An Improved Boundary Element Method for Strike-Slip Fault Mixed-Value Problems with Application to the Anza Seismic Gap, Southern California

Roderick A. Flores
New Mexico Institute of Mining and Technology
Socorro, NM 87801

Submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Geophysics (Solid Earth)

Table of Contents

Table of Figures

Abstract

Geoscientists are very interested in the stress patterns around seismically active areas. Many researchers have begun adapting engineering style boundary integral techniques to address this problem. These integral methods are a very powerful tool for describing an entire physical system, using only parameters given for a specified boundary. Once a solution satisfying the boundary conditions is obtained, all other points within the body can be described. Unlike finite difference or finite element approaches, boundary element methods have no need for an unwieldy mesh or grid to describe any system in detail.

The integral problem of particular interest with respect to elastic deformation is Kelvin's problem: a steady state homogeneous isotropic plane stress model. This approach has a simple boundary integral equation solution and can easily be adapted to model fault properties (Cruse, 1975). In particular, one of these boundary element integral solution algorithms will be applied to the Anza region of the San Jacinto fault zone. The Anza region is of particular interest because it is characterized by an elevated level of seismicity with respect to most of Southern California. The epicentral pattern of the subset of events consisting of moderate to large magnitudes defines an approximately 40 kilometer long slip deficit near Anza. The northern 20 kilometers of this slip gap are also characterized by anomalously low micro-seismicity.

Near the region of this seismic gap, several thousand focal mechanisms were inverted to produce stress magnitude ratios and orientations. These data show impressive deviations from the regional trend of the principle stress direction. Specifically, the orientations roughly trend from east-of-north west of the Anza seismic gap and west-of-north east of the gap (Hartse *et al.* 1994).

Simple boundary element modeling of faults is typically done by specifying a displacement or a stress field along a specific boundary. Normally this method produces robust results, but in situations where a stress and displacement condition are given simultaneously (mixed value problem), the method becomes completely unstable. This problem is corrected by altering the boundary conditions for the standard algorithms so as to better describe a strike-slip system.

Introduction

The San Jacinto fault zone (SJFZ) is a major splay of the San Andreas fault system in southern California (Plate 1). It originates in the Cajon Pass between the San Bernardino and San Gabriel Mountains and strikes approximately $S40^oE$ through the San Bernardino, San Jacinto, Borrego, and Imperial Valleys, where it terminates near the Mexican border. The fault zone is made up of nine identifiable parallel to sub-parallel strands, with lengths between 30 to 70 kilometers (Vernon, 1989). The geology through which the fault passes is dominated by exposed and alluvium covered intrusive rocks of the mid-Cretaceous southern California batholith, as well as prebatholithic metasedimentary and possibly metavolcanic rocks (Sharp, 1967).

Using a number of right-lateral slip events, two seismic slip gaps have been identified along the SJFZ: one from Cajon Pass to Riverside and the other from Anza to Coyote Mountain (Abe, 1988, Thatcher *et al.* 1975). Both of these gaps are believed to be capable of producing earthquakes with a magnitude as high as $M_L = 7$, but because significant amounts of slip have accumulated on either side of the Anza gap, it is likely to be the more significant of the two (Vernon 1989). In addition, a zone of anomalously low seismicity rates defines an aseismic gap with the northern half of the Anza slip gap (Scott, 1992; Sanders and Kanamori 1984). Because seismicity and slip gaps such as these are considered to be excellent aides in forecasting earthquake hazard (WGCEP, 1988), a great deal of study has and is being done on understanding their origins.

One recent study used focal mechanism data to invert for stress ratios and orientations for the Anza region of the SJFZ (Hartse, *et al.* 1994). In general, the observed orientations of the principle stresses are those expected for a strike-slip dominated system: the most ($\hat{\sigma}_1$) and least ($\hat{\sigma}_3$) compressive stresses are nearly horizontal and the intermediate principle stress ($\hat{\sigma}_2$) vertical. The observed ratio of the intermediate to most compressive stress with respect to the ratio of the least to most compressive stresses is always between zero and one. When this value is near zero, then the magnitudes of the most compressive stress and the intermediate stress are nearly the same. When this value is near one, then the magnitudes of the intermediate stress and the least compressive stress are close to the same. (Hartse, personal communication). The inversion also found a north-south oriented regional stress condition which agrees well with other estimates for the Anza area (Hartse, *et al.* 1994).

The stress observations also display a great deal of regional variation in the orientation of the principle stresses; a reverse component of faulting occurs near the end of the mapped trace of the Hot Springs Fault where the orientation of $\hat{\sigma}_1$ plunges 30^o to the north-northwest; in the shallow off-SJFZ Cahuilla swarms just west of the slip gap, $\hat{\sigma}_1$ is oriented $N25^oE$; normal faulting becomes important in the trifurcation of the SJFZ, where $\hat{\sigma}_1$ plunges 45^o to the south; $\hat{\sigma}_1$ is oriented $N10^oW$ in the area of the Toro Peak swarms; and $\hat{\sigma}_1$ rotates in a clockwise fashion from east to west. These stress orientations are believed to the result of a block rotation superimposed onto the right-lateral strike-slip motion which is dominant in the SJFZ region (Hartse, *et al.* 1994). This model implies that the Anza seismic gap is the region of zero convergence between the northwest and southeast sides of the fault.

To better understand the stress anomalies that are associated with the highly clustered seismicity patterns from which they are calculated, a direct boundary integral computer code based upon the algorithm of Crouch and Starfield (1990) has been written. The purpose of this program is to analyze the role of fault geometry and strength in influencing localized stress anomalies. Similar studies have had success calculating the stresses resulting from the displacement fields of observed slip events (Stein *et al.*, 1992; Gomberg, 1991; Billham & King, 1989).

In slip gaps known displacement perturbations are, by definition, nonexistent. This problem is solved by placing a specified fault configuration within a far-field driving stress. Using this method, several scenarios for a slip gap are presented with increasing complexity both in numbers of faults and fault properties. A model describing the state of stress for the Anza gap is proposed using the information obtained from these test cases.

Mathematical Analysis and Methods

In order to understand the alteration of the algorithm presented by Crouch and Starfield (1990), it is necessary to begin with the basic mathematics of the method. Following Sokolnikov (1956), consider a steady-state homogeneous isotropic solid of infinite extent in the x_i directions. This solid has body forces F_i and tractions $\sigma_{ji}\nu_j$ applied to it, where σ_{ji} is the stress tensor and ν_j are the directional cosines that define the coordinate system. These forces must

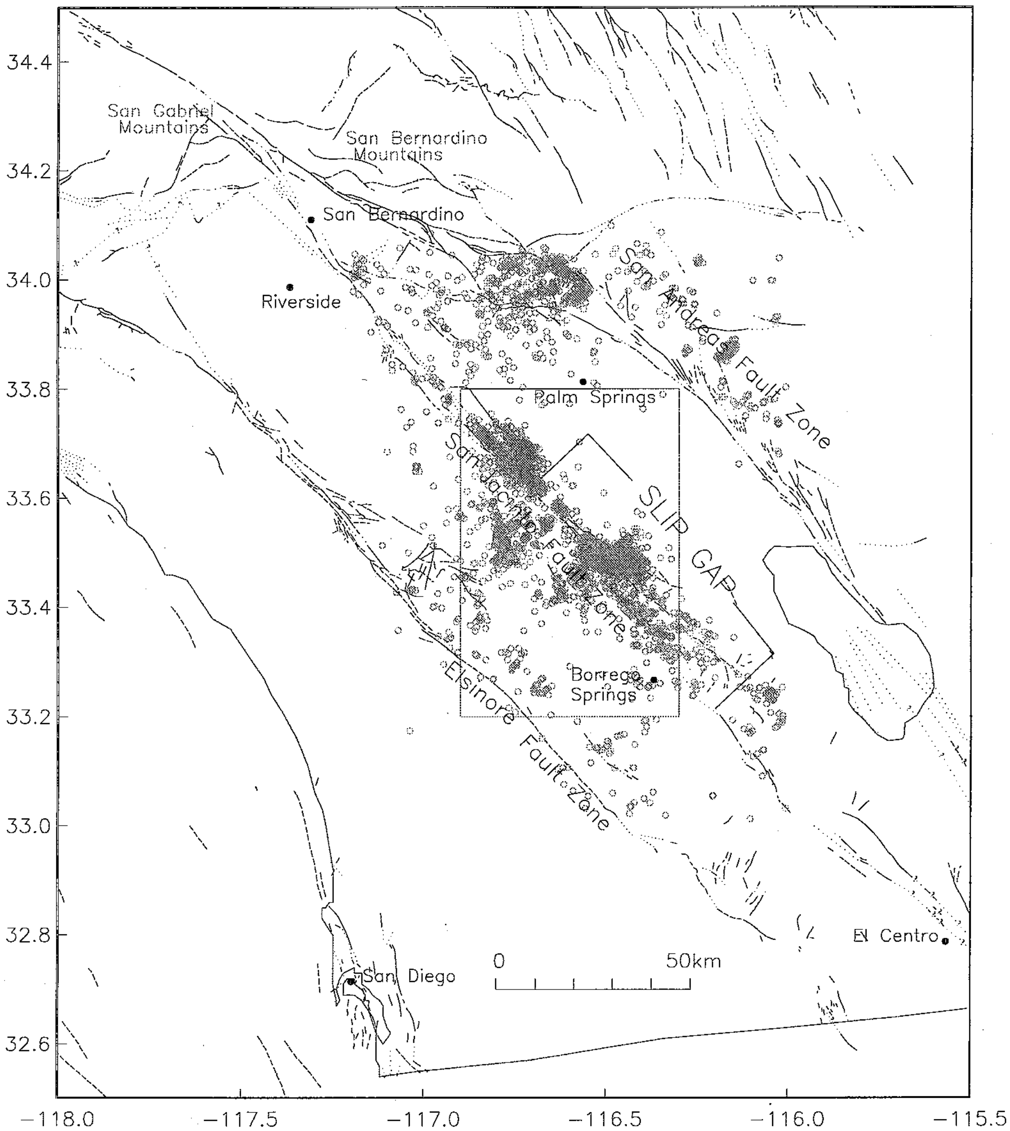


Plate 1: The region surrounding the San Jacinto fault zone showing major fault zones, 3846 seismic events recorded by the Anza seismic network (open circles) and the Anza slip gap. The area where the study is focused lies within the box outline.

sum to zero in order to satisfy the conditions of equilibrium:

$$\int_V F_i dV + \int_S \sigma_{ij}\nu_j dS = 0, \tag{1}$$

where dV is an infinitesimal unit of volume and dS is an infinitesimal portion of the surface of the body. Assuming that σ and its first-partial derivatives are continuous and single-valued within the volume, the divergence theorem can be applied to the left hand term of (1) to obtain:

$$\int_S \sigma_{ji}\nu_j dS + \int_V \sigma_{ij,j} dV = 0. \tag{2}$$

The volume V is arbitrary and the integrand is continuous, so (2) can be substituted into (1) to show:

$$\sigma_{ji,j} = -F_i. \tag{3}$$

The stresses in equation (3) can be replaced using Hooke's law:

$$\sigma_{ij} = \lambda\delta_{ij}\theta + 2\mu\epsilon_{ij}, \tag{4}$$

where λ and μ are the Lamé constants, δ_{ij} is the Kronecker delta, and θ is the the trace of the strain tensor, ϵ_{ij}. As a consequence the Navier equation is derived (Sokolnikov 1956; Love 1927):

$$\mu\nabla^2 u_i + (\lambda + \mu)\theta_{,i} = -F_i. \tag{5}$$

A plane stress fundamental solution to this integral equation (Lord Kelvin's problem) can be represented by the function (Crouch & Starfield 1990; Hartman, 1989),

$$g(x,y) = \frac{-1}{4\pi(1-\nu)} ln(x^2 + y^2)^{\frac{1}{2}}. \tag{6}$$

Using the solution function (6) and its derivatives, the displacements and stresses of the system can be described (Crouch & Starfield, 1990):

$$\begin{aligned} u_x &= \frac{F_x}{2G}\left[(3-4\nu)g - x\frac{\partial g}{\partial x}\right] + \frac{F_y}{2G}\left[-y\frac{\partial g}{\partial x}\right] \\ u_y &= \frac{F_x}{2G}\left[-x\frac{\partial g}{\partial y}\right] + \frac{F_y}{2G}\left[(3-4\nu)g - y\frac{\partial g}{\partial y}\right] \end{aligned} \tag{7a}$$

$$\sigma_{xx} = F_x\left[2(1-\nu)\frac{\partial g}{\partial x} - x\frac{\partial^2 g}{\partial x^2}\right] + F_y\left[2\nu\frac{\partial g}{\partial y} - y\frac{\partial^2 g}{\partial x^2}\right]$$

$$\sigma_{yy} = F_x\left[2\nu\frac{\partial g}{\partial x} - x\frac{\partial^2 g}{\partial y^2}\right] + F_y\left[2(1-\nu)\frac{\partial g}{\partial y} - y\frac{\partial^2 g}{\partial y^2}\right]$$

$$\sigma_{xy} = F_x\left[(1-2\nu)\frac{\partial g}{\partial y} - x\frac{\partial^2 g}{\partial x \partial y}\right] + F_y\left[(1-2\nu)\frac{\partial g}{\partial x} - y\frac{\partial^2 g}{\partial x \partial y}\right]. \qquad (7b)$$

where F_x and F_y are components of the force located at the points x and y along the boundary. The stresses in equation (7b) follow the standard engineering convention : positive is dilatational and negative is compressional (Figure 1).

In order to make use of these equations it is necessary to introduce the reciprocal theorem: *"The whole work done by the forces (stresses) of the first set (including kinetic reactions) acting over the displacements produced by the second set, is equal to the whole work done by the forces of the second set, acting over the displacements produced by the first,"* (Love, 1927). This can be demonstrated mathematically as follows:

$$\int\limits_C \left(\sigma_s u'_s + \sigma_n u'_n\right) dS = \int\limits_C \left(\sigma'_s u_s + \sigma'_n u_n\right) dS, \qquad (8)$$

where σ_s is the fault parallel (shear) stress, σ_n is the fault perpendicular (normal) stress, u_s is the shear displacement, u_n is the normal displacement. The primed system in this equation corresponds to the known "first set" of stresses and displacements and the unprimed system to the unknown "second set" (Crouch & Starfield 1990; Love 1927). Assuming that the boundary can be approximated by n straight line segments, equation (8) becomes:

$$\sum_{j=1}^{n} \int\limits_{\overset{j}{\Delta}_s} \left(\sigma_s u'_s + \sigma_n u'_n\right) dS = \sum_{j=1}^{n} \int\limits_{\overset{j}{\Delta}_s} \left(\sigma'_s u_s + \sigma'_n u_n\right) dS, \qquad (9)$$

where $\overset{j}{\Delta}_s$ is the j^{th} line segment which can also be thought of as a boundary element. Next assuming that the stresses and displacements are constant over each element of the unknown (unprimed) system, equation (9) becomes:

$$\sum_{j=1}^{n} \overset{j}{\sigma}_s \int\limits_{\overset{j}{\Delta}_s} u'_s dS + \sum_{j=1}^{n} \overset{j}{\sigma}_n \int\limits_{\overset{j}{\Delta}_s} u'_n dS =$$

$$\sum_{j=1}^{n} \overset{j}{u}_s \int\limits_{\overset{j}{\Delta}_s} \sigma'_s dS + \sum_{j=1}^{n} \overset{j}{u}_n \int\limits_{\overset{j}{\Delta}_s} \sigma'_n dS. \qquad (10)$$

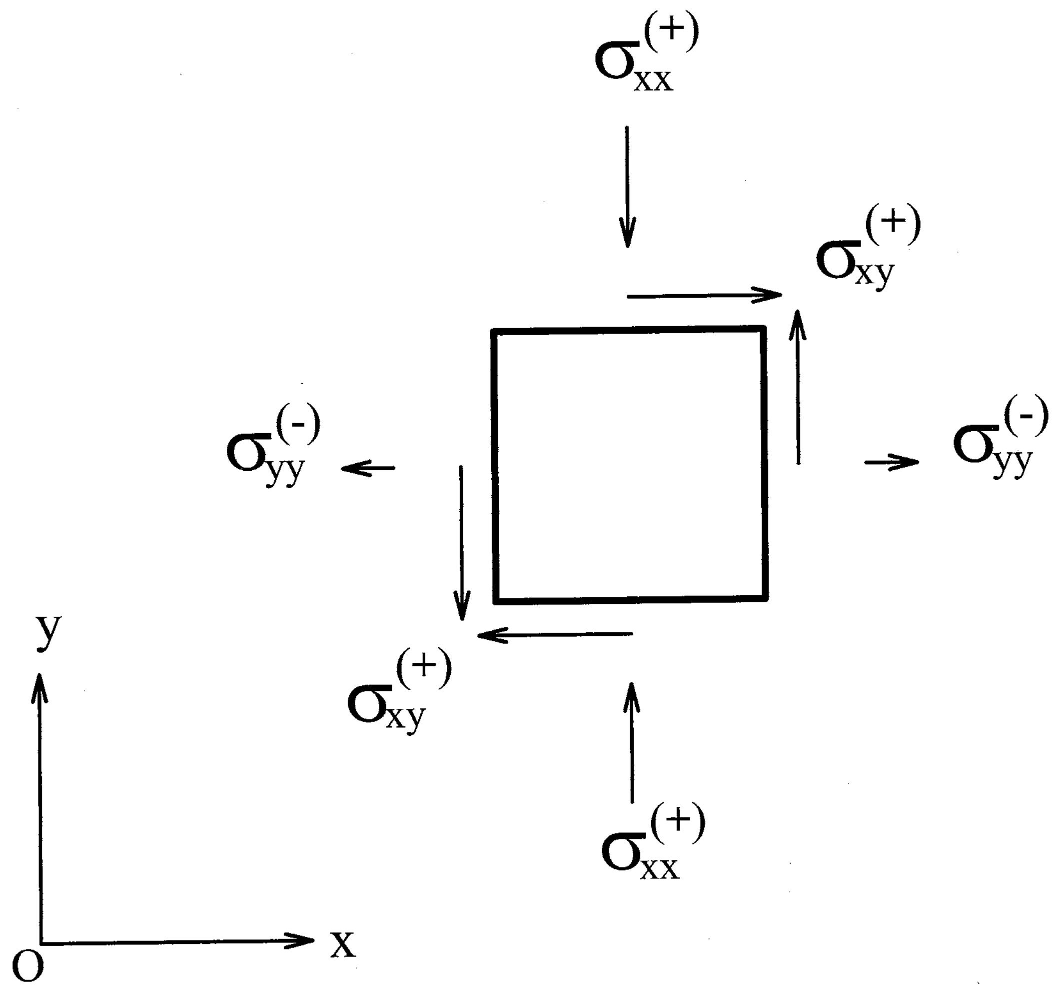


Figure 1: The convention of the stresses in this study follow standard engineering definition: positive is dilatational and negative is compressional.

where $\overset{j}{\sigma}$ and $\overset{j}{\sigma}_n$ are the stresses and $\overset{j}{u}_s$ and $\overset{j}{u}_n$ are displacements at the j^{th} element, and $\overset{j}{\sigma}{}'_s$, $\overset{j}{\sigma}{}'_n$, $\overset{j}{u}{}'_s$ and $\overset{j}{u}{}'_n$ are the stresses and displacements defined in equation (7).

The stresses and displacements from equation (7) can be rotated into local shear and normal coordinate systems for each boundary element in the system. Through the application of concentrated unit forces in the shear ($\overset{i}{F}_s$) and normal ($\overset{i}{F}_n$) directions at the midpoint of each of these elements, the integrals in equation (10) define a number of Green functions describing the response of each element based upon Poisson's ratio (ν), Young's modulus (E), and the x and y coordinates. If boundary conditions are placed upon two of the four unknown stresses and displacements of each element, the Green functions can be used to solve the linear system of equations for the remaining values.

Modeling

Application of the previous mathematical method typically utilizes, but is not limited to a homogeneous plate that is infinite in extent. The elastic behavior of this body is defined by density and P-wave and S-wave velocities. These observable bulk-properties are then converted into the constants of Young's modulus and Poisson's ratio.

Individual fault zones are represented by a series of vertically plunging straight line segments connected head-to-toe along the strike of the fault trace. Boundary conditions are prescribed and kept constant over each of these defined elements. Far-field driving stress patterns are specified by a linear addition with the Green functions (equation 10) which describe the response of the system. Modeled faults, as a rule, must be longer than the elastic thickness of the lithosphere so as to comply with the plane stress condition imposed by the solution function (equation 6).

