

$$\rho_i = (\kappa\Delta x_i)^{-2}. \tag{1.5}$$

It is clear that (1.3) is consistent to second order with (1.1). In particular, equations (1.3b-c) require the equality of certain averages, all of which are within $O(\Delta x^2)$ of the continuous values at the center of the cell. In [1] discrete energy estimates were used to show that the scheme converges as $\Delta x \to 0$ for any fixed values of the parameters, satisfying $\gamma \geq 0$, $\kappa > 0$ (specific values result when the general construction indicated in [5] is used). The choice of $\gamma \neq 0$ guarantees that a vanishing τ implies a constant displacement. Any positive κ allows one to eliminate the numerical stresses in terms of displacements:

$$\mu_i\tau_{ij} = \tau_{ij}(u) + \gamma^2\Delta x_i^2\omega_{ij}(u)$$

$$\delta_i\tau_{ij} = \rho_i\,(\mu_i u_j - \lambda_j(u)) \tag{1.6}$$

Since the formulas above define the sum and difference of τ_{ij} on the two faces $x_i = const$ of a cell, the values of τ itself can be found:

$$\tau_{ij}^{\pm} = \mu_i\tau_{ij} \pm \frac{\Delta x_i}{2}\delta_i\tau_{ij} \tag{1.7}$$

Using (1.6), $\tau_{ij}^{\pm}$ is thus determined by the cell displacement values when the cell is in isolated equilibrium.

The static elasticity equations can be deduced from a variational principle; as shown in [1] and [5], there is a discrete version of this principle stating that on every face the forces from its two neighboring cells must balance. For instance, in the x direction:

$$\textit{residual force} = \Delta x_2 \Delta x_3 \left((\tau_{1j}^{+})_{klm} - (\tau_{1j}^{-})_{k+1lm} \right) = 0 \qquad (1.8)$$

The surface element $\Delta x_2 \Delta x_3$ is needed to convert the tractions - which have the physical dimension of pressure - into forces.

This equation and its counterparts in the y and z directions form the final discrete system to be solved, and the only unknowns are the numerical displacements. Being an Euler-Lagrange equation , formula (1.8) guarantees that the residuals are produced by a symmetric operator; because of uniqueness, this operator is also definite. This turns out to be very advantageous in the iterative solution of our equations, which is discussed in the next section.

Consider now boundary conditions; if displacements are specified, this just reduces the number of unknowns. If tractions are specified on part of the boundary, $x=x_{\max}$ or $x=x_{\min}$, say, the corresponding equations will be

$$a. \quad (\tau_{1j}^{-})_{k_{\min} lm} = \textit{given}$$

$$b. \quad (\tau_{1j}^{+})_{k_{\max} lm} = \textit{given} \qquad (1.9)$$

There are obvious extensions to y or z directions.

These formulas are quite natural and computationally convenient. The stresses in every cell depend only on the displacements of that cell. This is even more important for inhomogenous materials, where the stiffness coefficients vary from point to point - we only need assume continuity in each cell separately. To emphasize this point, consider a finite difference scheme under the same assumption that c_{ijkl} may change abruptly from cell to cell, as it does in a laminate. It is easy to write the equilibrium relations in terms of displacements and stresses defined on a mesh $(i\Delta x\ ,\ j\Delta y\ ,\ k\Delta z)$, but a difficulty is immediately apparent when the constitutive relations (1.1b) are discretized. Because of the discontinuity of c_{ijkl} , one cannot use more than one cell, so simple central divided differences, for instance, cannot be used. If we circumvent this problem by employing staggered grids, putting displacements at $(i\Delta x\ ,\ j\Delta y\ ,\ k\Delta z)$ and stresses at $((i+\frac{1}{2})\Delta x\ ,\ (j+\frac{1}{2})\Delta y\ ,\ (k+\frac{1}{2})\Delta z)$, we find it impossible to specify both displacements and stresses on the same coordinate plane. For instance, we cannot simply describe frictionless sliding over a plane (zero normal displacement and tangent traction).

Yet another attractive property of the compact scheme is its large degree of parallelism. One may divide the variables into three subsets, according to the cell faces on which they are defined. Then it is seen that a numerical x-derivative depends only on the variables located on $x = const$ faces, etc., so one may compute the three partial differences with respect to x, y and z in parallel. The evaluation of ε_{ij}, τ_{ij} combines, however, all the

partial differences, see (1.1b-c), and the cell average λ depends on the variables on all the six faces. Still, once ε, τ and λ are ready, the residuals given by (1.8) may again be computed in parallel. This procedure is similar to the division of variables into two subsets in the familiar "red-black" relaxation scheme, with the only difference that we have three subsets, on the x, y and z faces and a "three-color" scheme for our problem.

The very simple box geometry of our domains results in a particularly straightforward elimination of stresses in terms of displacements; however, as pointed out in [5], the compact scheme is a nonconforming finite element method and the formulas (1.3) may be generalized to apply to almost arbitrary cells. Any degree of regularity in the subdivision of the domain into cells may be used to simplify the solution procedure, in the sense that a fixed finite difference stencil of neighboring points is simpler to use than the arbitrary and varying connection of a finite element to its neighbors.

2. ITERATIVE METHODS

The compact scheme discretization results in a system of equations for the displacements in each cell. Because the independent variable domain is three dimensional and the fact that the displacements themselves are three dimensional vectors, one has to deal with quite a sizable number of unknowns. Even a small debug problem with 3×3×3 cells produces 384 variables: 27 cells with 6 faces each, and on every face 3 displacement components - most faces, however, are common to two cells . It is clear that a direct solution of the algebraic system for general three dimensional bodies is out of question, as it would require enormous memory. On the other hand, the sparse, symmetrical positive definite problem lends itself naturally to iterative solution methods.

In view of the potentially large number of variables, as well as future applications to nonlinear problems, we decided to use a vector processor algorithm, in order to exploit the machine speedup on long vector computations. This, of course, complicates the use of the simplest iterative technique - point Gauss-Seidel, with possible overrelaxation. One rather has to use a residual which has been obtained at all the points simultaneously, by vector operations. In fact, each of the residuals on x,y or z faces may be computed independently of the others (or all of them in parallel) and it is possible to update the variables on these faces separately. This procedure saves memory, since only one third of the full residual is stored. We performed some trial runs of this type, and also some in which the residual computation and updating were done on all the "three-colored" variables at once. It appears that the rate of convergence is significantly improved by the second approach and consequently we compute and store in our program the residual on all three faces, and update all the unknowns together. Finally, from the variety of schemes which use a global residual, we tested two methods: minimal residual Richardson iteration and conjugate gradient.

The Richardson algorithm for the system $A\,x = b$ is:

$$x \leftarrow \textit{arbitrary} \;;\; r \leftarrow b - A\,x \tag{2.1}$$

repeat the following steps until the residual r *is small enough* :

$$w \leftarrow A\,r \;;\; \alpha \leftarrow \frac{(r,w)}{(w,w)} \;;\; x \leftarrow x + \alpha\,r \;;\; r \leftarrow r - A\,w$$

Provided that the symmetric part of A is positive or negative definite, this algorithm reduces the residual norm at every step, and is therefore convergent.

The conjugate gradient method, as defined below, converges to the exact solution for n unknowns in n steps or less, whenever the matrix A is symmetric and definite. Usually one performs a much smaller number of iterations, typically of the order of $\sqrt{n}$.

$$x \leftarrow \textit{arbitrary} \;;\; r \leftarrow b - A\,x \;;\; \rho \leftarrow (r,r) \;;\; p \leftarrow r \tag{2.2}$$

repeat the following steps until ρ, *the residual norm , is small enough* :

$$w \leftarrow A\,p \;;\; \alpha \leftarrow \frac{\rho}{(p,w)} \;;\; x \leftarrow x + \alpha\,p \;;\; r \leftarrow r - \alpha\,w$$

$$\sigma \leftarrow \rho \;;\; \rho \leftarrow (r,r) \;;\; p \leftarrow r + \frac{\rho}{\sigma} p$$

To implement both algorithms one needs a routine which computes the residual Ax (or the inhomogeneous residual $b - Ax$ needed in the first step). Comparison of (2.1) and (2.2) shows that the conjugate gradient method requires an additional buffer p; apart from that there is little difference in the computational work, as the most time consuming task, the residual evaluation, is performed just once per iteration, in both procedures. As mentioned before, the residuals given by (1.8) imply a positive definite matrix A, thus ensuring the convergence of our iterations.