The assumption that stresses and displacements are continuous along these discrete boundaries (11) dictates that each boundary element essentially decomposes into a "particle" of a radius equal to half of its length. No additional stresses or displacements may be accurately calculated within the area encompassed by these particles. boundary element length must be shorter than the area that requires resolution. In addition, simple boundary integral equations do not produce results accurate to within two or three radii of the elements (Wendland, 1987). This framework requires that the number and positioning of the boundary element line segments be accurately chosen to best describe the mapped trace of the faults.

Breakdown of the Mixed Value Problem

The Anza Gap is dominated by strike-slip faulting. Modeling of this type of fault by specifying a known displacement or stress field is quite robust (Stein *et al.*, 1992; Gomberg, 1991; Billham & King, 1989). No such *a priori* information exists for seismic gaps. Stresses must be calculated from the relaxation of a fault geometry and/or strength distribution under regional driving stresses. Individual faults are mimicked using given mixed boundary conditions in directions normal and parallel to each boundary element. The typical model for a strike-slip fault is a given relaxation shear stress strength and zero opening normal to the strike of the fault:

$$\begin{aligned} \sigma_s &= f(s) \\ u_n &= 0. \end{aligned} \tag{11}$$

Application of this type of mixed boundary condition problem for simple models results in small instabilities. However, as the complexity of the given problem increases, the instability becomes quite pronounced. (Figure 2). In this situation, an antisymmetric stress distribution about the fault is expected (Das & Scholz, 1982; Chinnery, 1963), but what is observed is a large accumulation of stress and displacement to one side of the fault. This result is likely caused by the unstable nature of mixed boundary condition problems. Problems of this type are normally solved using a biharmonic rather than simple harmonic integral solution function (Sanford, 1959).

The generally accepted strike-slip fault model which imposes a constant normally-oriented displacement condition in a plate under regional stress conditions is equivalent to placing an impassable barrier in a fluid medium (over geologic time). Material is only allowed to strain along fault, causing an extreme build-up of the normal component of the strain and therefore stresses. In order to eliminate the problem of a constant normally directed displacement while maintaining zero fault opening, double-sided boundary constraints are proposed (Figure 3):

$$\left.\begin{aligned} u_n^{(+)} &= -u_n^{(-)} \\ \sigma_n^{(+)} &= \sigma_n^{(-)} \end{aligned}\right\} \text{calculated (characterizes normal stress and displacement discontinuity)}$$

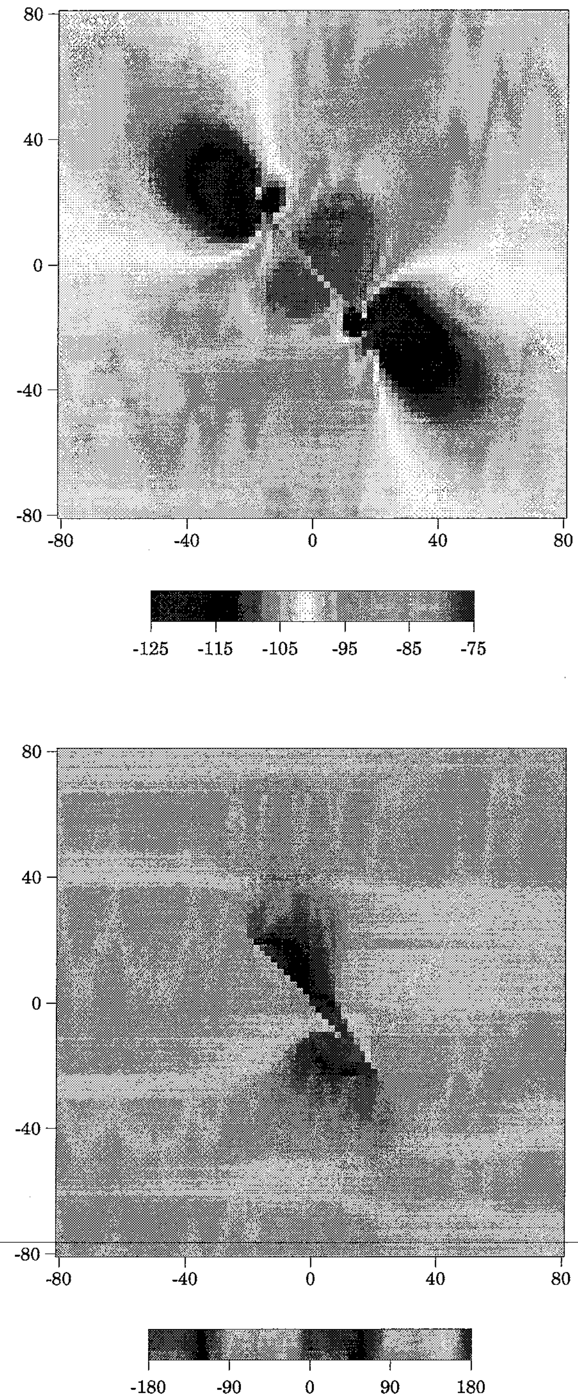
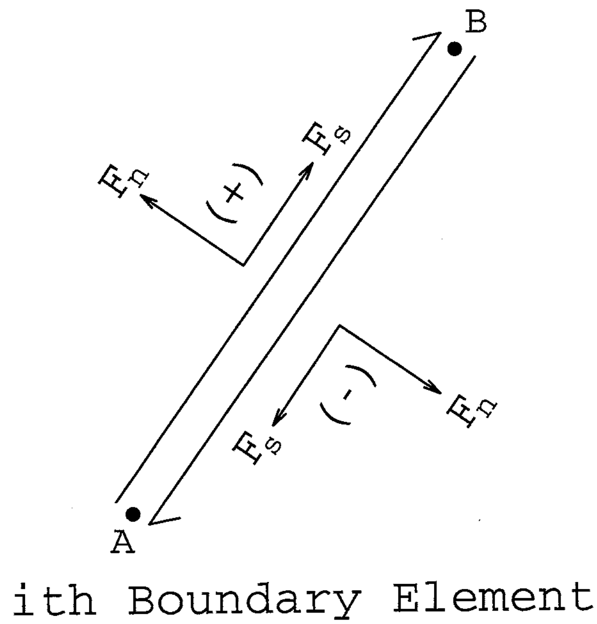


Figure 2: Most compressive stress magnitude (upper) and orientation (lower) for a zero-strength fault under 100 MPa uniaxial N-S compression extending from the X-Y coordinates (-20,20) to (20,-20) calculated using the standard direct boundary integral algorithm of Crouch and Starfield (1990). The magnitude of the stress ranges from approximately - 125 MPa (purple) to - 75 MPa (red) and should be anti-symmetric about the fault. The orientations of the stresses also display this undesirable pattern.



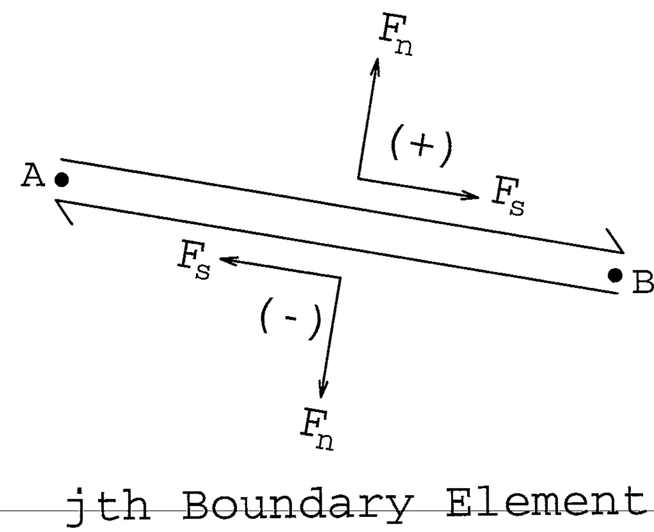


Figure 3: Two randomly oriented double-couple boundary elements. These elements are constructed by given initial and final X-Y coordinates. They are assumed to be double-sided and are in contact. The couple formed by the endpoints is followed in a clockwise fashion from A to B to define the positive side of an element and from B to A for the negative side. Each side defines localized coordinate system with shear (element parallel) and normal (fault perpendicular) directions. In this representation, the unit forces that are used to create the Green functions necessary to the system of equations of the algorithm are placed along these coordinate axis directions.

$$\left.\begin{array}{l}\sigma_s^{(+)}\\ \sigma_s^{(-)}\end{array}\right\}\text{ known (characterizes shear strength)}$$

$$\left.\begin{array}{l}u_s^{(+)}\\ u_s^{(-)}\end{array}\right\}\text{ calculated (characterizes fault parallel motion)} \tag{12}$$

This is the statement of normal displacement and normal stress continuity for a fault of known strength. These conditions rearrange (11) into the new system of equations:

$$\begin{aligned}\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{+})\int_{\overset{j}{\Delta}_s} u'_s(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] &= -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s} u'_n(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]\\ &+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{+})\int_{\overset{j}{\Delta}_s} \sigma'_s(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]\\ &+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s} \sigma'_n(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] \quad (13a)\end{aligned}$$

$$\begin{aligned}\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{+})\int_{\overset{j}{\Delta}_s} u'_s(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\right] &= -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s} u'_n(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\right]\\ &+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{+})\int_{\overset{j}{\Delta}_s} \sigma'_s(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\right]\\ &+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s} \sigma'_n(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\right] \quad (13b)\end{aligned}$$

$$\begin{aligned}\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{-})\int_{\overset{j}{\Delta}_s} u'_s(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] &= -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{-})\int_{\overset{j}{\Delta}_s} u'_n(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]\\ &+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{-})\int_{\overset{j}{\Delta}_s} \sigma'_s(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]\\ &+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{-})\int_{\overset{j}{\Delta}_s} \sigma'_n(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] \quad (13c)\end{aligned}$$

$$\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_s^{-})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\Big] = -\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_n^{-})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_s^{-})\int_{\overset{j}{\Delta}_s} \sigma_s'(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_n^{-})\int_{\overset{j}{\Delta}_s} \sigma_n'(\overset{i}{F}{}_s^{-},\nu,E,x,y)dS\Big] \tag{13d}$$

$$\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_s^{+})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] = -\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_s^{+})\int_{\overset{j}{\Delta}_s} \sigma_s'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s} \sigma_n'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] \tag{13e}$$

$$\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_s^{+})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\Big] = -\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_s^{+})\int_{\overset{j}{\Delta}_s} \sigma_s'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s} \sigma_n'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\Big] \tag{13f}$$

$$\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_s^{-})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] = -\sum_{j=1}^{n}\Big[(\overset{j}{\sigma}{}_n^{-})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_s^{-})\int_{\overset{j}{\Delta}_s} \sigma_s'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] + \sum_{j=1}^{n}\Big[(\overset{j}{u}{}_n^{-})\int_{\overset{j}{\Delta}_s} \sigma_n'(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\Big] \tag{13g}$$

$$\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{-})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\right] = -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{-})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{-})\int_{\overset{j}{\Delta}_s} \sigma_s'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{-})\int_{\overset{j}{\Delta}_s} \sigma_n'(\overset{i}{F}{}_n^{-},\nu,E,x,y)dS\right], \quad (13h)$$

where the positive and negative signs indicate the relationship between the orientation of the i^{th} boundary surface to that of the j^{th}, $\overset{i}{F}$ is a concentrated force, ν is Poisson's ratio, E is Young's modulus, and X-Y are position coordinates.

This system of equations can be reduced by examining the geometry of randomly oriented elements. The components of the distance, within the i^{th} element's coordinate system, between the positive side of the i^{th} element and the j^{th} element are constant. These distances are the same for the negative side, but they have a negative sign due to the 180^o rotation of the base coordinate system. Using this information, it can be shown that equation ($13a$) is equivalent to ($13b$), ($13c$) is equivalent to ($13d$), ($13e$) is equivalent to ($13f$), and ($13g$) is equivalent to ($13h$). Further, by applying the double-sided boundary constraints, the final system of equations is:

$$\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{+})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] = -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{+})\int_{\overset{j}{\Delta}_s} \sigma_s'(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s} \sigma_n'(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] \quad (14a)$$

$$\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{-})\int_{\overset{j}{\Delta}_s} u_s'(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] = -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s} u_n'(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{-})\int_{\overset{j}{\Delta}_s}\sigma'_s(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right]$$

$$-\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s}\sigma'_n(\overset{i}{F}{}_s^{+},\nu,E,x,y)dS\right] \quad (14b)$$

$$\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{+})\int_{\overset{j}{\Delta}_s}u'_s(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right] = -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s}u'_n(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{+})\int_{\overset{j}{\Delta}_s}\sigma'_s(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s}\sigma'_n(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right] \quad (14c)$$

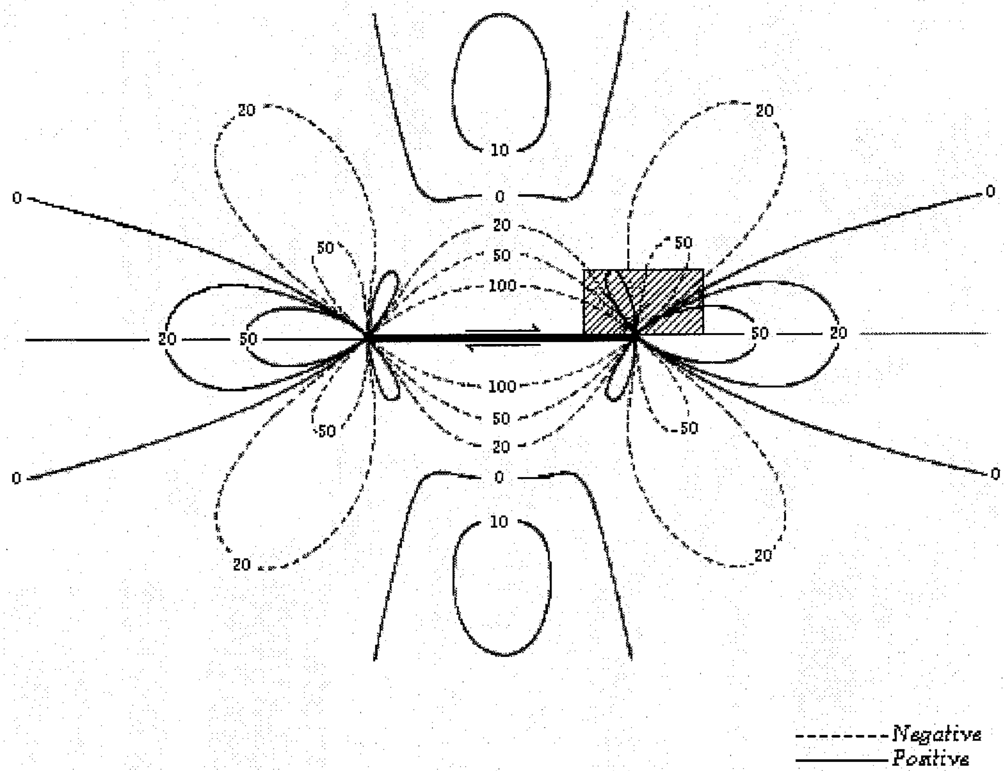
$$\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_s^{-})\int_{\overset{j}{\Delta}_s}u'_s(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right] = -\sum_{j=1}^{n}\left[(\overset{j}{\sigma}{}_n^{+})\int_{\overset{j}{\Delta}_s}u'_n(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right]$$

$$+\sum_{j=1}^{n}\left[(\overset{j}{u}{}_s^{-})\int_{\overset{j}{\Delta}_s}\sigma'_s(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right]$$

$$-\sum_{j=1}^{n}\left[(\overset{j}{u}{}_n^{+})\int_{\overset{j}{\Delta}_s}\sigma'_n(\overset{i}{F}{}_n^{+},\nu,E,x,y)dS\right]. \quad (14d)$$

Validation

To test this system, a straight zero-strength fault under 100 MPa of stress which extends from the X-Y coordinates (-20,20) to (20,-20) is assembled. The magnitude of the driving force was chosen as it is an estimate of the quantity of stress thought necessary for the observed uplift in the region (3000 m). The off-fault stress distribution is propagated for a 81x81 grid (2 kilometer spacing) extending from the lower-left X-Y coordinate of (-80,-80) to the upper right coordinate of (80,80). The shear and normal displacement (Figure 5a), shear stress parallel to the fault (Figure 5b), the volumetric stress (a measure of the isotropic compression or tension which is the equivalent to the trace of



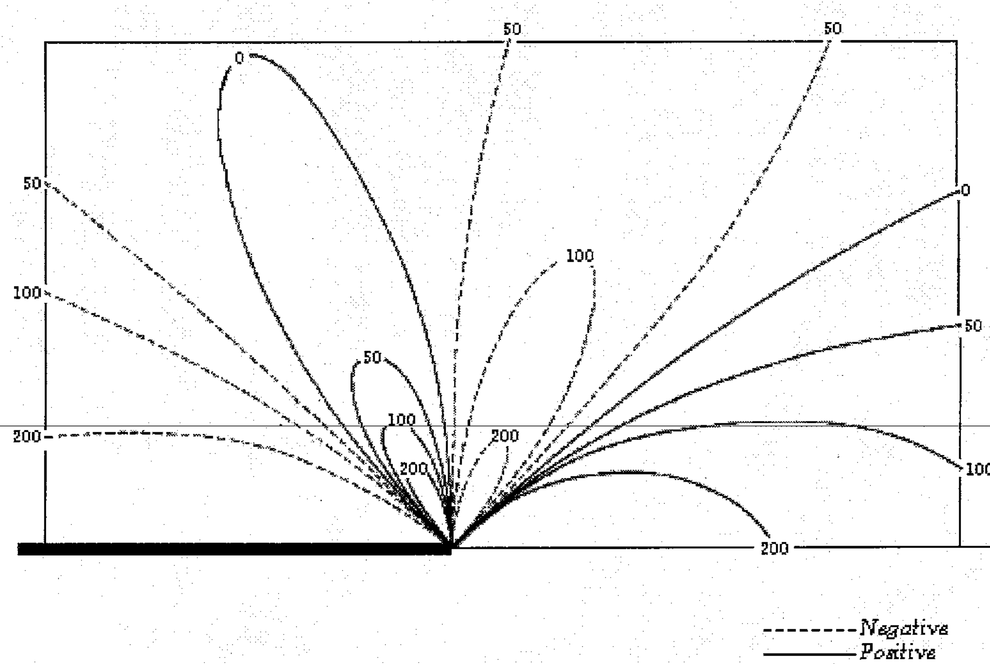


Figure 4a: The shear traction change caused by a slip discontinuity along a fault (after Figure 2, Chinnery (1963)). The lower figure is an enlargement of the shaded portion of the upper figure. Areas with negative values represent stress dissipation whereas positive values represent stress accumulation.