As expected, (see reference [4]) the conjugate gradient method is faster than the Richardson iteration, in fact very much so. Except on small problems, there was an order of magnitude difference between the number of steps in the Richardson and the conjugate gradient runs to reduce the initial residual by the same factor. Thus, after some preliminary testing, we only used conjugate gradient iterations.

It is possible, in principle, to further accelerate convergence by preconditioning. There are, however serious limitation imposed by the technicalities of the vector processor. Unless one uses a very special ordering of the variables in the computer memory, the preconditioner itself must operate on each variable separately, and cannot connect neighboring points, nor mix displacements in different directions at the same point. In other words, it must be a diagonal matrix (as opposed to a banded matrix). The simplest such preconditioner, using the diagonal elements of A did not produce any remarkable speedup. This probably happens because a residual in one component of the displacement is strongly influenced by the other components.

3. EXACT ISOTROPIC SOLUTIONS

As a first test of the numerical method presented in the previous sections, we computed the static equilibrium of a box-shaped, isotropic solid. On this body we imposed boundary conditions: on one side the displacements were specified, while on the other faces the stresses were given. These data were taken from a standard exact solution of isotropic elasticity, so that the accuracy of the computational results could be ascertained.

The solution is defined as follows:

$$\vec{u} = \nabla^2\vec{f} + \frac{1}{2\nu - 2}\nabla\nabla\cdot\vec{f} \quad , \tag{3.1}$$

where ν is the Poisson ratio of the material, and $\vec{f}$ solves the biharmonic equation:

$$\nabla^2\nabla^2\vec{f} = 0 \quad . \tag{3.2}$$

We employed a vector $\vec{f}$ with only one non-vanishing component:

$$f_1 = \mathrm{Re}\left(e^{a(x+iy)}\left(\frac{x}{2a} + bz\right) \right) \tag{3.3}$$

The results of these calculations are summarized in the Table 1 below:

Subdivisions on each side	Number of variables	Iterations to convergence	Relative error in displacements	Relative error in stresses
4	720	549	3.08-2	1.29-2
8	5184	1338	7.75-3	3.18-3
16	39168	2704	1.47-3	9.12-4

Table 1. Exact solution tests

All runs were performed on a cube of side 0.8, using $\Delta x = \Delta y = \Delta z$, $\kappa = 0.1$, $\gamma = 0$. Initially, all the variables were set to zero, with the exception of boundary values. The number of iterations required to reduce the original residual by 10^8 is seen to increase at a much slower rate than the number of variables, although the multiplier of $\sqrt{n}$ is rather large. It should be mentioned that all these runs are quite reasonable in terms of memory and time on the CDC VPS32 computer at NASA Langley. It is clear that the method is second order in both displacements and stresses; halving Δx reduces the error by approximately four. The errors are computed in l_2 norms, all components together.

In view of these results, we could confidently proceed to the more complex problem of a layered material to be presented next. Here we remark that the boundary conditions of this section are of the same type as those for a clamped rectangular laminate plate: at one end no displacement, at the opposite end prescribed traction or compression, the other faces stress-free. Thus we could use the same program setup for both problems.

4. THE LAMINATE RECTANGULAR PLATE

We consider a laminate of 4 layers of graphite/epoxy combination. Each such layer, when oriented with the fibers along the x axis may be taken as an orthotropic material with material constants

$$E_{11} = 20\times10^6\ psi$$

$$E_{22} = E_{33} = 2.1\times10^6\ psi$$

$$G_{12} = G_{23} = G_{13} = 0.85\times10^6\ psi \qquad (4.1)$$

$$\nu_{12} = \nu_{23} = \nu_{13} = 0.21$$

Here E_{ii} is Young's modulus in the direction i, G_{ij} is the shear modulus, and ν_{ij} is the Poisson ratio. In order to strengthen the laminate, the layers are glued with the fibers inclined to the x -axis; the angles for our material are 45° , –45° , –45° , 45°, in this order. The form of the stress-strain relation (1.1b) is given in the Appendix.

A rectangular block of this laminate is shown schematically in figure 1. It is clamped at $x = x_{min}$ and is under a constant traction at $x = x_{max}$; the other faces are stress-free.

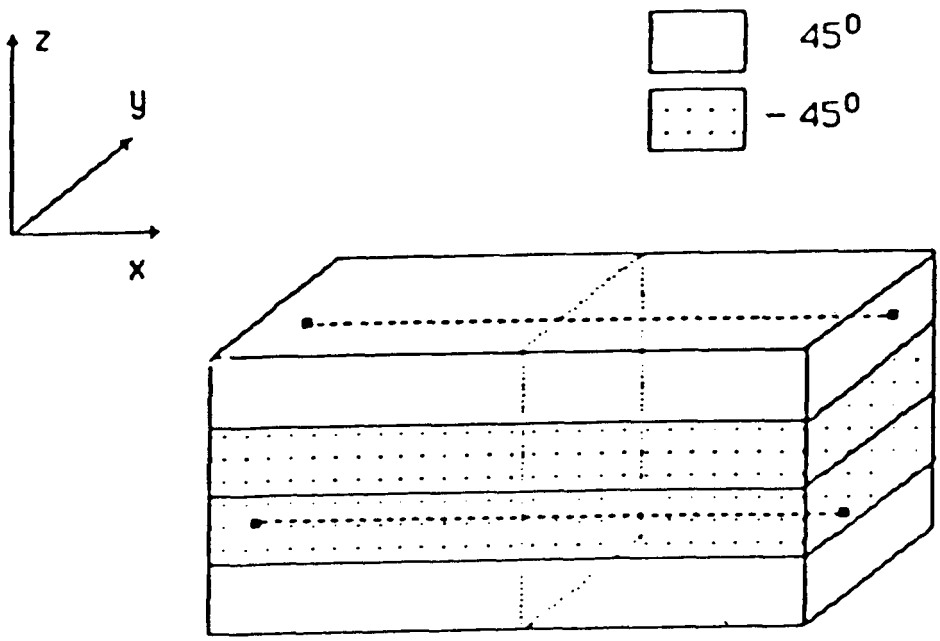


Figure 1. A rectangular laminate block (not to scale) .
Also shown the lines $y,z = const$ and the plane $x = const$
which are used in Figures 4 - 12.

A similar problem has been solved in [2], under the following special assumption (based on engineering experience):

$$u_1(x,y,z) = \varepsilon x + U(y,z)$$
$$u_2(x,y,z) = V(y,z) \qquad (4.2)$$
$$u_3(x,y,z) = W(y,z)$$

This means that the edge traction has been replaced by a constant strain $\varepsilon_{11} = \varepsilon$ throughout the body. The resulting equations have only two independent variables y and z and can be treated by standard two dimensional methods. Stress-free conditions are imposed on the boundary of the y,z domain. It is this solution that will serve as a benchmark for our calculations (there are, of course, no known exact solutions for composite materials).

We need not make any special ansatz like (4.2) but directly proceed to solve the problem (1.3) ; obviously, we have to pay the price of dealing with three independent variables. The discretization uses 10×20×32 uniform cells, with $\Delta x = \Delta y = 0.1$, and $\Delta z = 0.005$. This represents a fairly thin plate, of dimensions 1×2×0.16 and every layer is divided into 8 Δz subintervals.

The first question to be settled is how appropriate is formula (4.2) itself. By plotting u_1 as a function of x - see figures 4 and 5, we found graphs which look almost straight in certain regions away from the clamped edge; there (4.2) is a reasonable approximation. However, the slopes of the linear part of these graphs varies with y and z ; therefore, the agreement between our results and the results of [2] is only qualitative.

Figures 6 to 11 show plots of the various components of the stress in an $x = const.$ plane. The 45°/–45° interfaces are clearly discernible, and so is the singularity on the interface at $y = y_{min}$ and $y = y_{max}$. In fact, as we follow the interface towards the body surface, τ stays almost constant, changing abruptly only near the singularity. The same behavior has been reported in [2]. The calculated values $\tau_{33}(y,z)$, as shown in Figure 8, appear to be zero to the order of accuracy of the scheme. All the stress plots present values at the centers of the cells, i.e. the quantities $\tau_{ij}(u)$ as defined in (1.6).