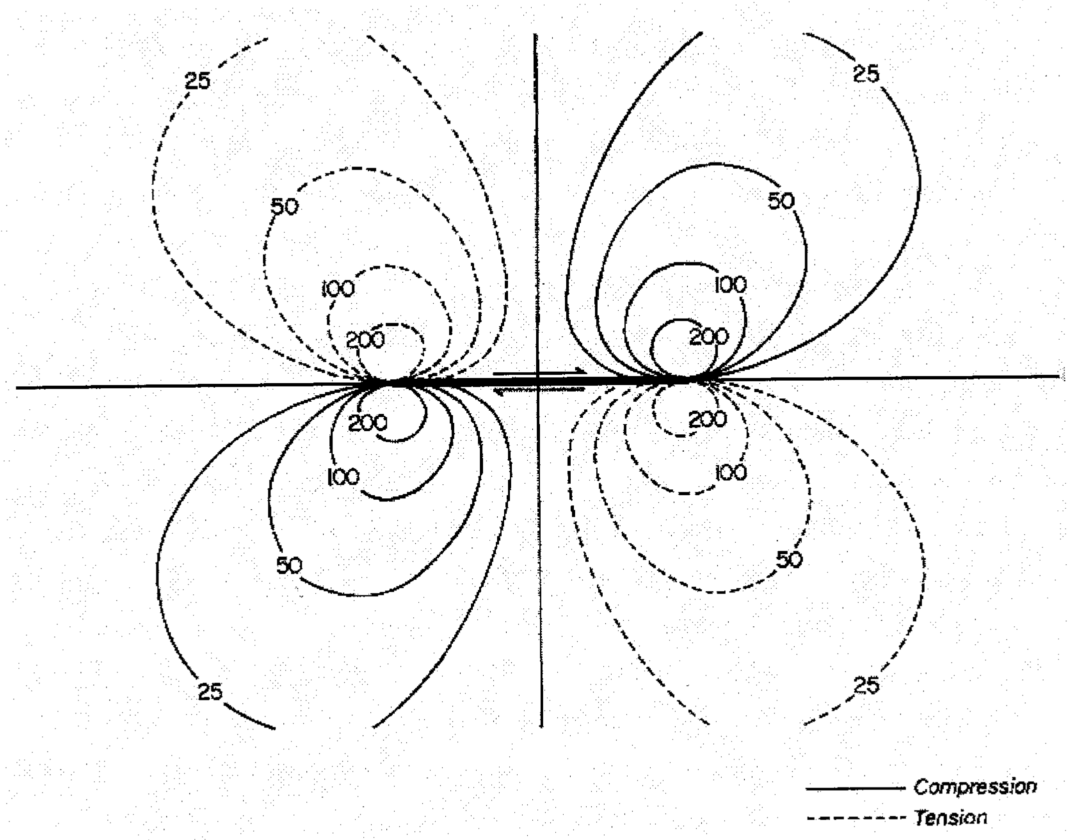


Figure 4b: The volumetric stress parallel to a fault with a given slip discontinuity (after Figure 3, Chinnery (1963)). Areas with negative values represent compressional stress and positive values dilatational stress.

the stress tensor) (Figure 5c), and the most (Figure 5d) and least (Figure 5e) compressive principle stress directions and orientations are plotted. At first glance it is clear that the numerical instability problem has been overcome.

The scale-independent shear and normal displacement distribution along the fault appears to be acceptable (Figure 5a). This fault has approximately .13 units of relative movement along fault. This value is slightly elevated near the crack tips due to the non-physical stress singularity found in elastic theory. The normal displacement is approximately zero near the middle of the fault and it gradually increases to .03 units near the crack tips. This tendency of the fault to rotate into the direction of the regional stress field (Hanson *et al.*, 1971) adds more realism to the fault model than the standard boundary element algorithm of Crouch and Starfield (1990).

However, to validate these new equations, comparison of the results is made with previous work by Chinnery (1963) which addressed the stresses associated with a right lateral strike-slip dislocation surface. The plot of shear stress drop distribution parallel to the fault (Figure 5b) correlates well with previous analytical solutions (Das & Scholz 1982; Chinnery 1963). In the plot, negative values represent zones of stress dissipation while positive values represent stress accumulation.

As predicted by Chinnery, a number of locations consistent with the boundary values that characterize the significant stress patterns have been identified. The area on either side of the fault has undergone a complete shear stress release. This zone tapers to zero stress drop in a circular fashion within a radius of about half the length of the fault (approximately (-30,30) to (30,30)). However, the shape of the zero stress contour varies significantly from that calculated by Chinnery as it has four lobes which connect from the crack tips to the zero stress contour. This behavior arises because there is no zero normal displacement condition placed upon the fault and therefore the crack tips rotate as described previously. As expected, the stress singularity of the fault tip creates areas of significant stress accumulation to occur beyond the fault along strike. The four lobes of stress dissipation found on either side of the crack tips (approximately (-40,10), (-10,40), (10,-40), and (40,-10)) are due to localized isotropic compression and tension of material (Chinnery, 1963). Just inside of the fault crack tips is a zone of anomalously high stress drop on the order of 100 MPa. This arises because the material at the crack tip desires to propagate in the same sense that the rest of the fault does: in opposite

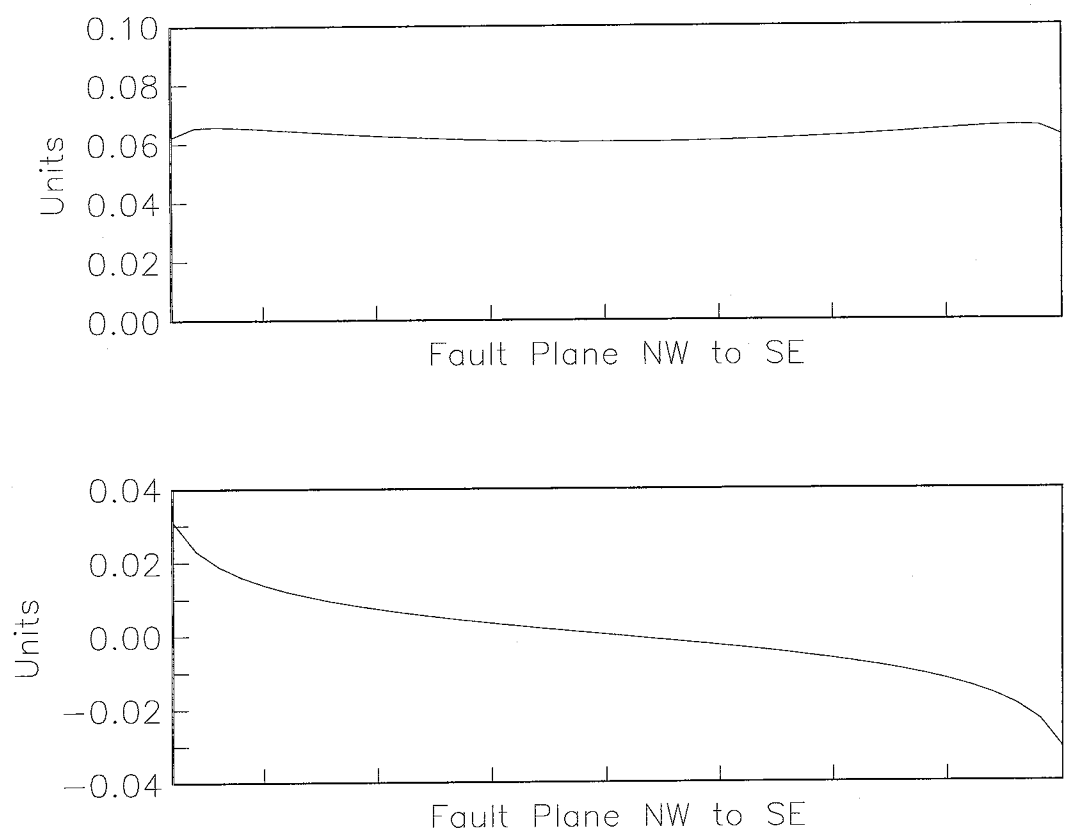


Figure 5a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a zero-strength fault extending from the X-Y coordinates of (-20,20) to (20,-20). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The elevated shear displacement near the edge of the fault is the result of the non-physical elastic theory stress singularity at the crack tips. The normal displacement is characterized by the tendency of the crack tips to rotate towards the direction of the regional stress (Hanson *et. al*, 1971).

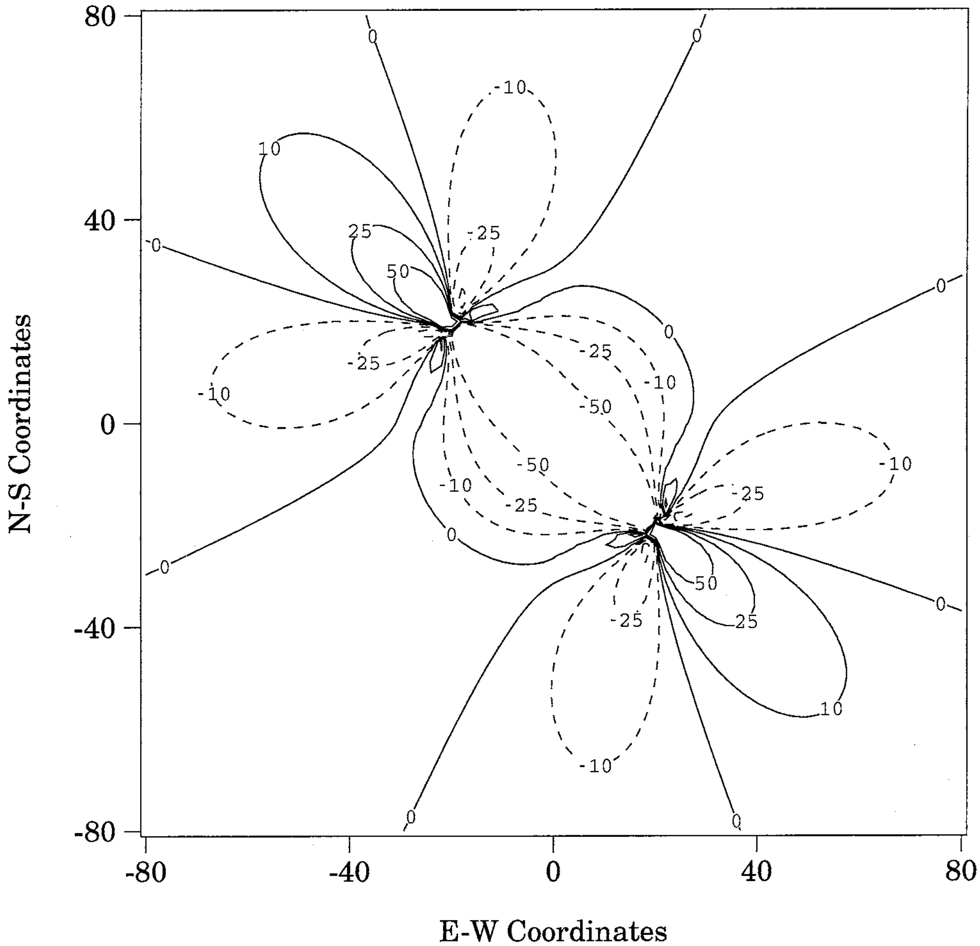


Figure 5b: The shear traction change parallel to a zero-strength fault which extends from the X-Y coordinates of (-20,20) to (20,-20). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. As expected, there is complete stress drop located for approximately 10km on either side of the fault. This zone tapers to zero within around half a fault length. There are large zones of stress accumulation in the direction of fault propagation.

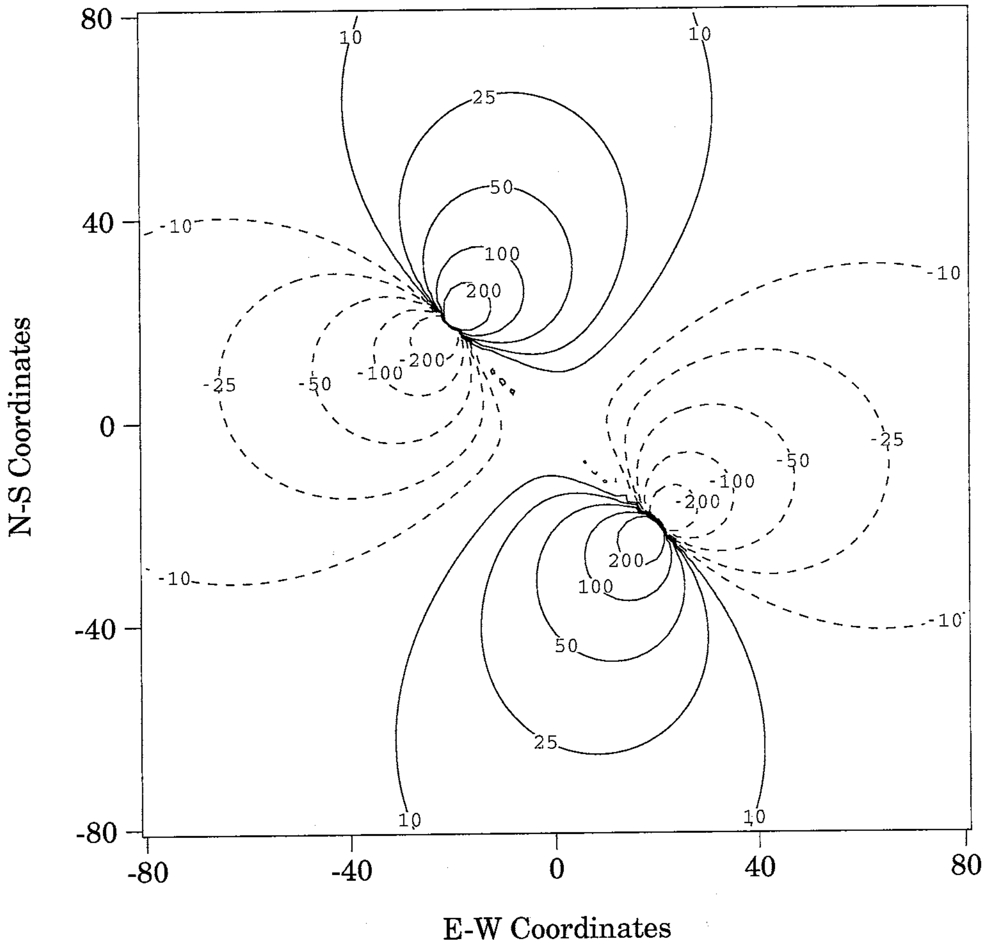


Figure 5c: The volumetric stress parallel to a zero-strength fault which extends from the X-Y coordinates of (-20,20) to (20,-20). Areas with negative values represent compressional stress and positive values dilatational stress. As expected, the features are anti-symmetric about the fault.

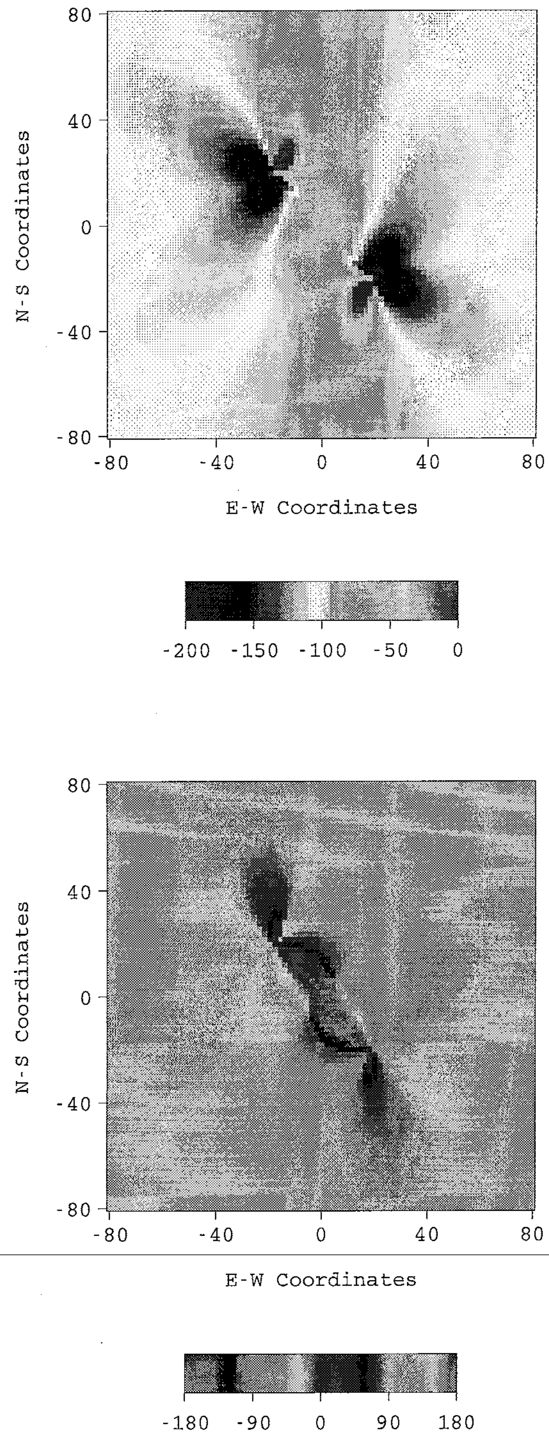


Figure 5d: The most compressive stress magnitudes (upper) and orientation (lower) for a zero-strength fault extending from the X-Y coordinates of (-20,20) to (20,-20). In general, the orientation of these stresses roughly coincides with the regional trends of north-south for the most compressive and east-west for the least compressive stresses.

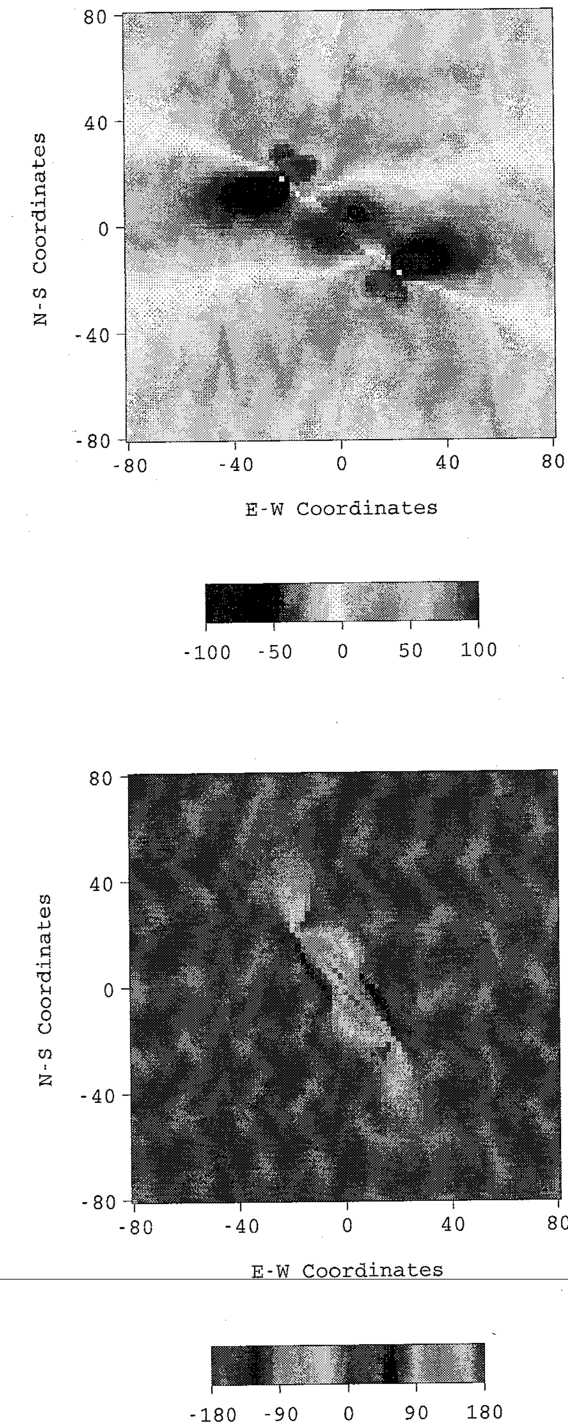
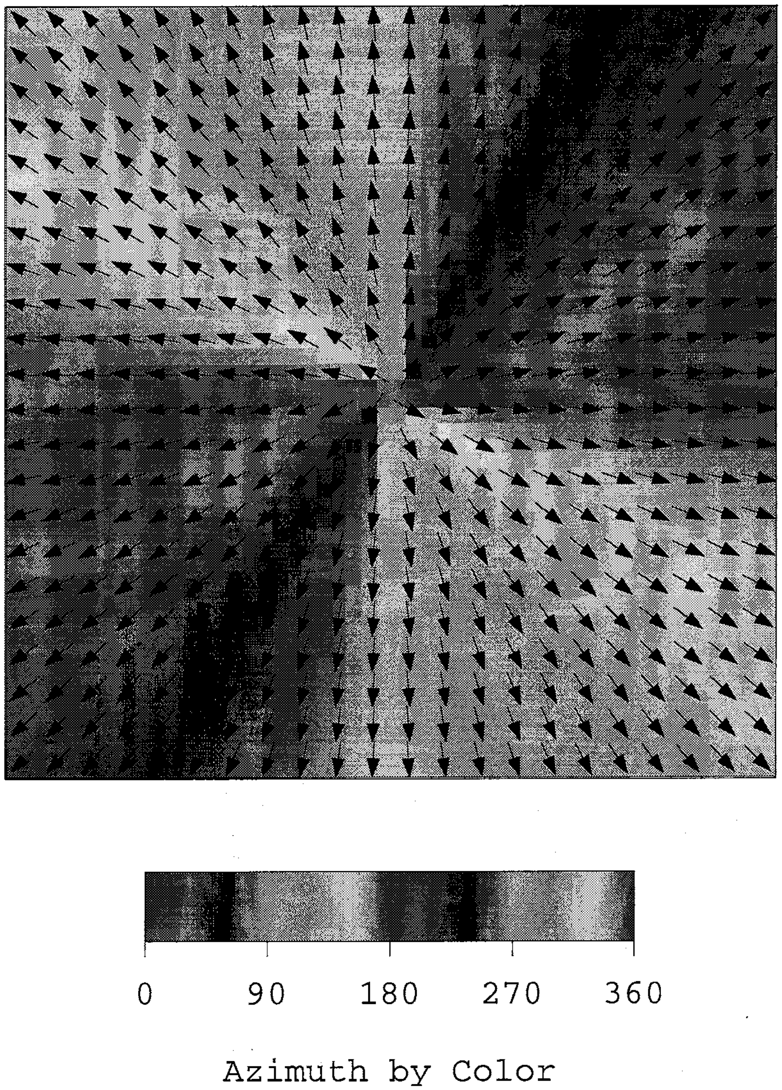


Figure 5e: The least compressive stress magnitudes in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) for a zero-strength fault extending from the X-Y coordinates of (-20,20) to (20,-20). In general, the orientation of these stresses roughly coincides with the regional trends of north-south for the most compressive and east-west for the least compressive stresses.