Observe that the contours are almost, but not quite, symmetric under vertical reflection through the figure center. Full symmetry (or antisymmetry) is implied by (4.2) in the form:

$$w(x,y,z) = \pm w(x, \pm y, \pm z) \qquad (4.3)$$

where w is any stress or displacement. The lack of symmetry in our results appears because of the clamped end; it is a small effect corresponding to the difference between an exact solution and the Saint-Venant approximation. If we change the boundary conditions to have tractions at both ends, the results show complete symmetry, as seen in figure 12.

Finally, let us remark that our program can solve many problems for which no simple two dimensional formula similar to (4.2) exists. For instance it is straightforward to solve for any forces imposed at $x = x_{max}$

$$\tau_{11}=given\ ,\ \tau_{12}=given\ ,\ \tau_{13}=given \qquad (4.4)$$

5. A MORE COMPLEX GEOMETRY

Until now we have concerned ourselves with a very simple computational domain which is bricklike in shape. It is, in fact, quite feasible to treat by our methods more general bodies built by joining such blocks across their faces. We began experimenting on such an object, motivated by the "focus problem" discussed in [3].

The focus problem involves the static equilibrium of a laminate plate with stiffeners, and a circular hole. In [3] a finite element solution is presented based on a two dimensional model with averaging across the laminate layers. Our compact scheme (1.3), while inadequate for such a complex geometry, may be used for local analysis. Since it applies the basic elasticity equations without further assumptions or simplifications and is fully three dimensional, it may illuminate those points, such as delamination, which are not readily treatable by two dimensional calculations.

Figure 2 shows a small rectangle cut in the plate near a stiffener. We represent the resulting body as one block (the plate) supporting another (the stiffener). Data for this problem may be taken from the two dimensional results, and the interior displacements and stresses computed. In order to conform to the specifications of [3], we assumed that the displacements are given on the cut faces, while the other faces are stress-free.

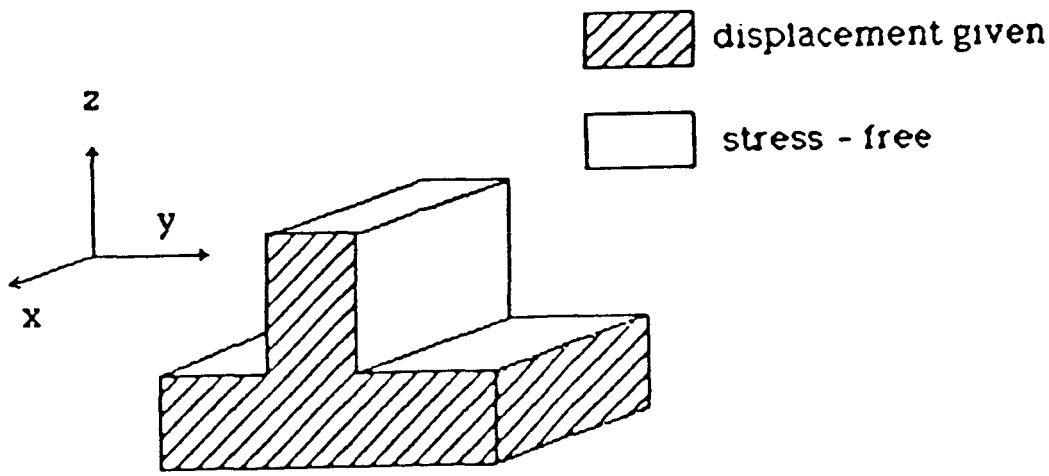


Figure 2. Plate and stiffener element.

The working algorithm for one block is readily extended to this plate-stiffener rectangle. There is only one tricky point: since the one-block solver requires residuals or data all over the faces, one must define appropriately the residuals on the junction between blocks. This definition has to preserve the symmetry of the problem, which is essential for the convergence of the conjugate gradient iterations. There are, of course, many other details one must take care of: for instance, the plate is layered in the z direction, while the stiffener is layered in the y direction ; also, the cell sizes in the two blocks need not match. Although the programming becomes tedious, there is no mathematical difficulty involved.

To further examine the block interface question, consider the two dimensional example in figure 3. One cell below the interface matches two cells above it; such a configuration may be expected as one may want to cluster points across the layers while having a coarser mesh in the layer planes. The residual at the point H is the difference between the tractions from above and below. In a simple-minded approach, one might define the traction from above as some average of the values in the two cells. Then the residual at H depends on the displacements at A,B,C,D etc. On the other hand, according to (1.8), the residual at D is the difference between two tractions from the left and the right. The compact scheme computes it using (1.6) and (1.7) on the displacements at A,B,C,D,E,F,G above the interface *without involving the displacement at H*. Thus, there can be no symmetry, as the residual at D does not depend on H, while the residual at H depends on D.

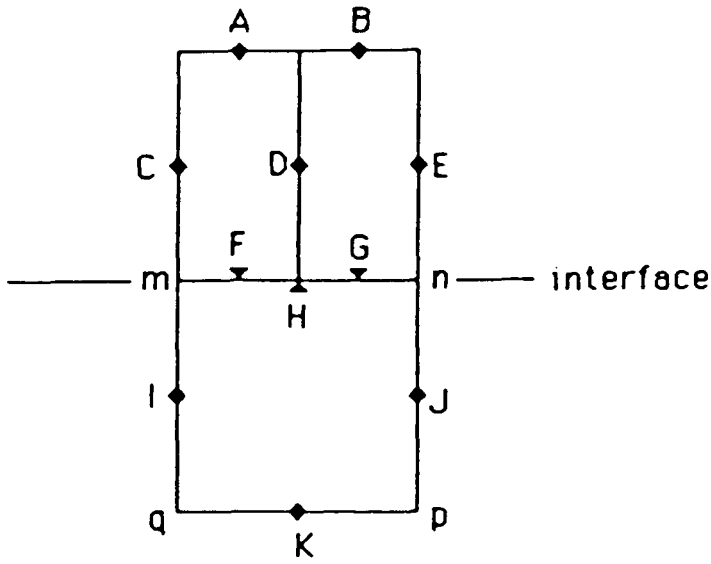


Figure 3. Cells on the interface.

The correct way of dealing with the block junction is to define a set of interface variables (displacements and residuals), distinct from the variables in the two blocks. Then these quantities are used, suitably averaged, for the separate calculations in each block. For instance, to compute the stress one needs the displacements on the surface of block 1; these may take values from the interface variables. Similarly, the residual computation in block 1 produces tractions on all the surfaces, and the tractions on the interface may be saved and then combined with the interface tractions from block 2 to obtain the residuals on the interface.

For example, we may use on the interface the finer of the two meshes , letting the values at F and G belong both to the interface and to the upper layer and then define the values below the interface:

$$u\,(H) \;=\; \frac{1}{2}\left(\, u\,(F) + u\,(G)\,\right) \tag{5.1}$$

$$\tau^{+}\,(H) \;=\; \frac{1}{2}\left(\, \tau^{-}\,(F) + \tau^{-}\,(G)\,\right)$$

One could also (according to the general method of [5]) treat the element below the interface as having 5 sides: mH, Hn, np, pq, qm, and develop equations corresponding to (1.3) to describe the equilibrium in this element. It is clear, however, that formulas (5.1) are much more convenient.

Once the interface treatment is understood, one can actually develop a general algorithm for joining blocks in arbitrary positions - a procedure somewhat akin to setting up coefficient matrices for finite elements (the blocks may be considered "macroelements" in this context).

To check the correctness of the two-block procedure we again resorted to the exact isotropic solution, as presented in section 3. The iterations are convergent and produce accurate results. The analysis of the laminate plate-stiffener element is in progress and the results will be published elsewhere.

CONCLUSIONS

We have extended the two dimensional method of reference [1] to three dimensional domains, and we have illustrated its power by solving several static equilibrium problems. The compact scheme solves the basic equations of elasticity to second order in both stresss and displacements (for smooth solutions) and allows a particularly simple numerical boundary treatment. We have identified an efficient iteration technique - the conjugate gradient method - and established the conditions for the symmetry of the residual matrix which are necessary for convergence. Our algorithm has been developed on a vector processor, and problems with $10^4 - 10^5$ dependent variables were tested.

As a relevant problem with a simple geometry, we have computed equilibrium solutions for laminate plates under various edge loadings. A tantalizing question is the resolution of the singularity which appears in composite materials at the intersection of layer interfaces and body boundary; this could be clarified by mesh refinement techniques, for which our method is well suited. Our results compare favorably with those of [2], and suggest that the compact scheme may be useful in the global-local analysis of structures. As a byproduct, we have shown that the special two dimensional ansatz of [2] is accurate, locally.

We have also extended the simple box geometry of [1] to a more complex assembly of such boxes, joined on their faces. We can treat these objects even when made of composite materials, layered in any of the x, y or z directions.

This work, although definitely of an exploratory nature, shows the viability of compact schemes in numerical elasticity and the role iterative methods can be expected to play in three dimensional problems.

We appreciate helpful discussions with W. J. Stroud, N. F. Knight and I. S. Raju, at NASA Langley .

APPENDIX : STRESS-STRAIN RELATIONS FOR ORTHOTROPIC MATERIALS

Because of obvious symmetries, the stress and strain both have only 6 independent components. Define:

$$\varepsilon = \left[\varepsilon_{11},\varepsilon_{22},\varepsilon_{33},\varepsilon_{23},\varepsilon_{13},\varepsilon_{12}\right]^{transp} \tag{a.1}$$

$$\tau = \left[\tau_{11},\tau_{22},\tau_{33},\tau_{23},\tau_{13},\tau_{12}\right]^{transp} \tag{a.2}$$

Then the relation (1.1b) may be written in terms of a 6×6 matrix C (the stiffness matrix):

$$\tau = C\,\varepsilon \tag{a.3}$$

However, the material properties are usually specified in terms of $C^{-1} \overset{def}{=} S$; the elements of S, the compliance matrix, are expressed in terms of quantities which directly measurable in the laboratory. Moreover, both C and S are symmetric matrices, which therefore may have up to 21 independent elements..

For fiber reinforced materials, like those considered in this work, there is one special direction - the fiber direction - and any two orthogonal planes through the fiber are symmetry planes. If the fiber is aligned with the x_1 axis ,

$$S = \begin{bmatrix} \frac{1}{E_1} & -\frac{\nu_{21}}{E_2} & -\frac{\nu_{31}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{12}}{E_1} & \frac{1}{E_2} & -\frac{\nu_{32}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{13}}{E_1} & -\frac{\nu_{23}}{E_2} & \frac{1}{E_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{23}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{13}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{12}} \end{bmatrix} \tag{a.4}$$

The Young moduli E_i , Poisson ratios ν_{ij} and shear moduli G_{ij} are the experimentally measured quantities. As S is symmetrical, one must have the consistency relation :

$$\frac{\nu_{ij}}{E_i} = \frac{\nu_{ji}}{E_j} \quad , \tag{a.5}$$

reducing the number of free parameters to 9. This is a consequence of the symmetries mentioned above, which in fact define an *orthotropic* material.

In a composite material layers are superimposed with varying orientations of the fibers. Taking into account the angle θ between the fiber and the x_1 axis, we may write :

$$\tau = O(-\theta)\, C\, O(\theta)\, \varepsilon \quad , \tag{a.6}$$

where $C = S^{-1}$ and S is given by (a.4). The matrix $O(\theta)$ is defined by:

$$O(\theta) = \begin{bmatrix} \cos^2\theta & \sin^2\theta & 0 & 0 & 0 & -\sin 2\theta \\ \sin^2\theta & \cos^2\theta & 0 & 0 & 0 & \sin 2\theta \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ \sin\theta\cos\theta & -\sin\theta\cos\theta & 0 & 0 & 0 & \cos 2\theta \end{bmatrix} \tag{a.7}$$