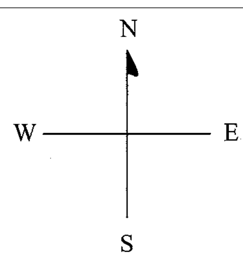


Figure 6: Azimuthal color wheel where the colors on the wheel denote principle stress orientation and where opposite azimuths have the same color. The overlying vectors point roughly towards the direction represented by each color. The color method was chosen over vectors due to the loss of resolution.

directions simultaneously. In static elastic theory, cracks are not allowed to propagate, so the shear displacement is singular in nature. Because of this, the shear stresses become infinite. In the boundary element method, these crack tips are smeared out to the length of an element. This has the effect of raising the shear stress to quite large values, while avoiding the infinite stress. Since faults are low strength, the crack tips, both at the ends of faults and when a fault bends, are the dominant stress features found using this method.

The volumetric stress also correlates well with the results of Chinnery (Figure 5c). In this plot, negative values represent compressional stress and positive values dilatational stress. The lower magnitude stress features observed are four teardrop shaped lobes. As the magnitude of the stress increases, the lobes become increasingly circular and orient themselves tangentally to the crack tip. These compressional and extensional lobes are antisymmetric about the fault and clearly display its dextral sense. In addition to these results predicted by Chinnery, two minor oblong lobes cause by the rotation of the crack tips occur parallel to the fault. In summary, the correlation of the shear and volumetric stresses have confirmed that the new system of equations is consistent with intuitive expectations and previous analytic solutions.

Rotating the previously described stresses into principle direction yields interesting results as well (Figure 5d and 5e). In the stress magnitude plots, negative values are compressive and positive values are tensile. The orientations of the stresses are plotted on an "azimuthal" color wheel (Figure 6.) In this representation, red colors approximately correspond to the east and west azimuths, purples through blues to the northeast and southwest azimuths, and yellows through greens to the northwest and southeast azimuths. The azimuthal color wheel representation was chosen in order to display maximum detail on the 81x81 grid of orientations without overwhelming the plot with overlapping directional indicators.

The regional trend of the simulations is 100 MPa of north-south compression and a 0 MPa of east-west compression. Although dominated by the regional trend (approximately white), the most compressive principle stress (Figure 5d) has significant perturbations. The stress near the crack tips varies from approximately zero to over twice the background level. The most pervasive perturbation occurs north and south of the fault. This stress is greatly reduced from the regional values and is caused by the fault slipping so as to dissipate its shear stress. The least compressive stress magnitude (Figure 5e) affected over a broader region than its counterpart. The most significant variation of this stress is orthogonal to that of the most compressive stress. Areas of elevated compressional stress exist east and west of the fault.

Unfortunately, these results cannot be completely correlated to the inversion of focal mechanisms at Anza by Hartse *et al.*, 1994). This arises because the method assumes that the principle stresses are continuous for a region (*e.g.*: Gephart and Forsyth, 1984; Jones, 1988; Hartse *et al.*, 1994). The basic elastic theory associated with the boundary element method employed in this algorithm has a discontinuity in principle stress across the fault. This can be seen by examining the two-dimensional stress tensor near either end of the fault (Figure 7). Since the volumetric stress is symmetric and the shear stress is anti-symmetric, the diagonal of the tensor on one side of the fault is the negative of the other. Decomposition of these tensors into the principle domain yields distinctly different orientations of the stresses. Therefore, rather than attempting to strictly match the observations of Hartse *et al.* (1994), our results will be used as end-member characterizations of seismic gap phenomenon.

Results

In order to examine possible scenarios explaining the existence of the Anza slip and seismic gaps as well as possible causes for associated the clustered seismicity, a number of models will be examined in order of increasing complexity. In particular, the models will be shown as an 81x81 section of an infinite plate, as described previously. The plate used is characterized by an P-wave velocity of 6100 meters per second, an S-wave velocity of 3600 meters per second (Scott, 1992), and an assumed density of 2750 kg per cubic meter. These values translate into the elastic constants of 87.87 GPa for Young's modulus and 0.233 for a Poisson's ratio. For comparison, Young's modulus becomes 95.86 GPa for a density of 3000 kg per cubic centimeter and 79.88 GPa for 2500 kg per cubic centimeter. Poisson's ratio does not change for these changes in density. The first eight models examine abrupt (Figure 8) and smooth (Figure 9) variations in shear strength along fault. slip gaps.

Strong Gaps

Strong Gaps, or central zones of reduced stress drop relative to peripheral regions, can be considered areas where where seismic slip cannot easily occur. These zones could be the result of relatively elevated stresses normal to the

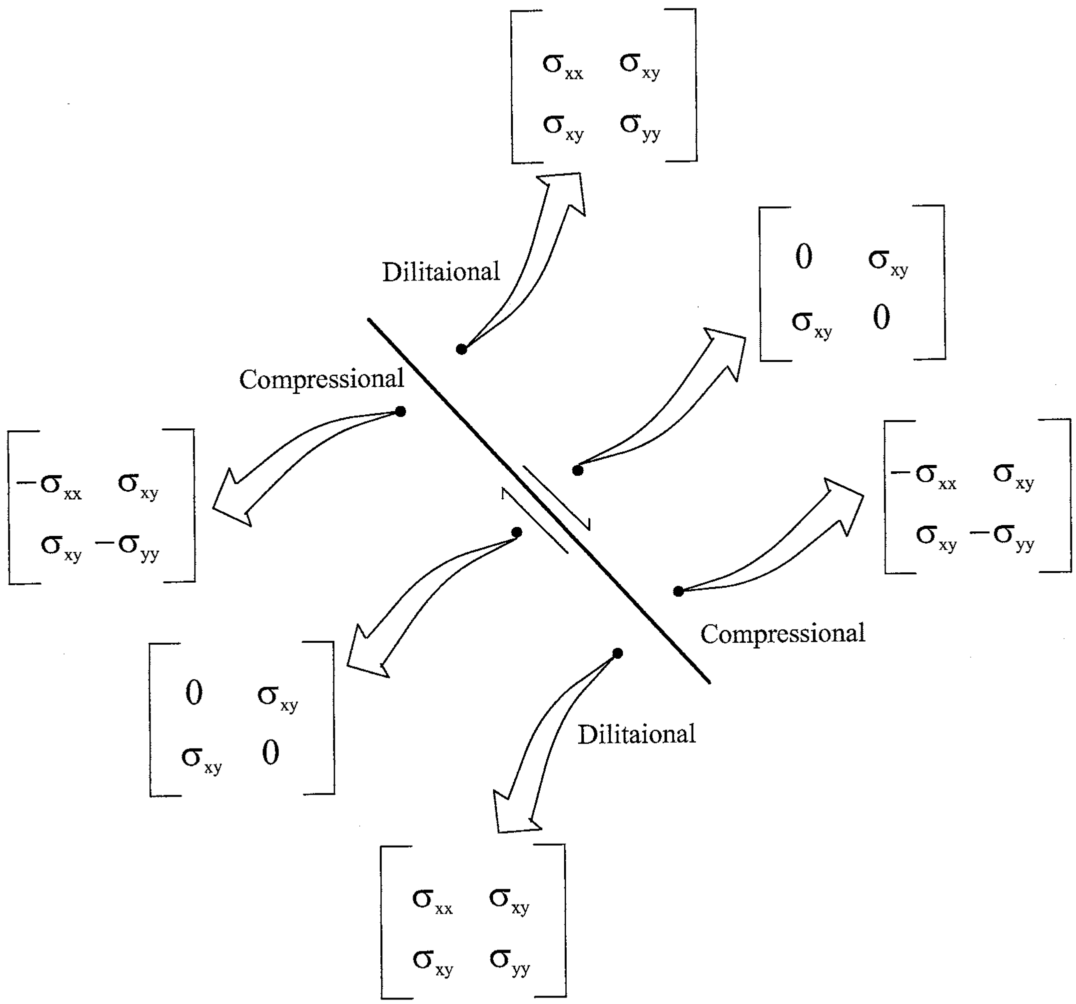


Figure 7: Two-dimensional static stress tensors for a theoretical zero-strength fault under constant shear stress (assuming the elastic behavior of a symmetric shear stress and an anti-symmetric volumetric stress). The tensors are displayed at points placed an infinitesimal distance from the fault. The decomposition of these tensors into principle stresses displays a discontinuity in orientation for any transverse across the fault, except in the center. Under closer examination, it is observed that the stress orientation on either side of the fault approaches the fault-parallel stress orientation off the fault via opposed rotations (one clockwise and the other counter). Therefore, the results from our boundary element program cannot be directly compared with studies which assume that this discontinuity has been smoothed through stress relaxation (*e.g.*: Gephart and Forsyth, 1984; Jones, 1988; Hartse *et al.*, 1994).

fault or the onset of more competent material along the strike of the fault. If this were the mechanism operating in the Anza region of the SJFZ, we might expect to see a reduction in the seismic activity if the region were locked. However, a more competent material might elevate the average magnitude of events compared to surrounding areas.

The models shown in Figures 8(C), 8(D), 9(C), and 9(D) characterize this situation. Model 8(C) is a fault with constant 10 MPa strength except in the gap, where the strength increases to 20 MPa. Given the driving force used in these models, this is the equivalent to a fault having the ability to resolve 80% of the applied shear stress except in the gap, where it resolves only 60%.

The displacement fields (Figure 12a) calculated for this model are as expected. The shear displacement quickly rises from the crack tips to a constant value. At the onset of the gap it drops by approximately one-third its original value. The normal displacement is characterized by severe twisting at the crack tips of over 50% of the shear displacement. A minor cusp of twisting oriented in the opposite direction of the tips occurs at the transition to the gap. This is because the transition of strength is from higher to lower at the gap and from lower to higher at the crack tips. The shear traction drop along planes parallel to the fault (Figure 12b) has a number of interesting features. The gap itself has zero change in traction and large edge affects near the strength transition. The traction drop bulges around the gap within the circle defined by a radius equal to the length of the fault. This material is required to resolve the stress accumulation caused by the gap. The volumetric stress (Figure 12c) is very similar to that of a fault without a gap excepting the stress caused by the strength transition. Similarly, the principle stresses (Figures 12d and 12e) vary only slightly from a fault of constant strength as well. This difference primarily lies near the strength transition of the gap where two lobes of elevated stress occur. In addition, the deviations of the orientations of the stress from the regional trend are slightly more pronounced within the circle created by the length of the gap.

Comparing the previous model 8(C) to that of 8(D), where the gap resolves only 40% of the applied shear stress, emphasizes the features of a strong gap. The shear displacement (Figure 13a) is equivalent except in the gap, where the added strength has reduced it relative to the background twice as much as in model 8(C). Similarly, the normal displacement is equivalent everywhere except at the strength transition of the gap. Here the localized twisting cups have increased in magnitude. The shear traction drop along planes parallel to the fault (Figure 13b) is significantly different than for 8(C). The shear traction has doubled within the gap at the expense of primarily the area within a circle of one gap radius and less significantly within one fault length circle centered on the gap. The stress accumulation in the direction of fault propagation is only very slightly affected. The volumetric stress (Figure 13c) with respect to model 8(C) displays a significant stress magnitude increase at the edge of the gap with a corresponding reduction at the ends of the fault. The principle stresses (Figures 13d and 13e) are also remarkably different from the previous model. The magnitude of the most and least compressive stresses have become dramatically more compressive while their deviation from the regional trend has become only slightly more pervasive.

In an attempt to eliminate the overbearing affect of the singularity at the crack tips, models 8(C) and 8(D) were smoothed to create models 9(C) and 9(D) respectively. Although the smoothing will alter the average strength of the fault, these changes will only cause linear variations in the calculated stress and displacement fields. Therefore they will be comparable to the abrupt changes in strength.

The behavior of these models is extremely similar for both 8(C) to 9(C) and 8(D) to 9(D), so they will be discussed simultaneously. The shear displacement (Figures 16a and 17a) has much smoother transitions and is slightly greater in the 9(C,D) than in 8(C,D). However the moment is, to first approximation, the same. The normal displacement (Figures 16a and 17a) is smoothed at the strength transitions and the twisting of the fault is far less apparent except for an unusual twist at the crack tip. The displacement is reversed with respect to the boxcar models at this point. The contours of the shear traction drop along planes parallel to the fault (Figures 16b and 17b) do not change as rapidly and are much more rounded than the boxcar models. There is also a greater accumulation of stress at the crack tips in the abrupt models. There is also a much smaller are of stress accommodation. In contrast, the volumetric stresses (Figures 16c and 17c) are larger in magnitude because of the reduction of normal displacement accommodation. The principle stresses (Figures 16d, 16e, 17d, and 17e) display smearing into a more even distribution of the stresses over the area parallel to the fault. The extent of the deviation of their orientations from the regional value has become less circular and much more pervasive along the fault. Otherwise the general observations of the boxcar models 8(C,D) do not make any remarkable deviations. The smoothed models are, in general, less chaotic and more geophysically realistic.

Weak Gaps

Weak Gaps, or central zones of increased stress drop relative to peripheral regions on the fault, can be considered areas where where seismic creep is the dominant mechanism for stress release. These areas could arise due to relatively lower stresses normal to the fault, or the onset of less competent material along the strike of the fault. Specifically, if this were the case in the Anza region, this model could explain both the lack of seismicity as well as the observed slip deficit. However, this scenario seems more unlikely than the strong gap hypothesis because there is no observed creep along this section of the SJFZ (Johnson, 1993).

The models shown in Figures 8(A), 8(B), 9(A), and 9(B) characterize this behavior. Model 8(A) is a fault with 20 MPa strength everywhere except in the gap, where the strength decreases to 10 MPa. Given the driving force used in these models, this is the equivalent to a fault having the ability to resolve 60% of the applied shear stress, except in the gap where it resolves 80%.

The displacement fields (Figure 10a) calculated for this model are as expected. The shear displacement quickly rises from the crack tips to a constant value. At the onset of the gap it increases by approximately one-third its original value. The normal displacement is characterized by severe twisting at the crack tips of over 50% of the shear displacement. A minor cusp of twisting oriented in the same direction of the tips occurs at the transition to the gap. The shear traction drop along planes parallel to the fault (Figure 10b) has a number of interesting features. Nearly the entire fault is characterized by 100% traction drop. The gap accommodates the stress through displacement and therefore the material surrounding the gap is not required to resolve as much stress. The volumetric stress (Figure 10c) is very similar to that of a fault without a gap excepting the stress caused by the strength transition. The principle stresses (Figures 10d and 10e) vary only slightly from a fault of constant strength as well. This difference primarily lies near the strength transition of the gap where two lobes of elevated stress occur. In addition, the deviations of the orientations of the stress from the regional trend are slightly more pronounced within the circle created by the length of the gap.

Comparing the previous model 8(A) to that of 8(B), where the main portion of the fault resolves only 40% of the applied shear stress, emphasizes the features of a weak gap. The shear displacement (Figure 11a) is equivalent only in the gap because the increased relative weakness of the fault has reduced its movement relative to model 8(A). The normal displacement has reduced everywhere especially at the strength transition of the gap. The shear traction drop along planes parallel to the fault (Figure 11b) is significantly different than for 8(A). The zone of stress accumulation has pinched a great deal with respect to model 8(A) while the stress drop in the gap has remained the same. This is due to the increased strength on the majority of the fault. Nevertheless, this does not affect the comparison of these models because they are linear and only their ratios matter. The stress accumulation in the direction of fault propagation is only very slightly affected. The volumetric stress (Figure 11c) with respect to model 8(A) displays a significant stress magnitude decrease at the edge of the gap with a corresponding reduction at the ends of the fault for the same reasons mentioned for the shear traction drop. The principle stresses (Figures 11d and 11e) are also remarkably different from the previous model. The magnitude of the most and least compressive stresses have become dramatically more compressive in the gap and less so at the crack tips. The deviation of these stress orientations from the regional trend has become only slightly less pervasive.

In an similar attempt as the strong gap simulations, the overbearing affect of the singularity at the crack tips is eliminated by smoothing , models 8(C) and 8(D) in order to to create models 9(C) and 9(D) respectively. Again, the smoothing will alter the average strength of the fault, but these changes will only cause linear variations in the calculated stress and displacement fields. Therefore they will be comparable to the abrupt change in strength models.

The behavior of these models is extremely similar for both 8(A) to 9(A) and 8(B) to 9(B), so they will be discussed simultaneously. The shear displacement (Figures 14a and 15a) has much smoother transitions and is slightly greater in the 9(A,B) than in 8(A,B). However the moment is, to first approximation, the same. Similarly the normal displacement (Figures 14a and 15a) is smoothed at the strength transitions and the twisting of the fault is far less apparent. The contours of the shear traction drop along planes parallel to the fault (Figures 14b and 15b) do not change as rapidly and are much more rounded than the boxcar models. There is also a greater accumulation of stress at the crack tips. In contrast, the volumetric stress (Figures 14c and

15c). The stresses are larger in magnitude, however because of the reduction in normal displacement accommodation. The principle stresses (Figures 14d, 14e, 15d, and 15e) display smearing into a more even distribution of the stresses over an area parallel to the fault. The extent of the deviation of their orientations from the regional value has become less circular and much more pervasive along the fault. Otherwise the general observations do not make any remarkable changes. In general, the smoothed models are less chaotic and more geophysically realistic.

Conclusions

Algorithm

- The results from this algorithm compare favorably to the dislocation discontinuity calculations of Chinnery (1963) and those of Das and Scholz (1982).
- We have improved the mixed value problem solution commonly used by altering the boundary conditions to better describe a strike-slip system. Thus the calculation of the stresses induced by faults within a regional driving force is allowed.
- The stress calculations are more realistic than the typical simple shear displacement driven model in the sense that they allow crack tip rotation.
- The results cannot directly compare to focal mechanism inversions based upon the method developed by Gephart and Forsyth, (1984) because general elastic theory has a discontinuity in principle stress across a fault.

Strong Gaps

- A significant increase in stress accumulation occurs within the gap. As the relative change in strength between the gap and the peripheral portion of the fault increases, this accumulation can be several times the size of the original regional driving stress. The gap acts as a stress concentrator.
- These models are characterized by a reduction in the area of stress dissipation within one-half of the fault radius of the gap. Again, the gap is acting to concentrate stress.
- A significant increase in volumetric stress occurs. Therefore, the question of causality arises: does increased volumetric stress cause a locked gap, or vic-versa?
- Stress magnitude and orientation perturbations are restricted to a circle of a radius equal to the length of the fault.

Weak Gaps

- A significant decrease in stress accumulation occurs in the weak gap models. This indicates that the gap acts as a stress guide such that it concentrates stress field heterogeneity in the vicinity of the fault.
- A reduction in stress accumulation in the vicinity of the crack tip also indicates that the gap accommodates additional stress.
- Stress magnitude and orientation perturbations are restricted to a circle of a radius equal to the length of the fault.

Refinements:

- Modeling the state of stress after appreciable time following an event requires the application of a diffusion algorithm to the results of this method.
- The reduction of crack-tip effects would make the solution more robust and realistic for geophysical problems.
- Modeling of non-strike-slip faults requires a 3-dimensional model.

References

Abe, K., *Magnitudes and Origin Times from Milne Seismograph Data: Earthquakes in China and California* Historical Seismograms and Earthquakes of the World, edited by W.H.K. Lee, H. Meyers and M. Shimazaki, Academic Press, London, Pp. 37-50, 1988.