Figure 4. $u_1(x)$ at y=0.55 , z=.0825

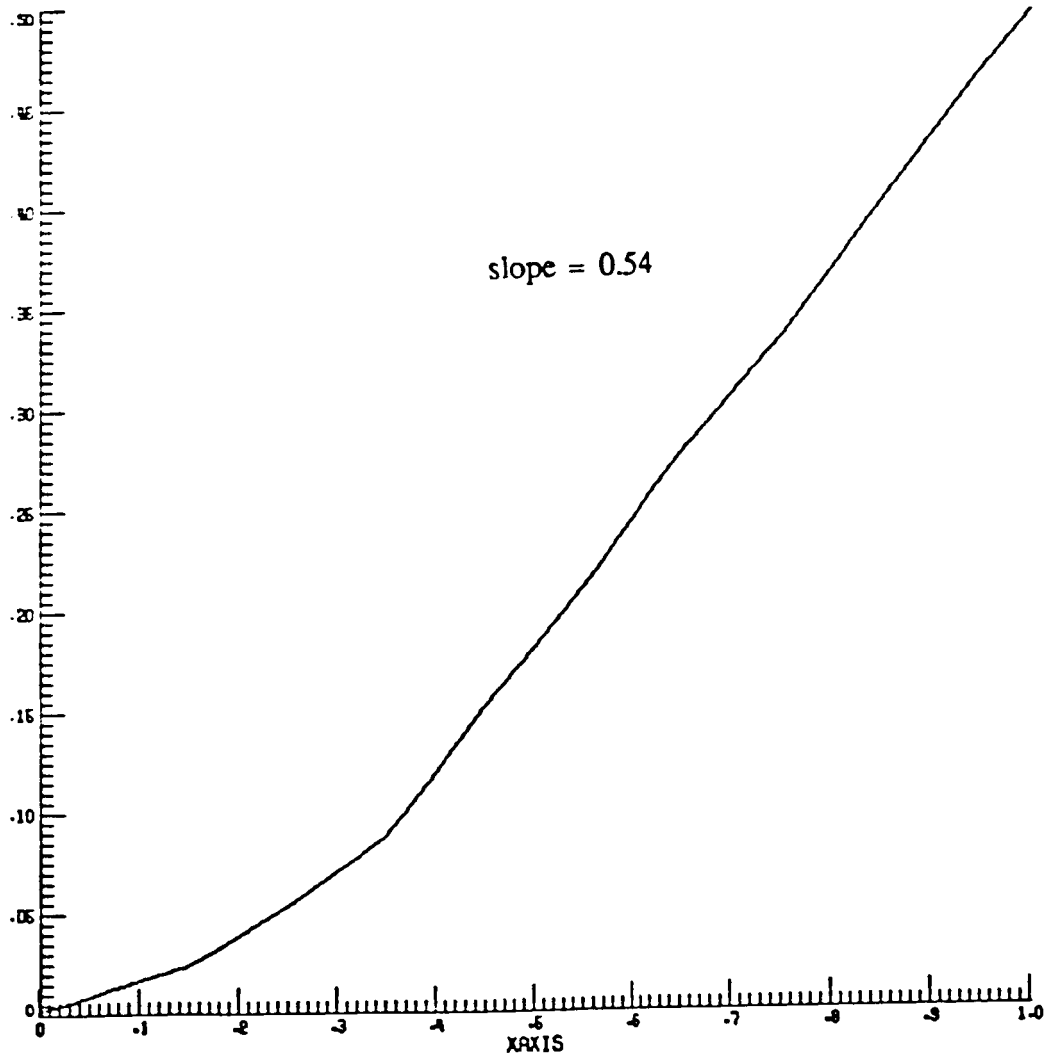


Figure 5. $u_1(x)$ at y=1.05 , z=.1175

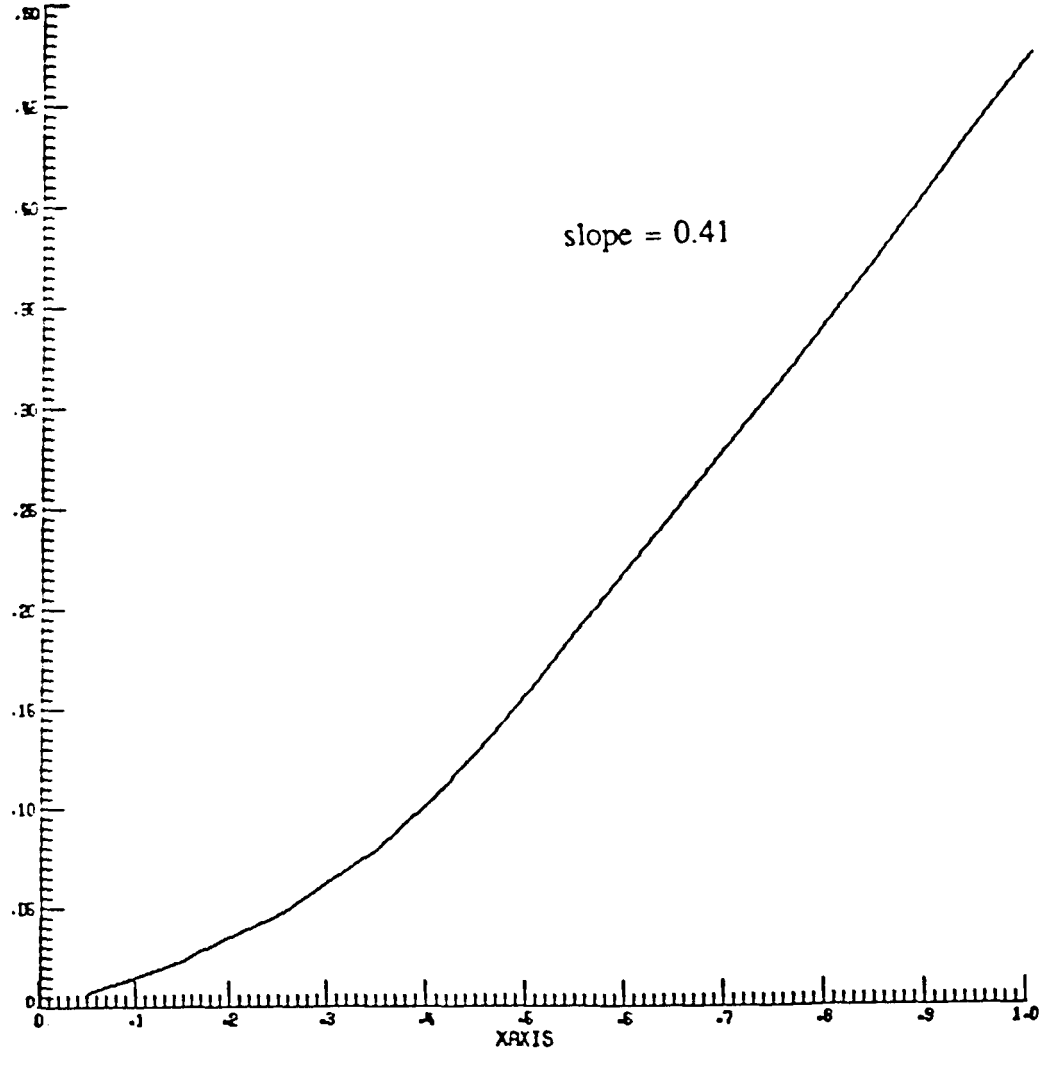


Figure 6. τ_{11} (y,z) *at* x=0.55