Billham, Roger and Geoffrey King, *The Morphology of Strike-Slip Faults: Examples from the San Andreas Fault, California,* JGR, Vol. 94, No. B8, Pp. 10204-10216, 10 August 1989.

Chinnery, M.A., *The Stress Changes that Accompany Strike-Slip Faulting,* BSSA, Vol. 53, No. 5, Pp. 921-932, October 1963.

Crouch, S.L. and A.M. Starfield, *Boundary Element Methods in Solid Mechanics,* Cambridge University Press, 1990.

Cruse, Thomas, A., *Boundary-Integral Equation Fracture Mechanics Analysis,* Boundary-Integral Equation Method: Computational Applications in Applied Mechanics, AMD-Vol. 11, American Society of Mechanical Engineers, 1975.

Das, S. and C.H. Scholz, *Off-fault Aftershock Clusters Caused by Shear Stress Increase?,* BSSA, Vol. 71, Pp. 1669-1675.

Gephart, J. and W. Forsyth, *An Improved Method for Determining the Regional Stress Tensor Using E arthquake Focal Mechinism Data: Application to the San Fernando Earthquake Seque nce,* JGR, Vol. 89, 9305-9320, 1984.

Gomberg, Joan, *Seismicity and Shear Strain in the Southern Great Basin of Nevada and California,* JGR, Vol. 96, No. B10, Pp. 16383-16399, 10 September 1991.

Hanson, Merle E., Allan R. Sanford, and Ronald J. Shaffer, *A Source Function for a Dynamic Bilateral Brittle Shear Fracture,* JGR, Vol. 76, No. 14, Pp. 3375-3383, 10 May 1971.

Hartman, F., *Introduction to Boundary Elements : Theory and Applications,* Springer-Verlag, 1989.

Hartse, Hans E., Richard Aster, Michael Fehler, Jennifer Scott, and Frank Vernon, *Evidence for Small-Scale Stress Heterogeneity in the Anza Seismic Gap, Southern CA*, JGR Vol 99, No. B4, 1994.

Johnson, Hadley O., *Techniques and Studies in Crustal Deformation,* Doctoral Thesis, UCSD Scripps Institution of Oceanography, 1993.

Jones, Lucille M., *Focal Mechanisms and the state of Stress on the San Andreas Fault in Southern California,* JGR, Vol. 93 No B8, Pp 8869-8891, 10 August 1988.

Love, A.E.H, *A Treatise on the Mathematical Theory of Elasticity,* Cambridge University Press, 1927.

Sanders, Chris O. and Hiroo Kanamori, *A Seismotectonic Analysis of the Anza Seismic Gap, San Jacinto Fault Zone, Southern California,* JGR, Vol. 89, No. B7, Pp. 5873-5890, 10 July 1984.

Sanford, Allan R., *Analytical and Experimental Study of Simple Geologic Structures,* GSA Bull., Vol. 70, Pp. 19-52, 1959.

Scott, Jennifer S., *Microearthquake Studies in the Anza Seismic Gap,* Doctoral Thesis, UCSD Scripps Institution of Oceanography, 1992.

Sharp, Robert V., *San Jacinto Fault Zone in the Peninsular Ranges of Southern California,* GSA Bull., Vol. 78, Pp. 705-730, June 1967.

Sokolnikoff, Ivan Stephan, *Mathematical Theory of Elasticity,* McGraw-Hill, 1956.

Stein, Ross S, G.C.P. King, and Jian Lin, *Change in Failure Stress on the Southern San Andreas Fault System Caused by the 1992 Magnitude = 7.4 Landers Earthquake,* Science, Vol. 258, Pp. 1328-1321, 20 November 1992.

Thatcher, W., J.A. Hileman, and T.C. Hanks, *Seismic Slip Distribution Along the San Jacinto Fault Zone, Southern California, and its Implications* Geol. Soc. Am. Bull., 86, Pp. 1140-1146, 1975.

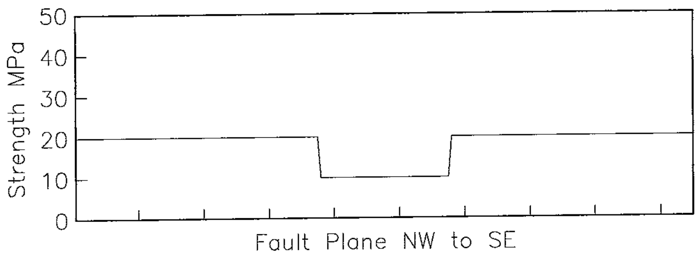
Vernon, Frank L. III, *Analysis of Data Recorded on the Anza Seismic Network,* Doctoral Thesis, UCSD Scripps Institution of Oceanography, 1989.

Wendland, W.L., *On the Asymptotic Convergence of Boundary Integral Methods,* Boundary Elements IX, Springer-Verlag, 1987.

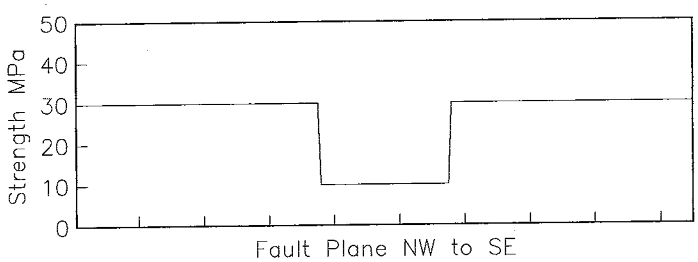
WGCEP(Working Group on California Earthquake Probabilities), *Probabilitites of large earthquakes Occuring in California on the SanAndreas Fault System,* U.S. Geol. Surv. Open File Report 88-398, 1988

Acknowledgments

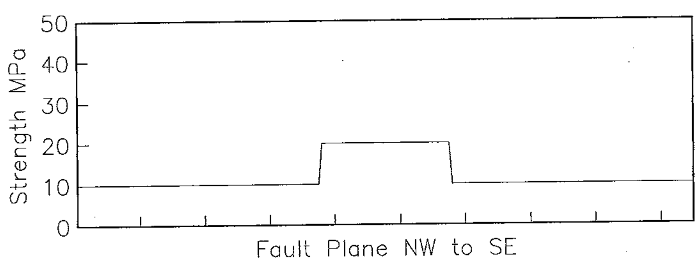
I would like to thank Allan Sanford and John Schlue for their insightful discussions on the subject. In addition, I am grateful to Terry Tafoya and Doug Warner for their help in the trials and tribulations of the personal computer world.



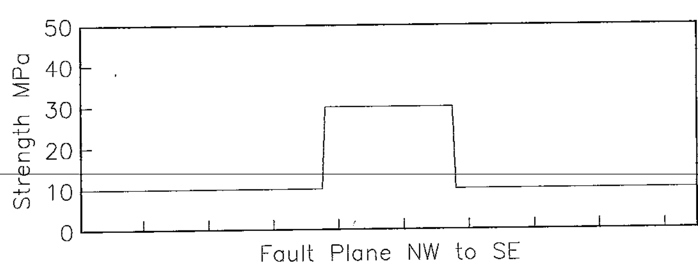
(A)



(B)

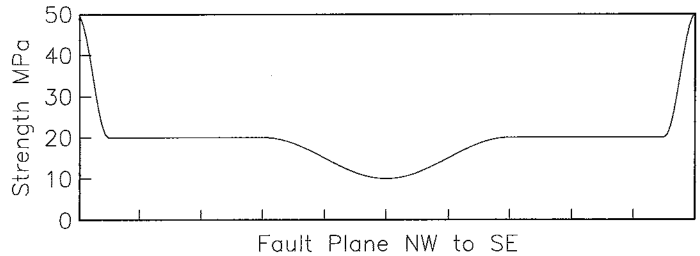


(C)

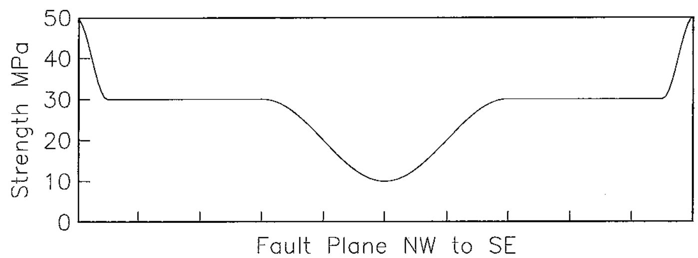


(D)

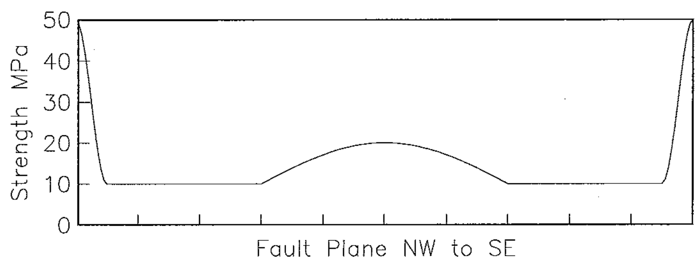
Figure 8: The fault strength (latent shear traction) for the abrupt change in strength models. The two upper models are weak gaps (Figures 10 and 11) and the two lower are strong (Figures 12 and 13). The most likely physical meaning of these gaps would be rapid changes in composition along fault.



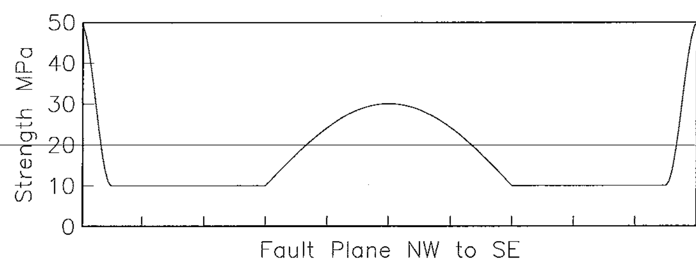
(A)



(B)



(C)



(D)

Figure 9: The fault strength (latent shear traction) for the smooth change in strength models. The two upper models are weak gaps (Figures 14 and 15) and the two lower are strong (Figures 16 and 17). The most likely physical meaning of these gaps would be slow changes in composition along fault.

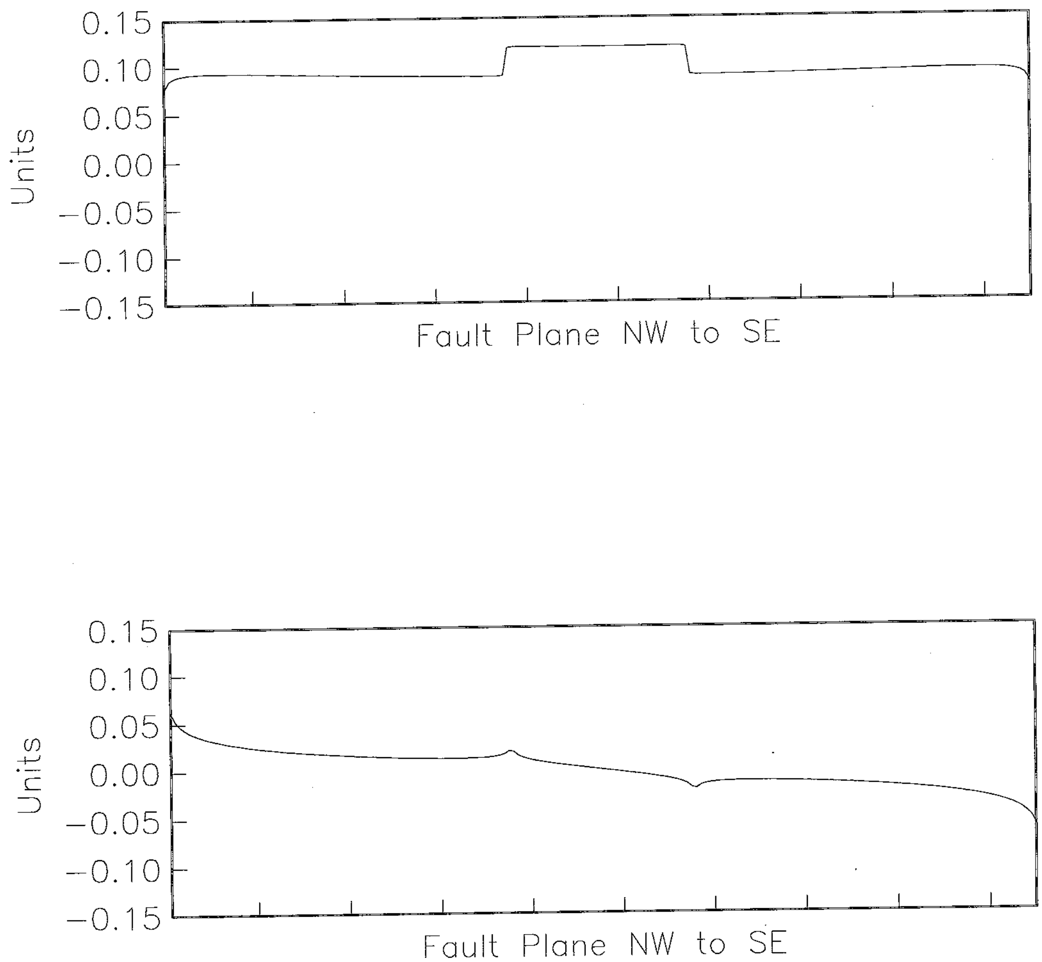


Figure 10a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The weak gap has elevated shear displacement and has small cusps of twisting near the strength transition.

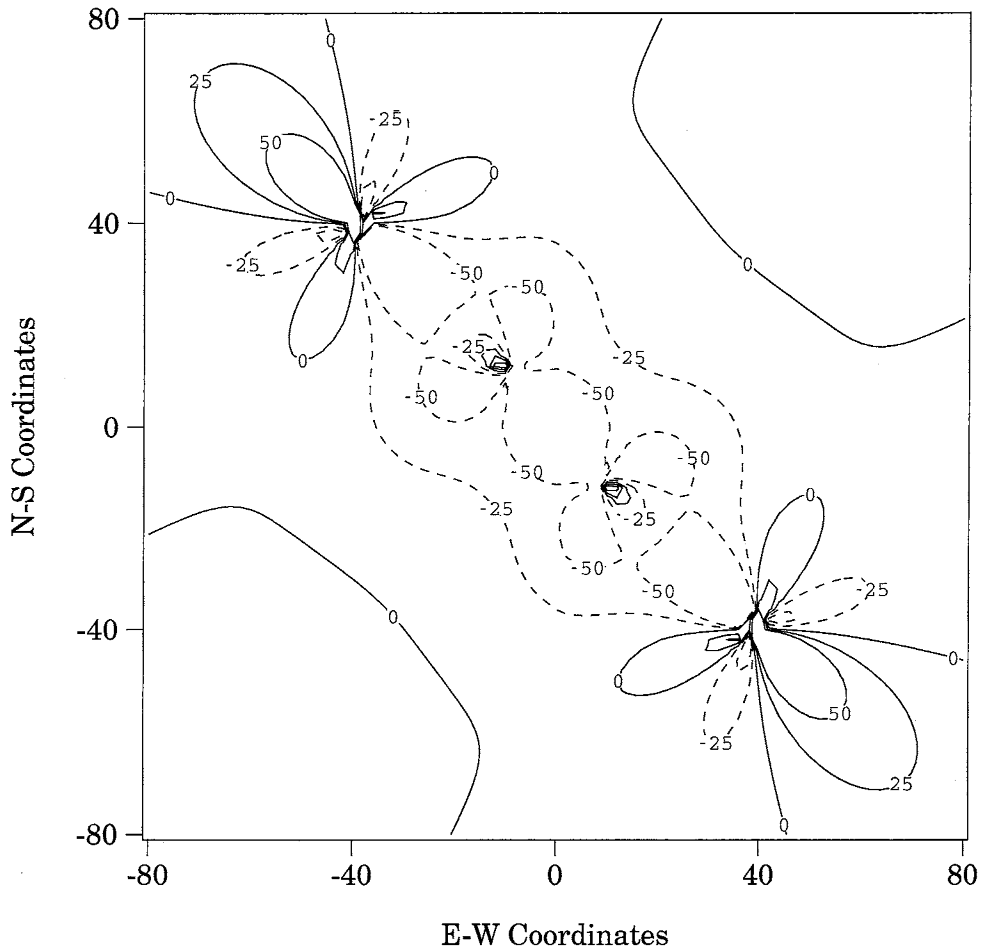


Figure 10b: The shear traction change parallel to a weak gap (Figure 8(A)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

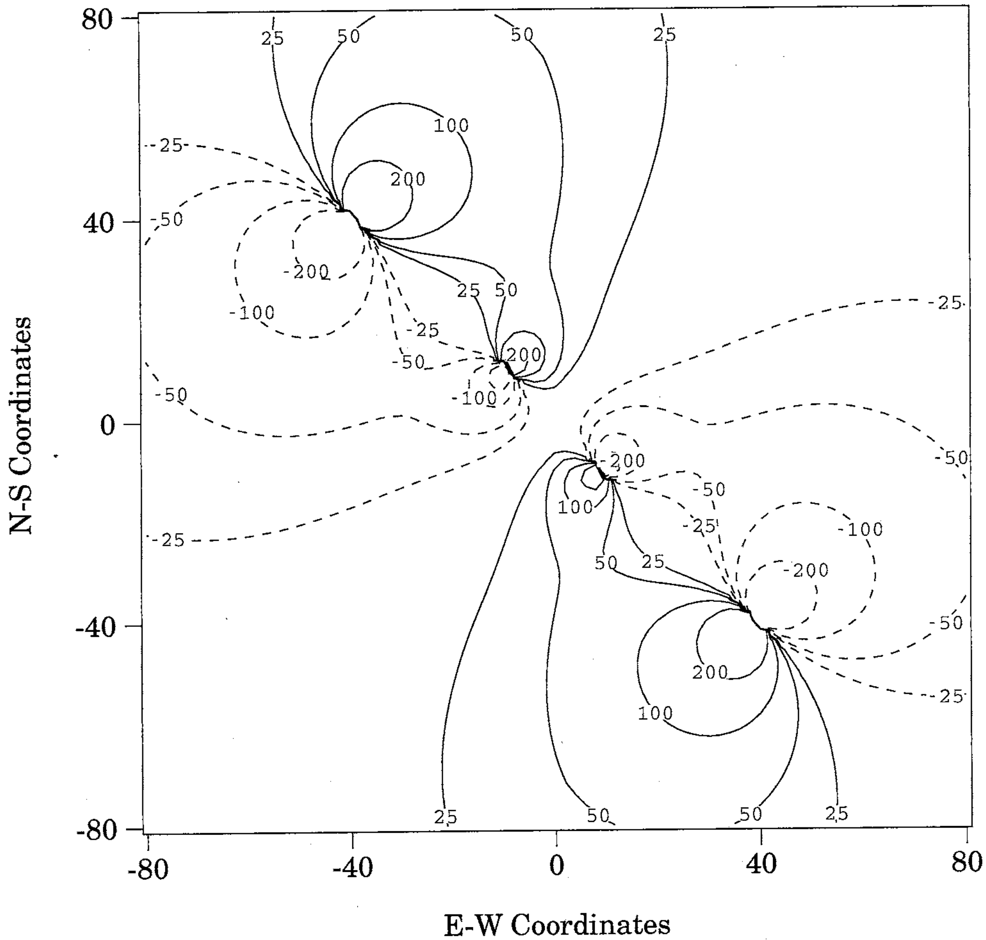
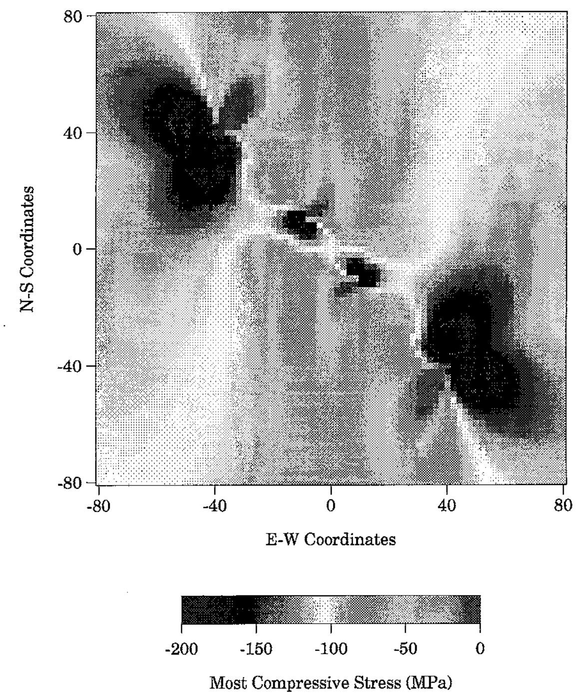


Figure 10c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



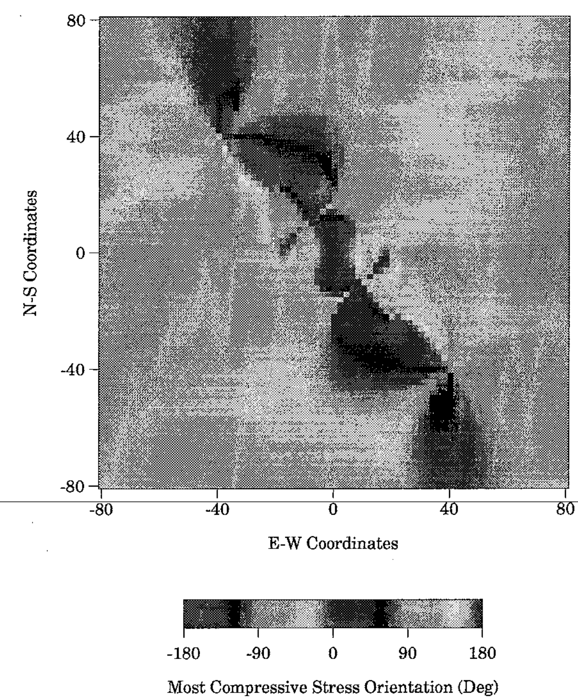
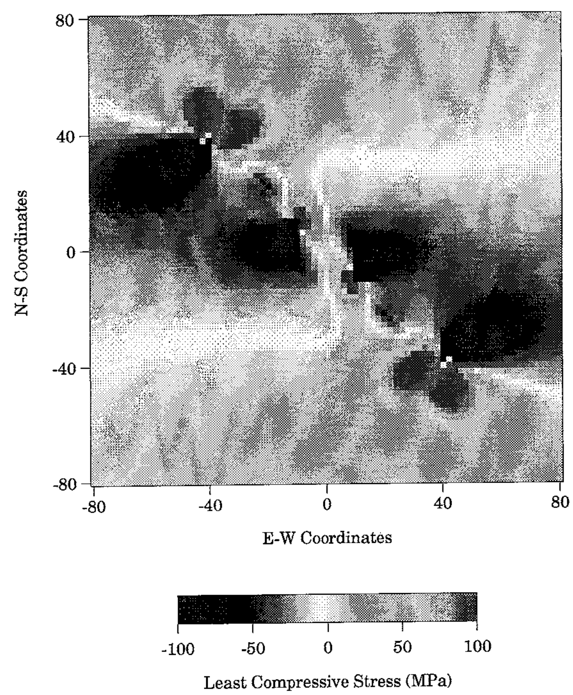


Figure 10d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



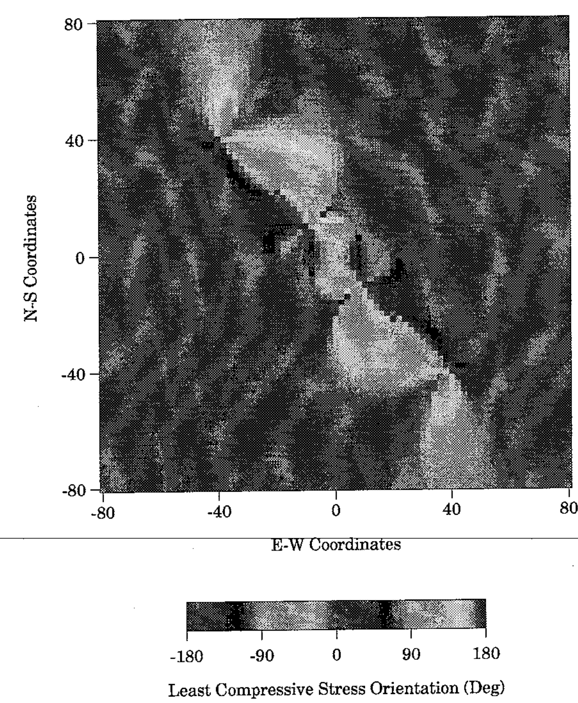


Figure 10e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

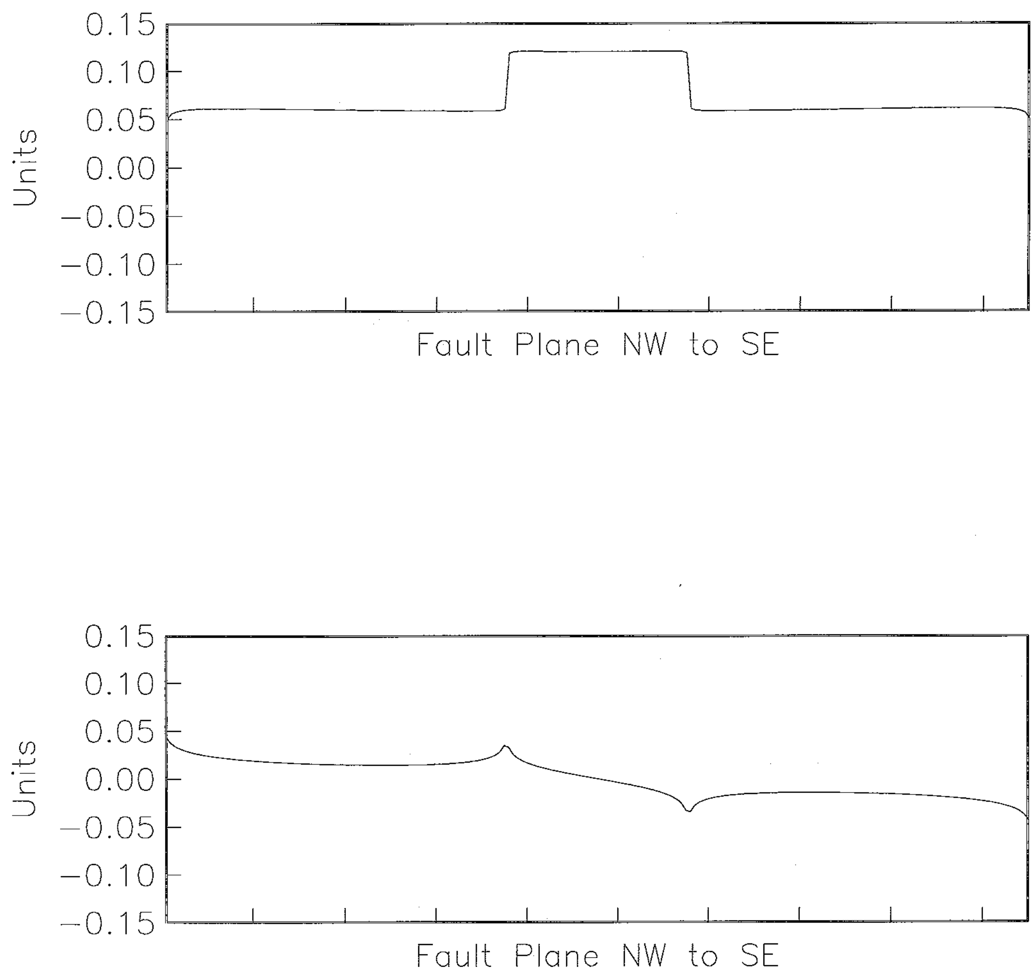


Figure 11a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The weak gap has elevated shear displacement and has small cusps of twisting near the strength transition.

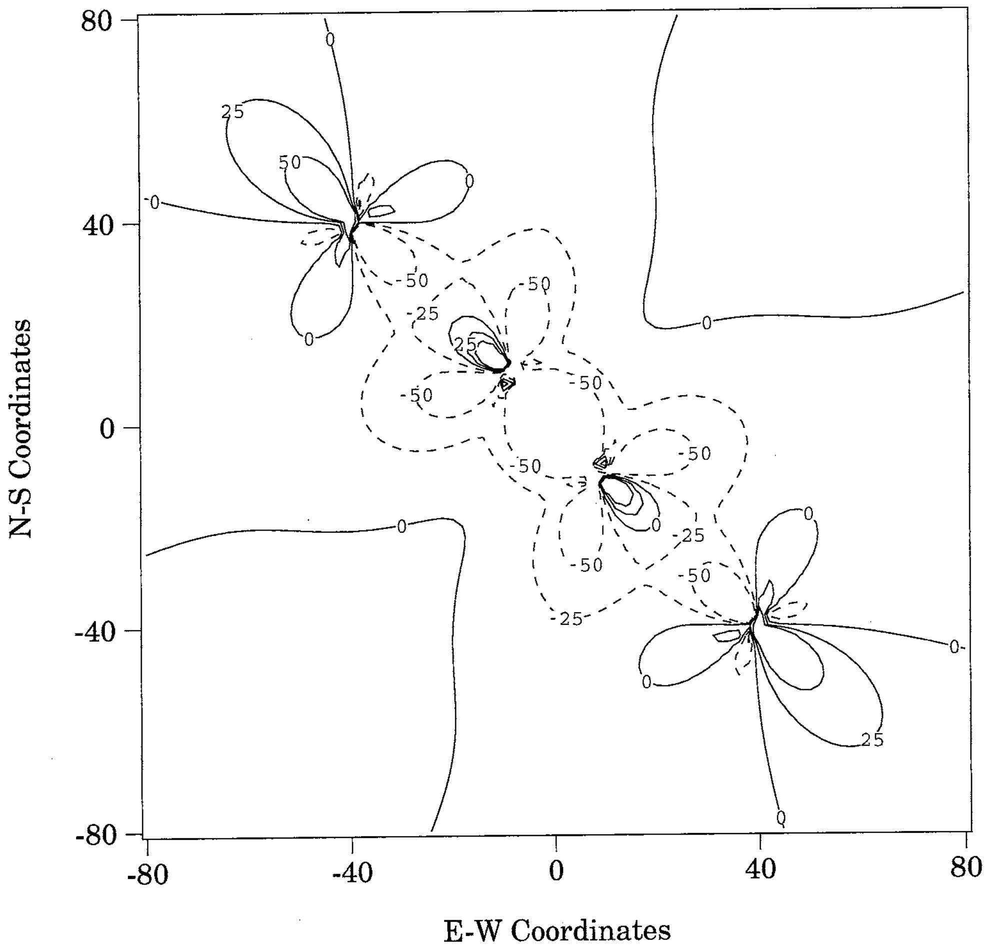


Figure 11b: The shear traction change parallel to a weak gap (Figure 8(B)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

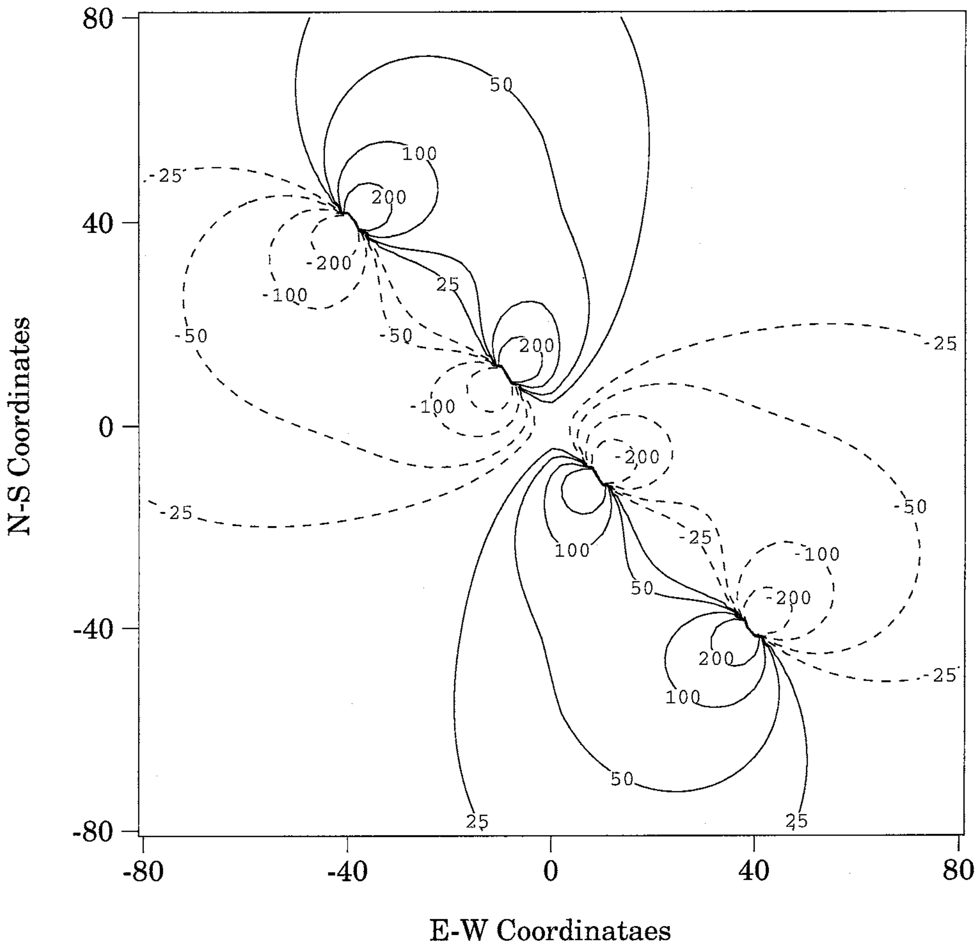


Figure 11c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

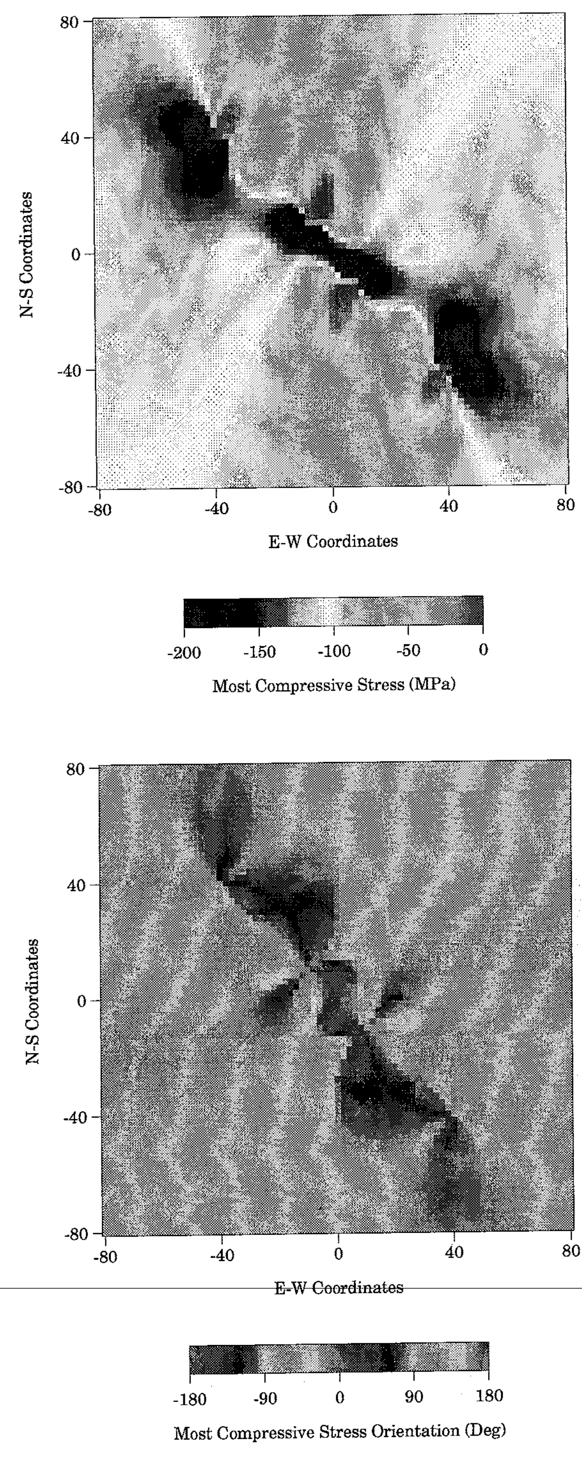


Figure 11d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

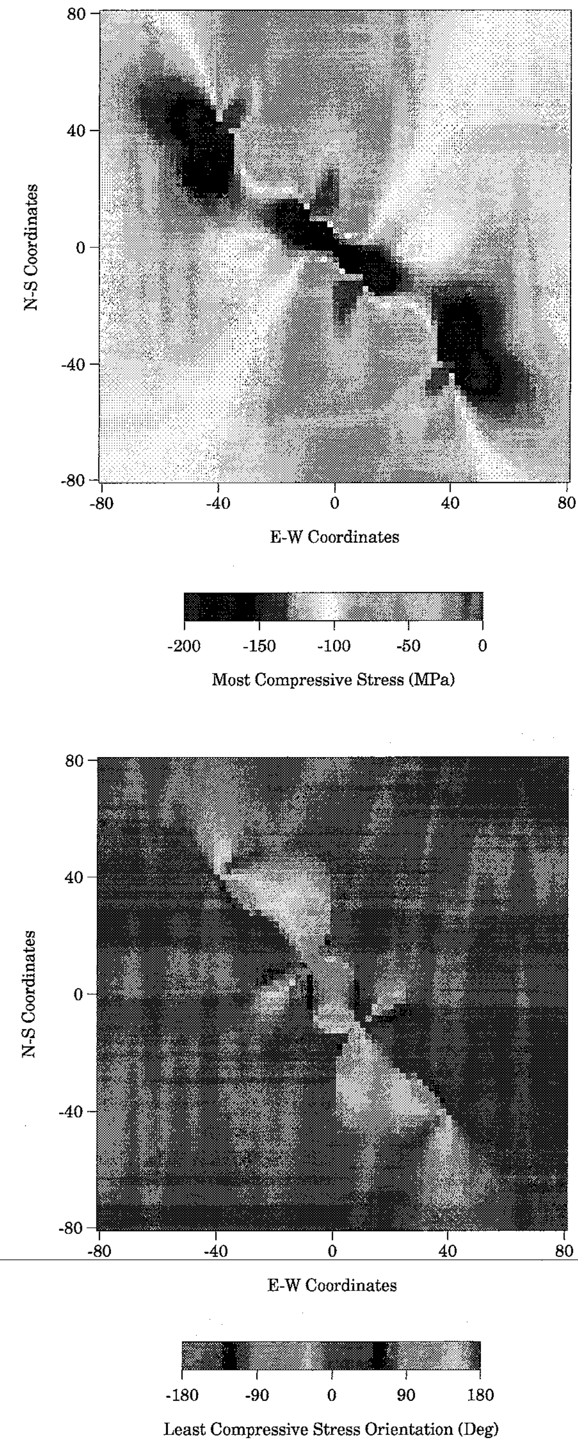
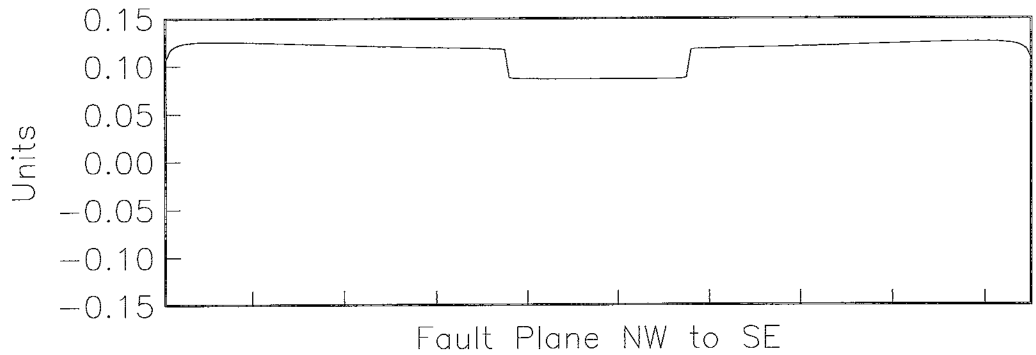


Figure 11e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 8(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



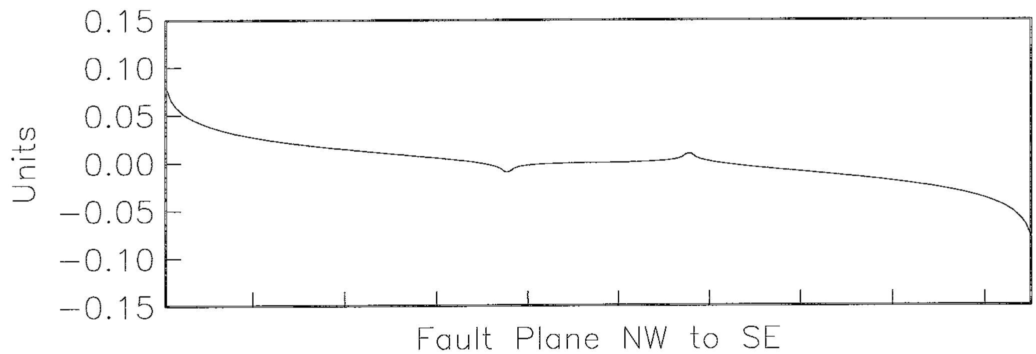


Figure 12a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The strong gap has lowered shear displacement and has small cusps of twisting near the strength transition.