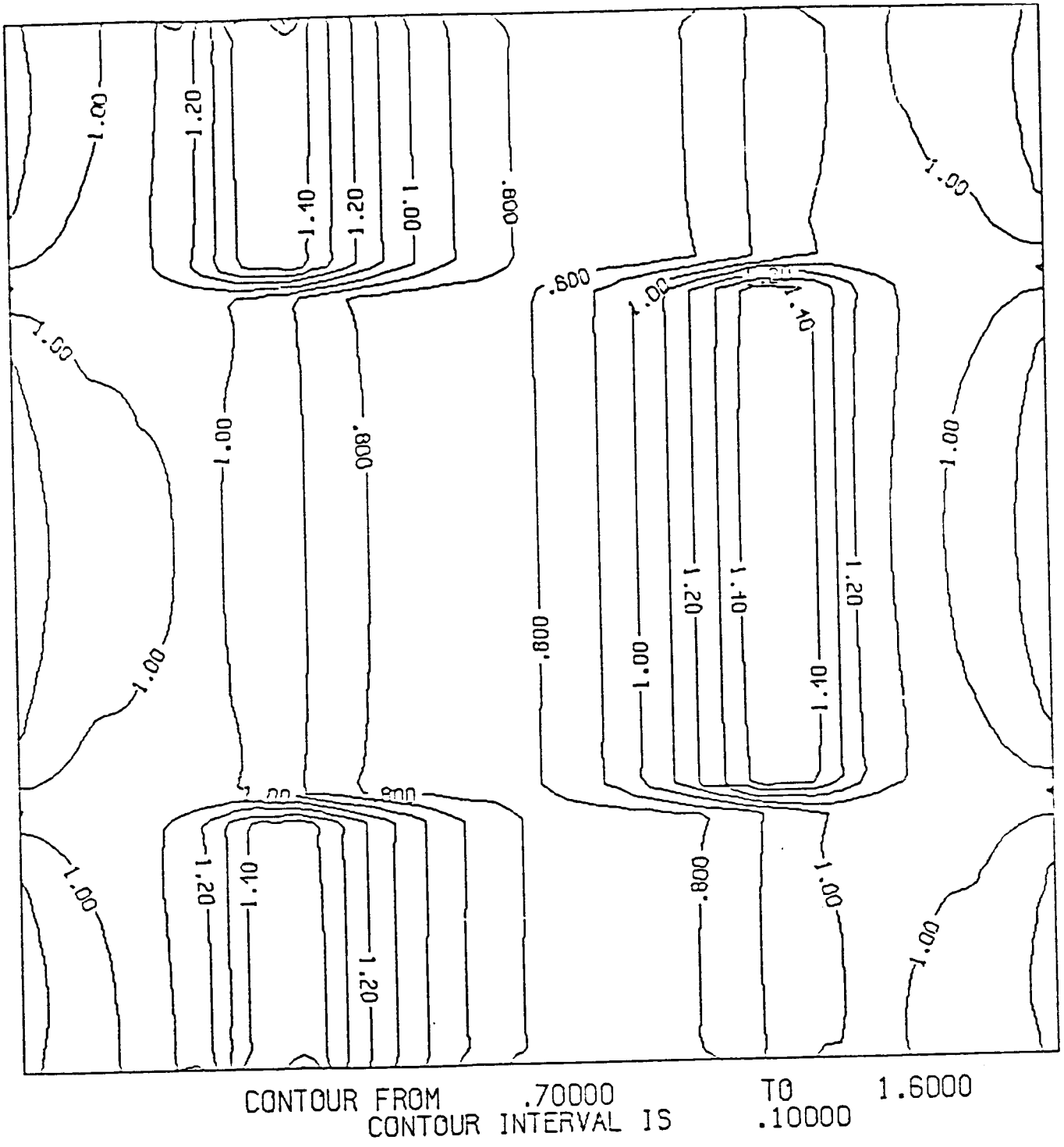


Figure 7. τ_{22} (y,z) *at* x=0.55

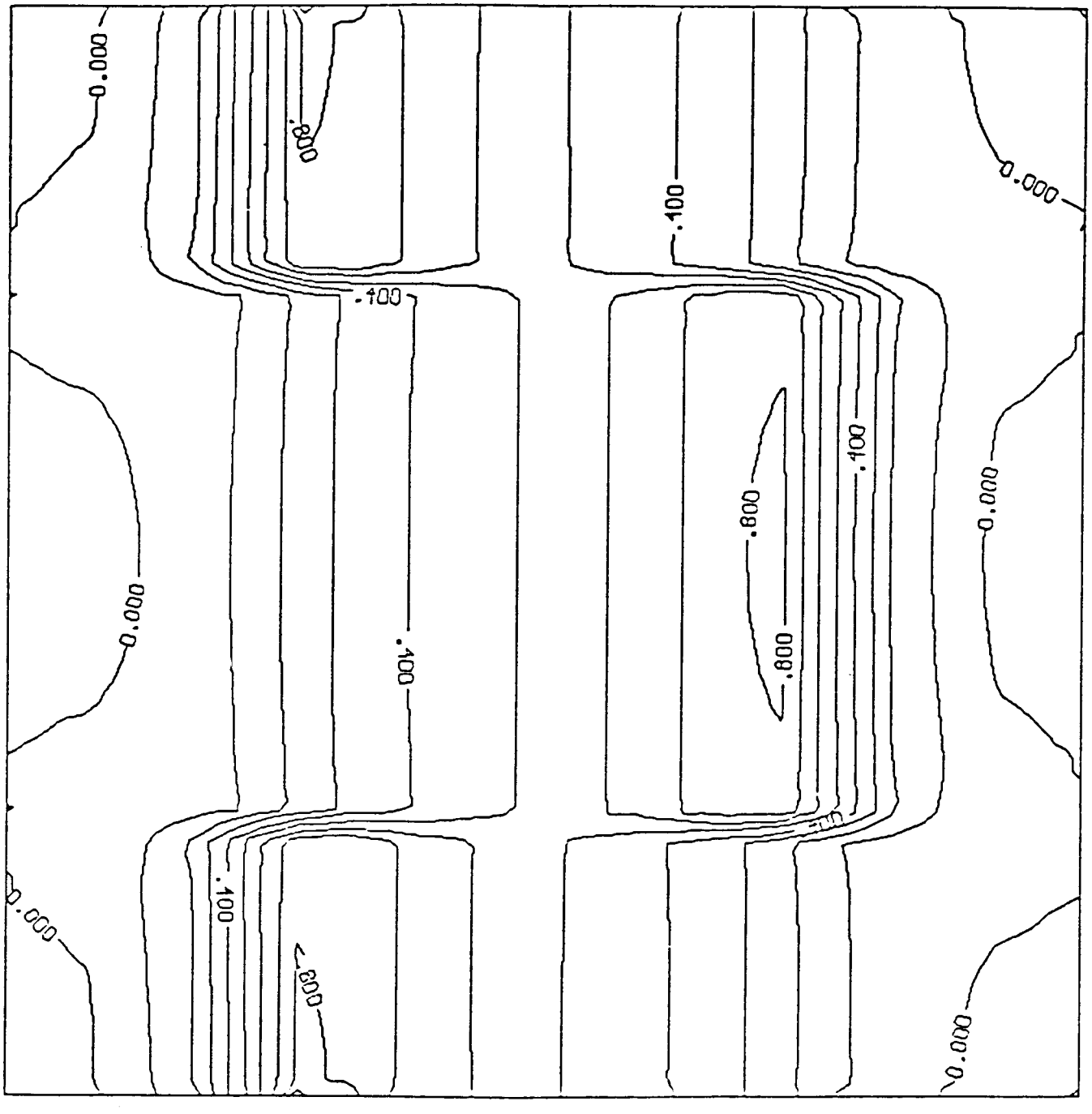


Figure 8. τ_{33} (y,z) at x=0.55
The numerous extrema seem to be numerical oscillations.