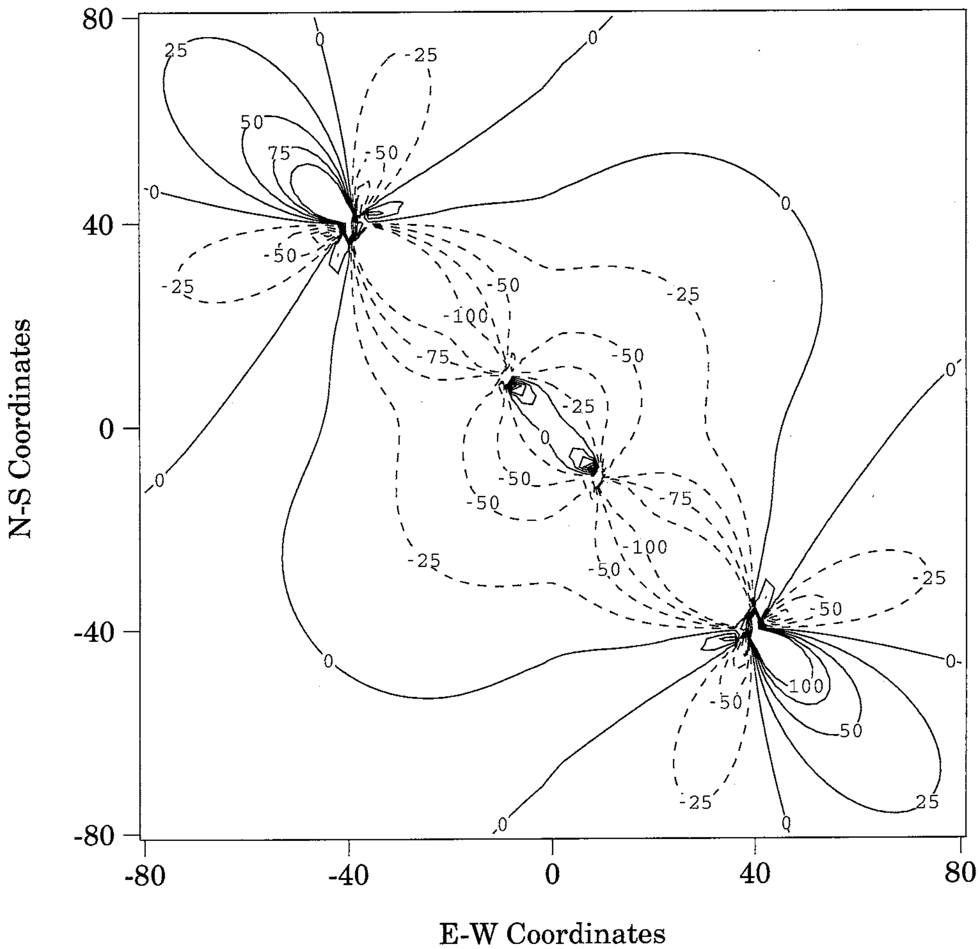


Figure 12b: The shear traction change parallel to a strong gap (Figure 8(C)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

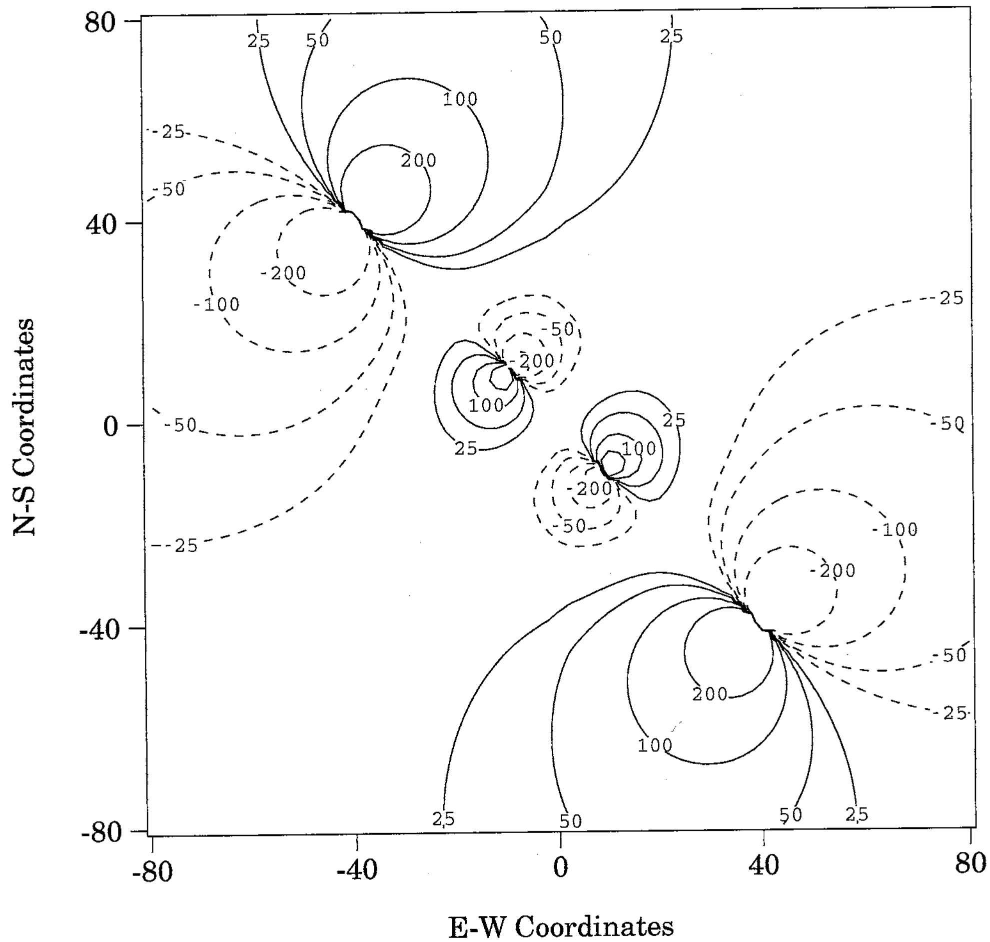


Figure 12c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

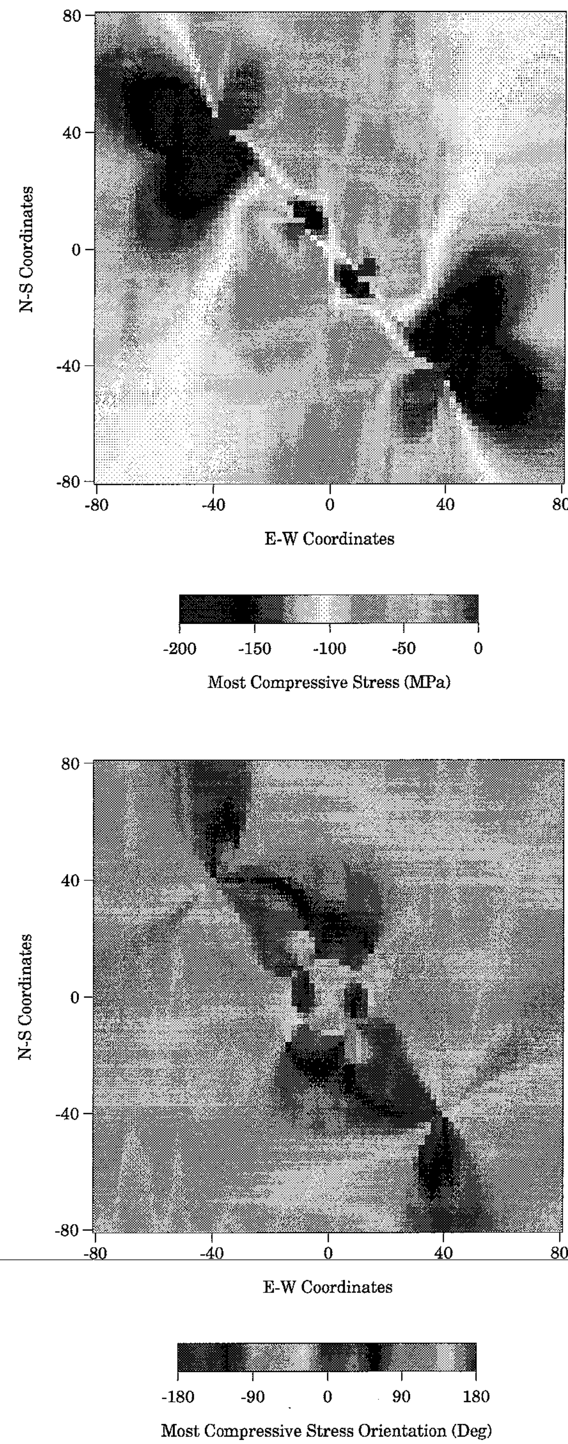


Figure 12d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

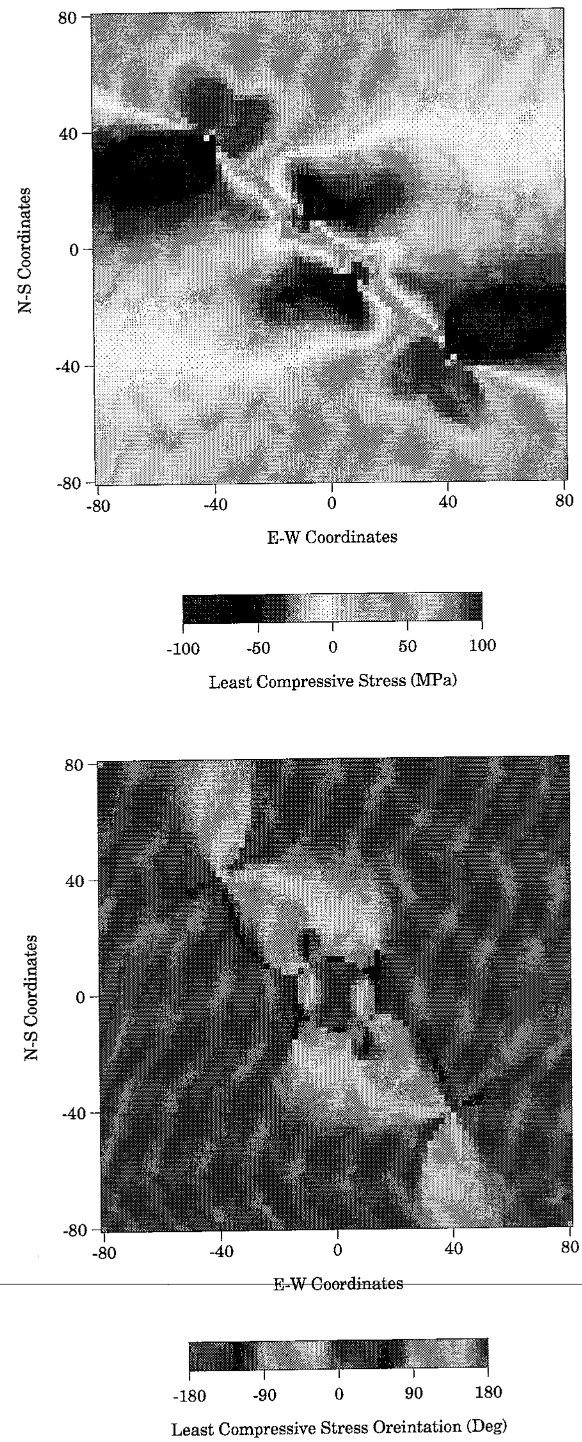


Figure 12e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

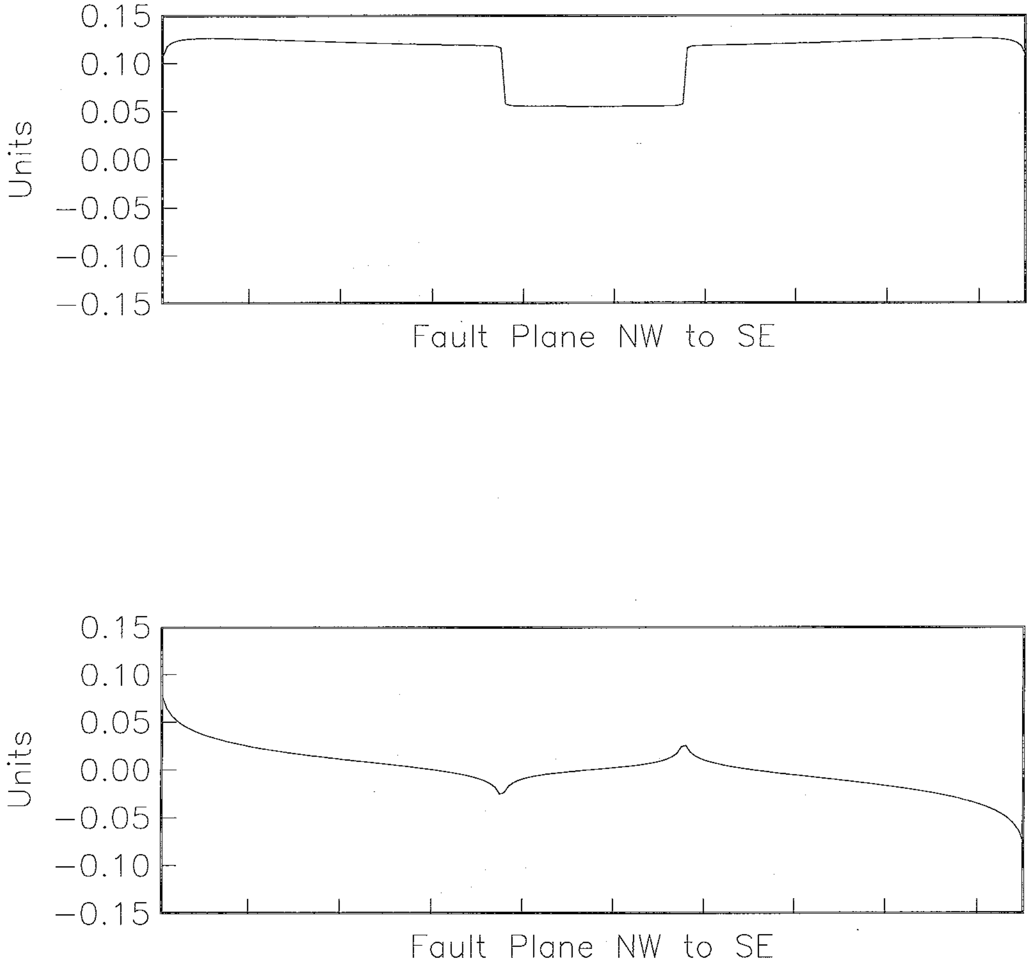


Figure 13a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The strong gap has lowered shear displacement and has small cusps of twisting near the strength transition.

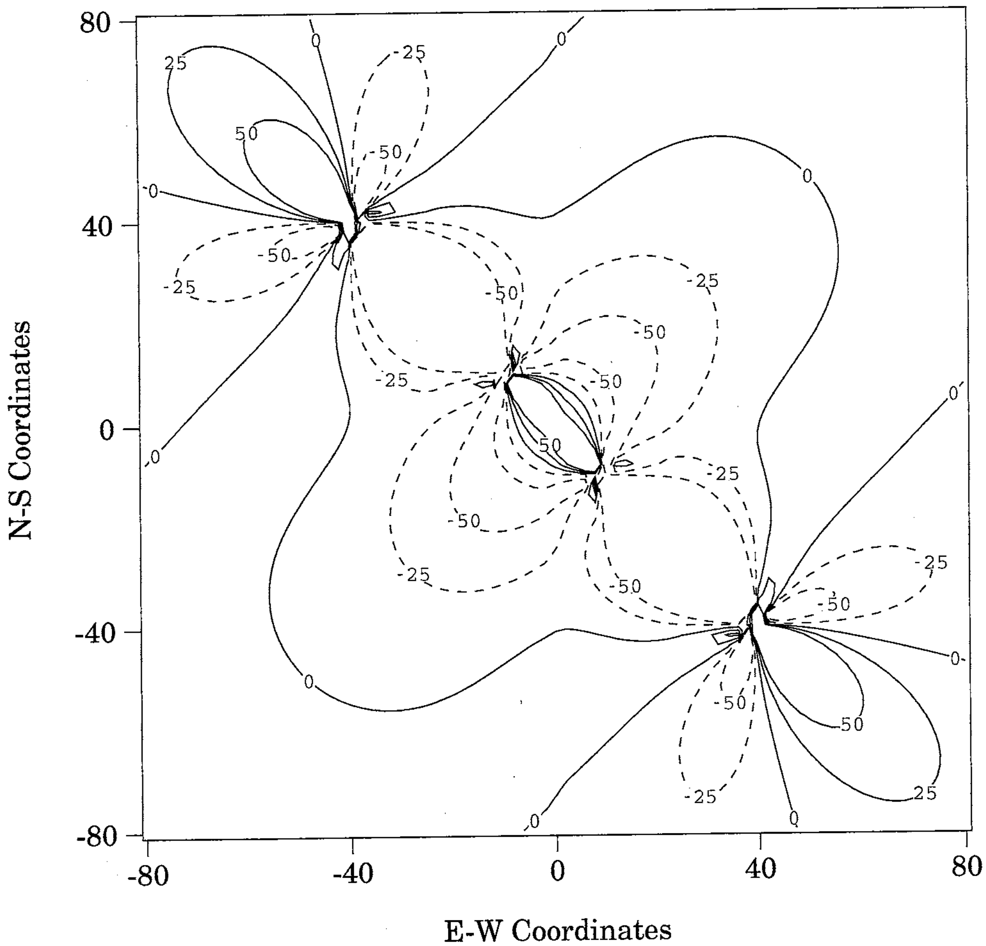


Figure 13b: The shear traction change parallel to a strong gap (Figure 8(D)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