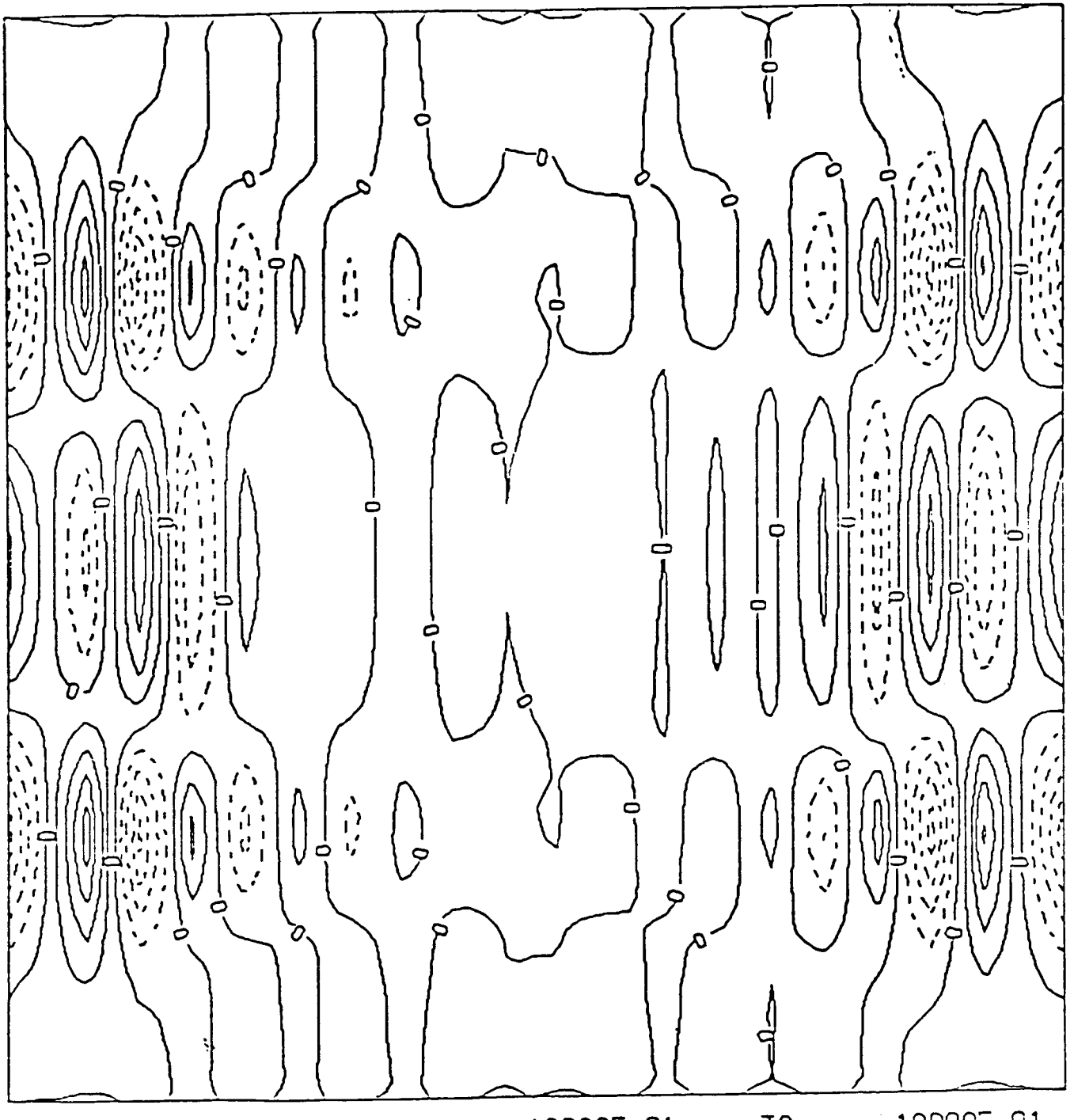


Figure 9. τ_{12} (y,z) *at* x=0.55

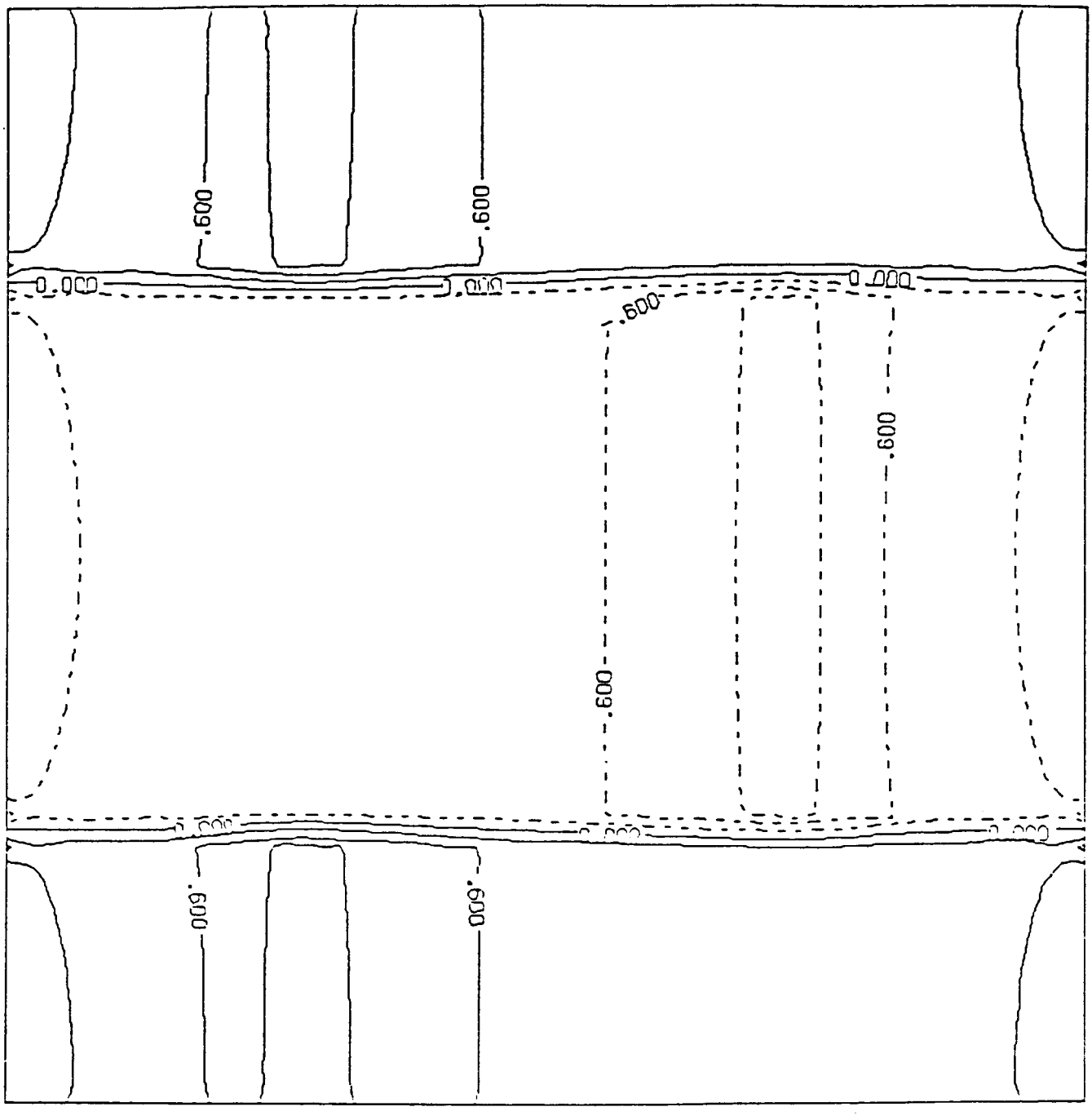


Figure 10. τ_{13} (y,z) *at* x=0.55

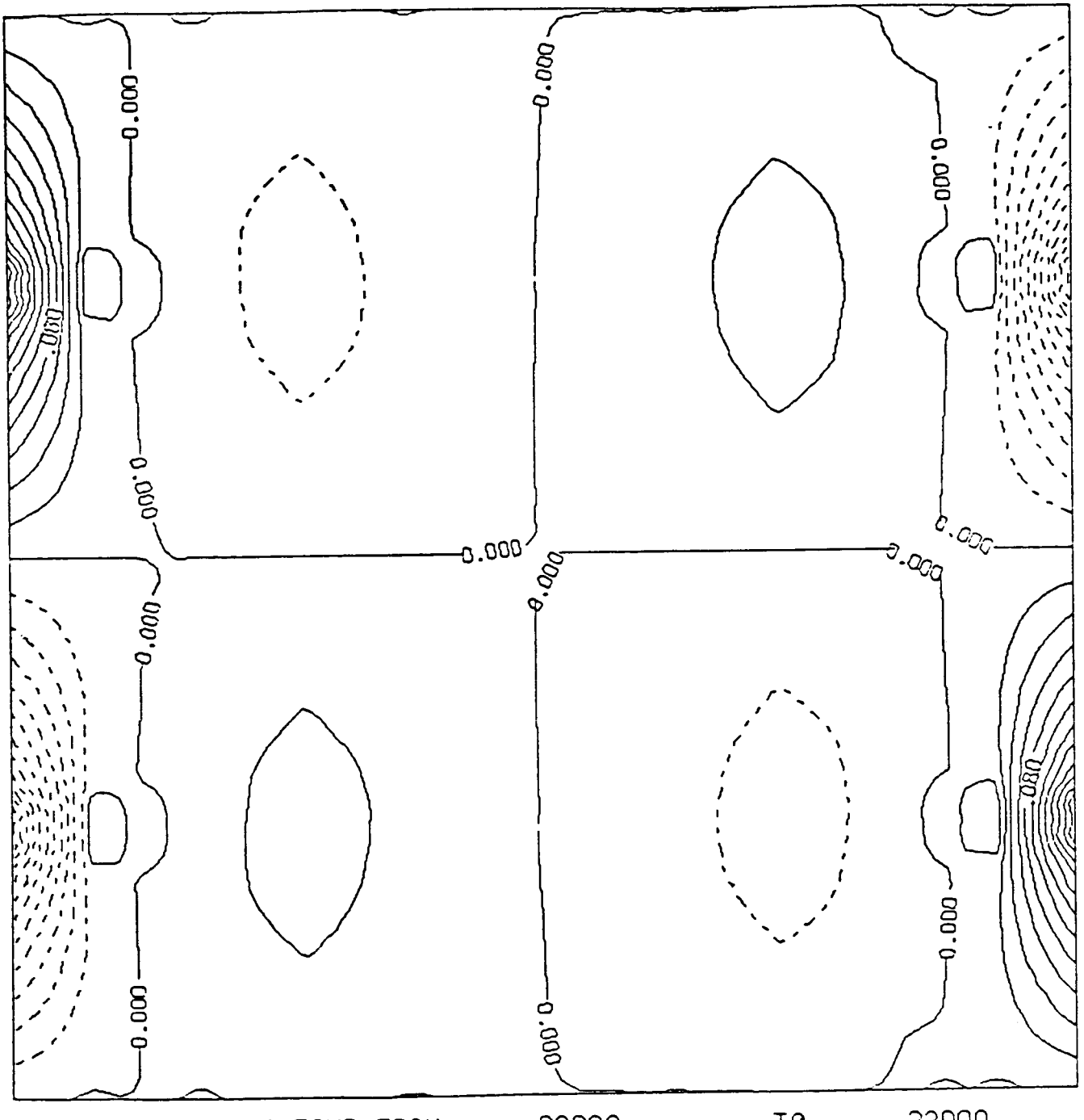


Figure 11. τ_{23} (y,z) *at* x=0.55

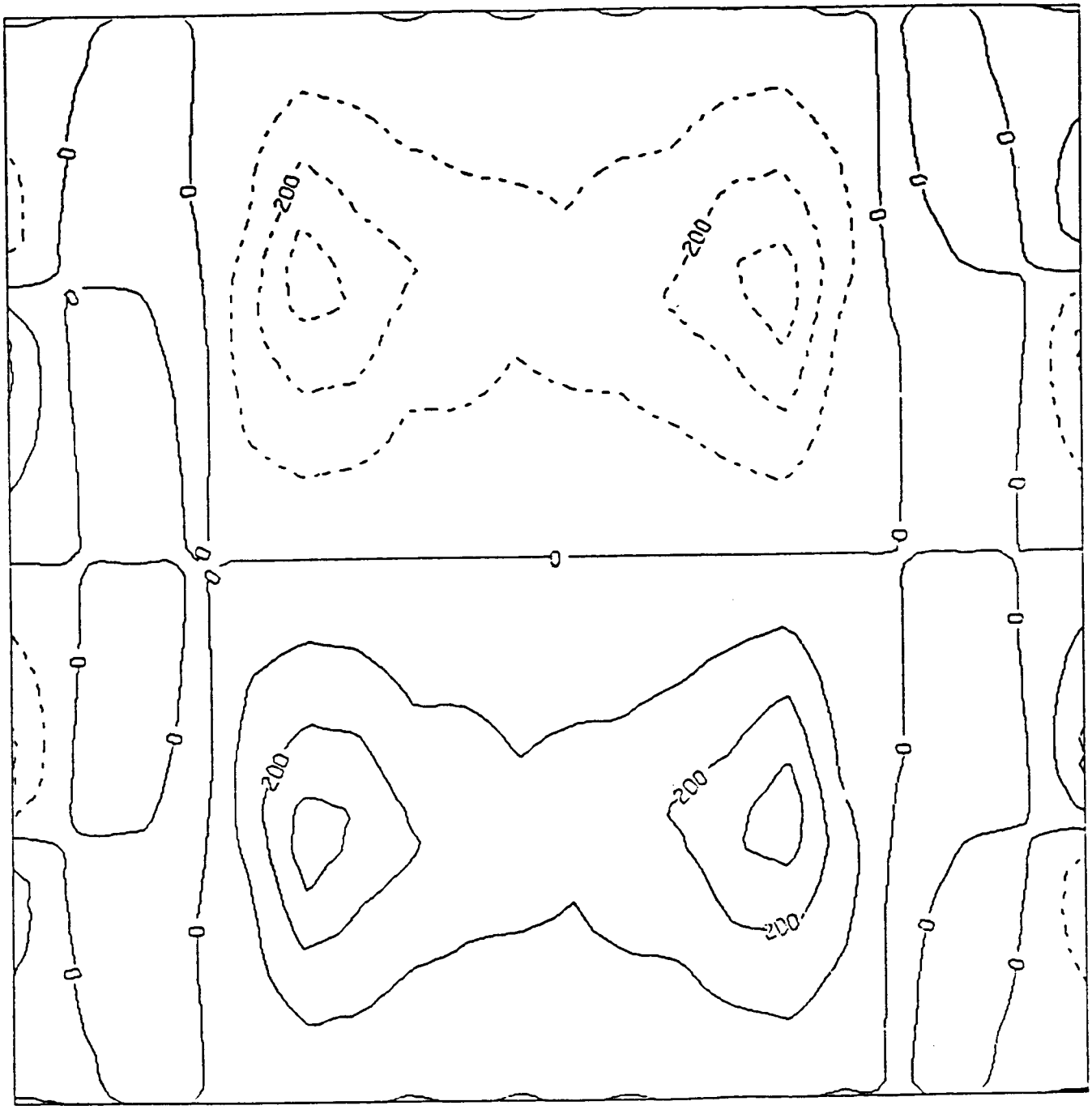
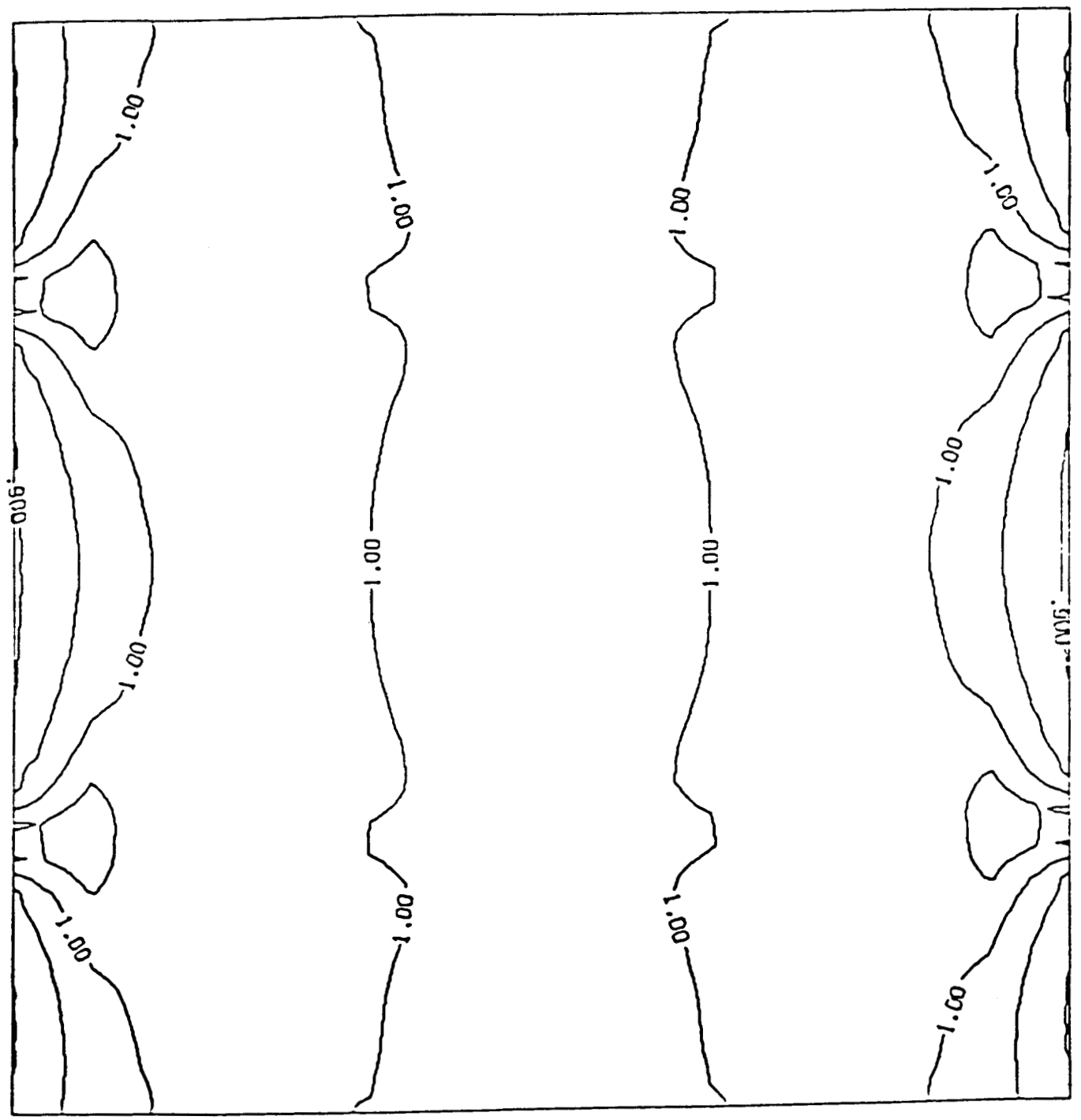


Figure 12. Symmetrical τ_{11} (y,z) at x=0.55 under traction boundary conditions at $x = x_{min}$, $x = x_{max}$



REFERENCES

1. Phillips, T. N. and Rose, M. E. "*A finite difference scheme for the equilibrium equations of elastic bodies*" SIAM J.Sci.Stat.Comput., Vol. 7, No. 1, 1986, pp. 288-300

2. Whitcomb, J. D., Raju I. S. and Goree, J. G. "*Reliability of the finite element method for calculating free edge stresses in composite laminates*" Computers and Structures, Vol. 15, No. 1, 1982, pp. 23-37

3. Knight, N. F. and Stroud, W. J. "*Computational structural mechanics : a new activity at NASA Langley Research Center*" NASA Technical Memorandum 87612, 1985

4. Golub, G. H. and van Loan, C. F. "*Matrix computations*" Johns Hopkins University Press, Baltimore, 1983

5. Rose, M. E. "*A compact finite element method for elastic bodies*" Numer.Meth.P.D.E., Vol. 3, 1985, pp. 209-228

Standard Bibliographic Page

1. Report No. NASA CR-178158 ICASE Report No. 86-51	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A THREE DIMENSIONAL CALCULATION OF ELASTIC EQUILIBRIUM FOR COMPOSITE MATERIALS		5. Report Date July 1986
		6. Performing Organization Code
7. Author(s) Liviu R. Lustman, Milton E. Rose		8. Performing Organization Report No. 86-51
9. Performing Organization Name and Address Institute for Computer Applications in Science and Engineering Mail Stop 132C, NASA Langley Research Center Hampton, VA 23665-5225		10. Work Unit No.
		11. Contract or Grant No. NAS1-17070, NAS1-18107
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report
		14. Sponsoring Agency Code 505-31-83-01

15. Supplementary Notes

Langley Technical Monitor:
J. C. South

Final Report

16. Abstract

A compact scheme is applied to three-dimensional elasticity problems for composite materials, involving simple geometries. The mathematical aspects of this approach are discussed, in particular the iteration method. A vector processor code implementing the compact scheme is presented, and several numerical experiments are summarized.

17. Key Words (Suggested by Authors(s))	18. Distribution Statement
compact scheme, elasticity, laminates, vector processor, conjugate gradient method	39 - Structural Mechanics 64 - Numerical Analysis Unclassified - unlimited

19. Security Classif.(of this report)	20. Security Classif.(of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	28	A03

For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1986