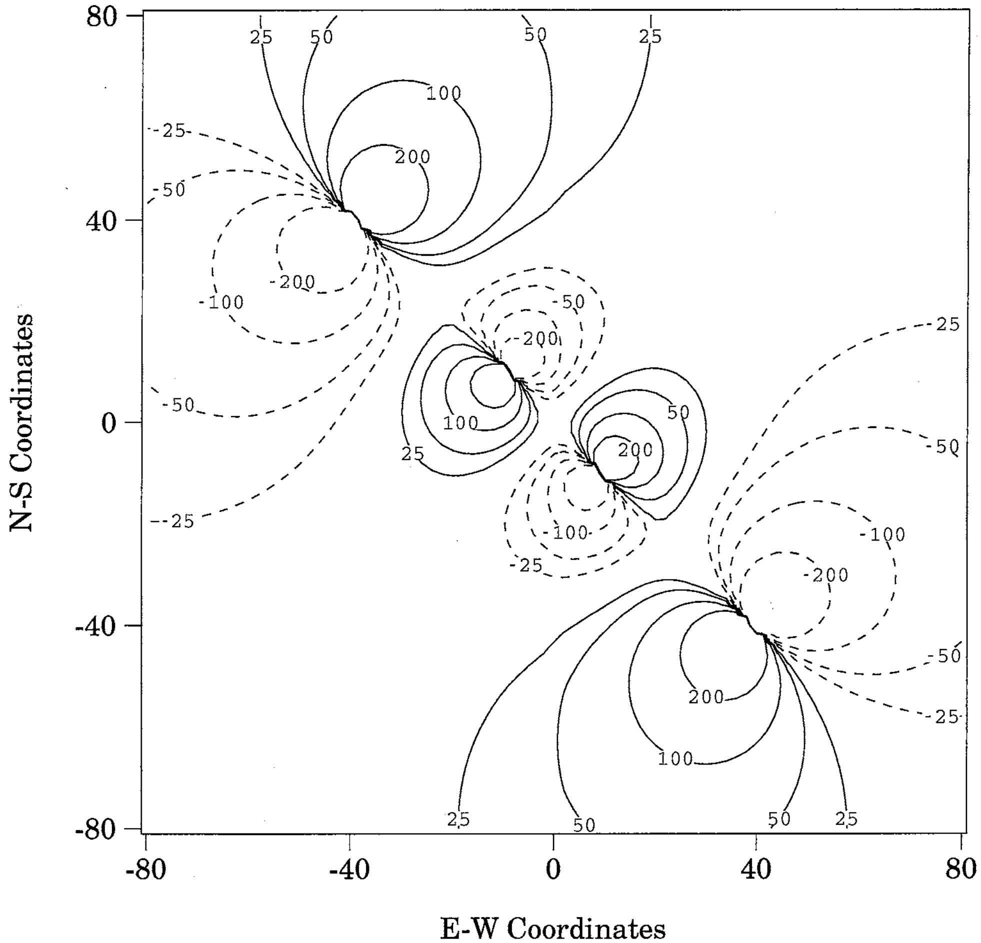


Figure 13c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

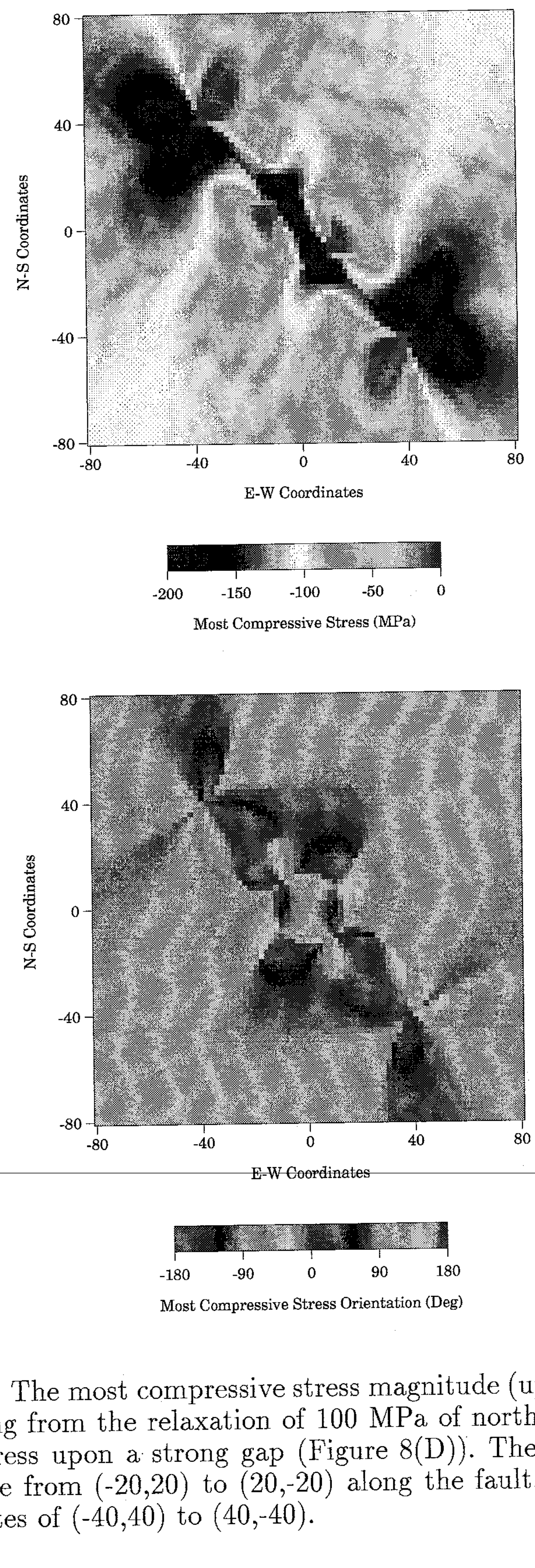


Figure 13d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

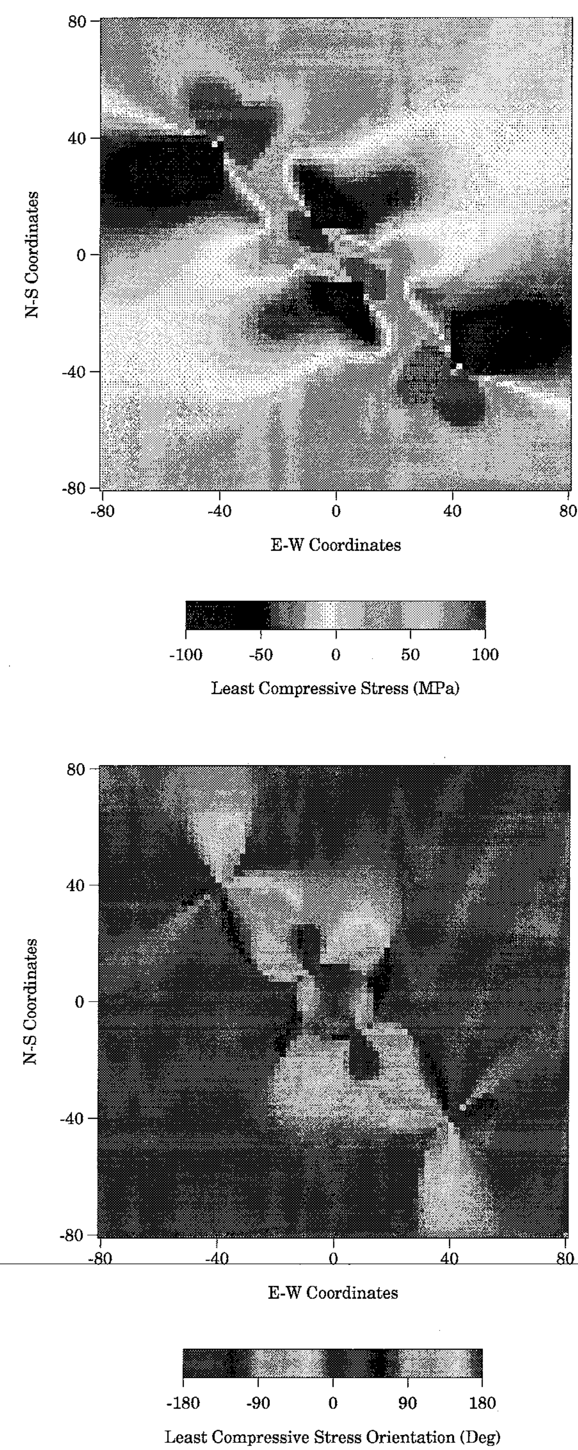
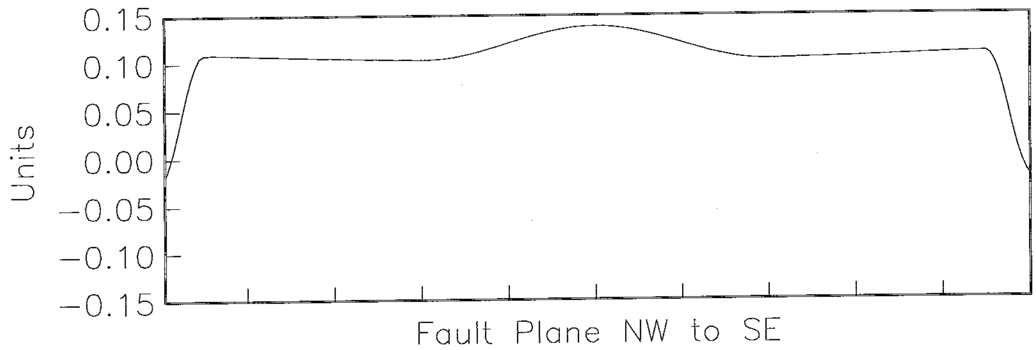


Figure 13e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 8(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



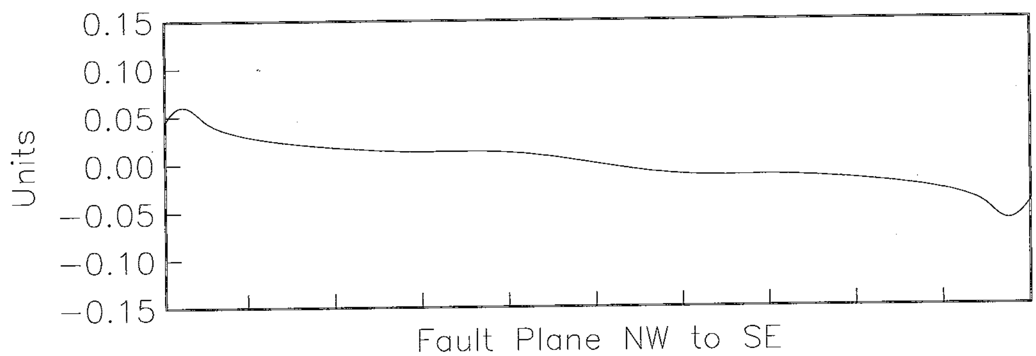


Figure 14a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The weak gap has elevated shear displacement and has small lobes of twisting near the strength transition.

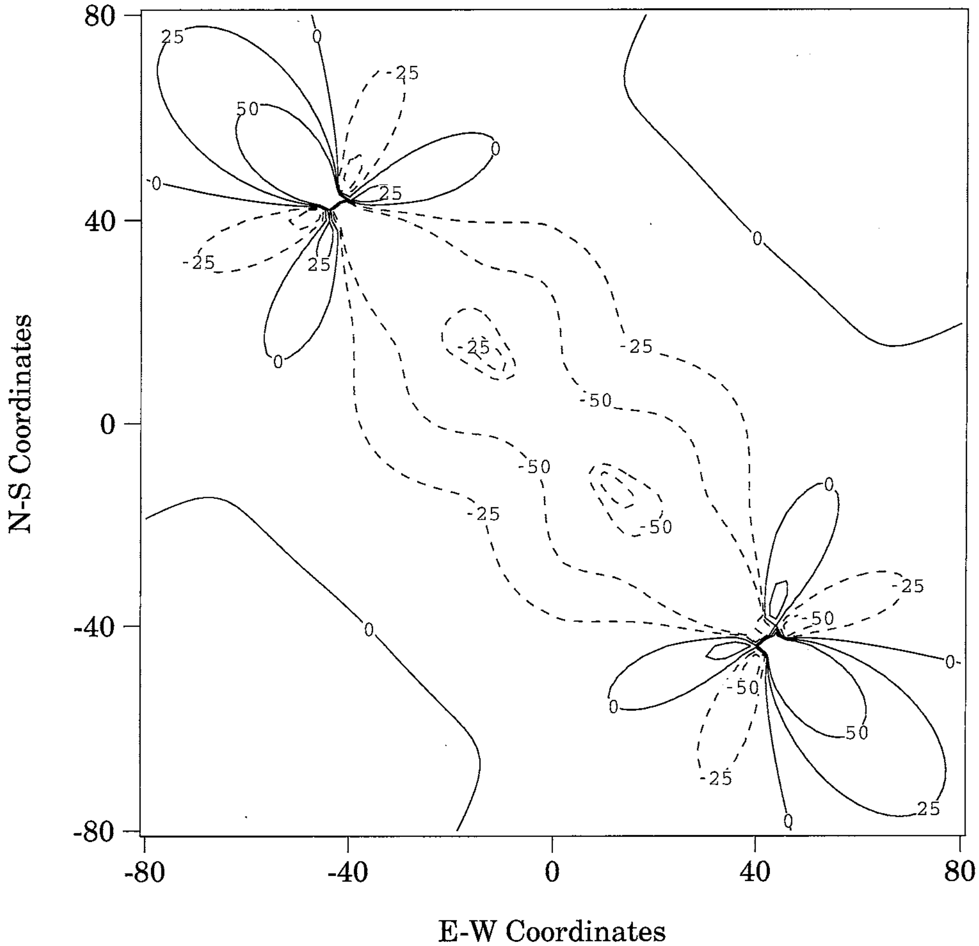


Figure 14b: The shear traction change parallel to a weak gap (Figure 9(A)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

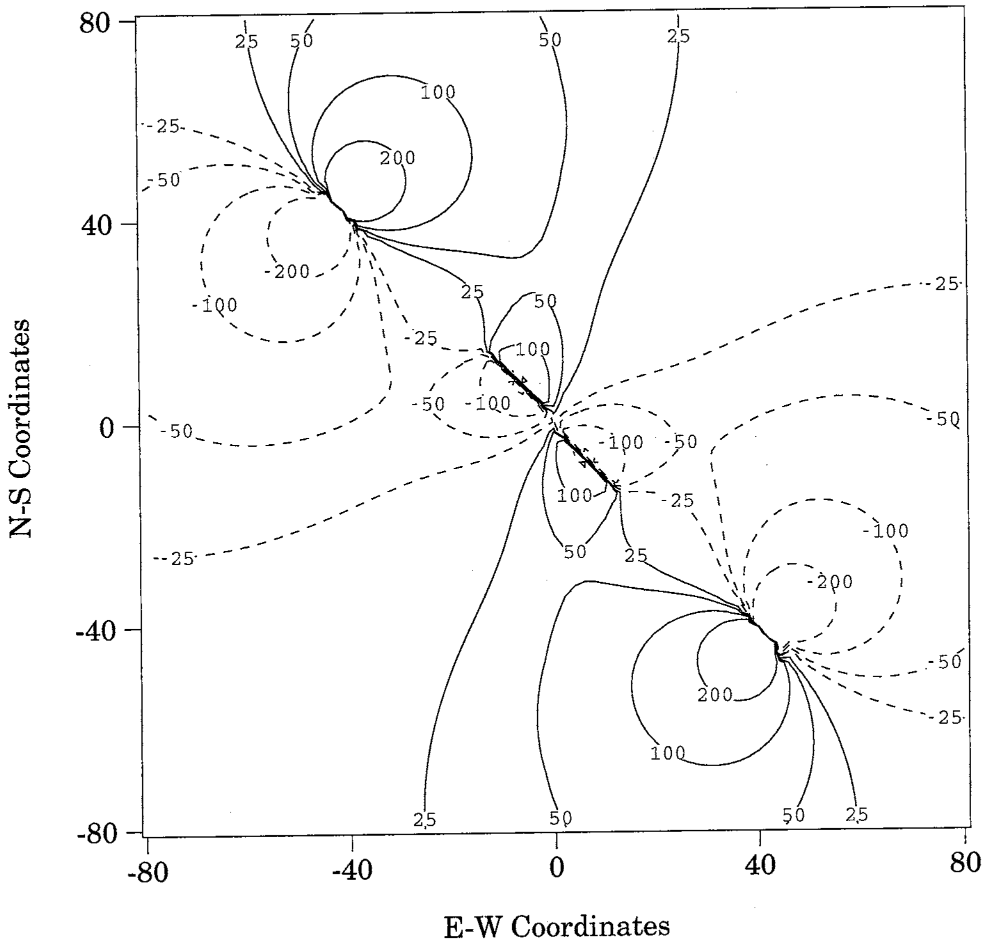


Figure 14c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

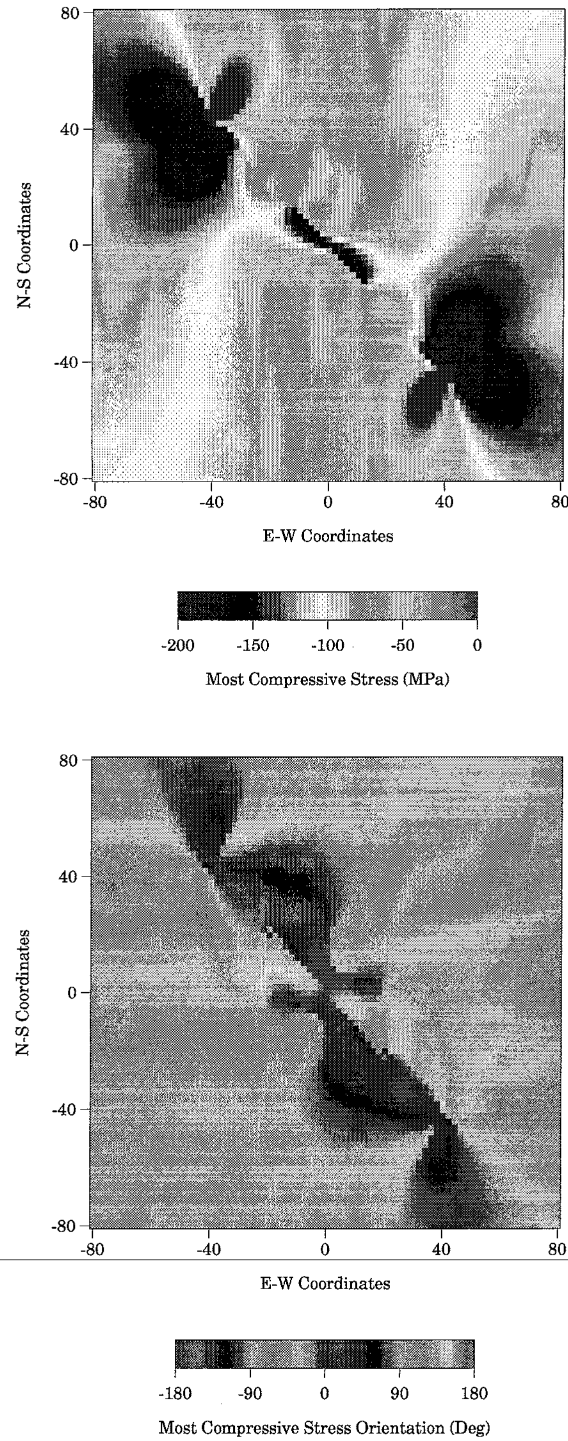


Figure 14d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

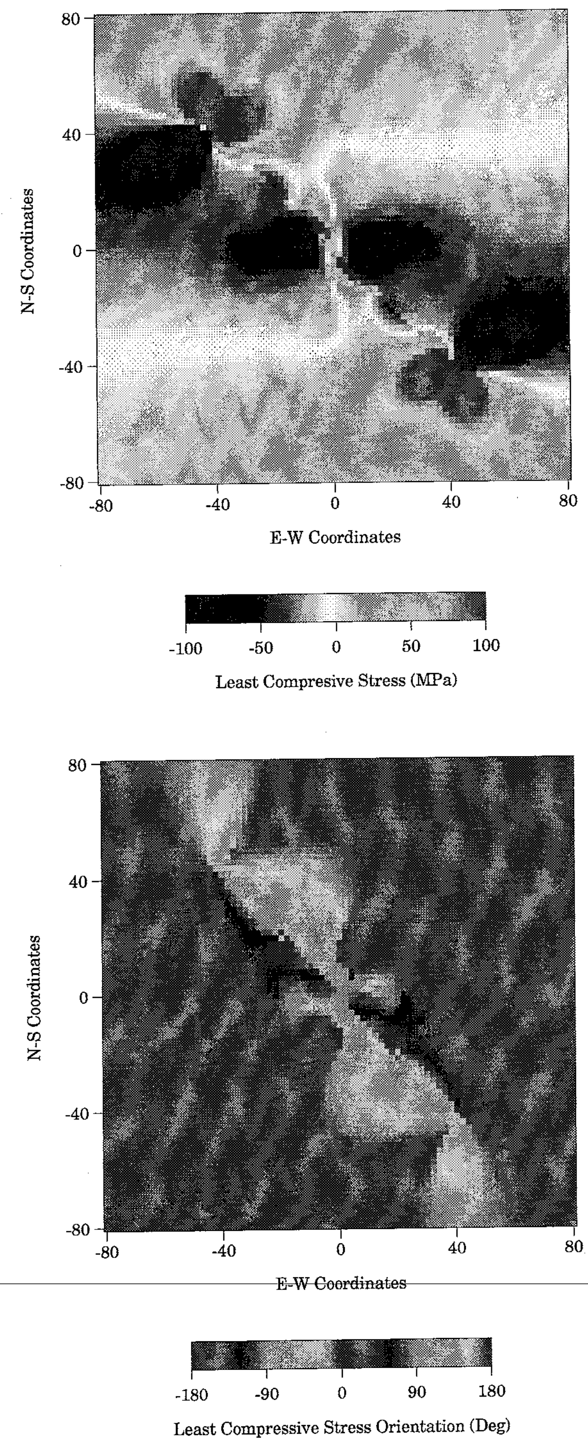


Figure 14e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(A)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

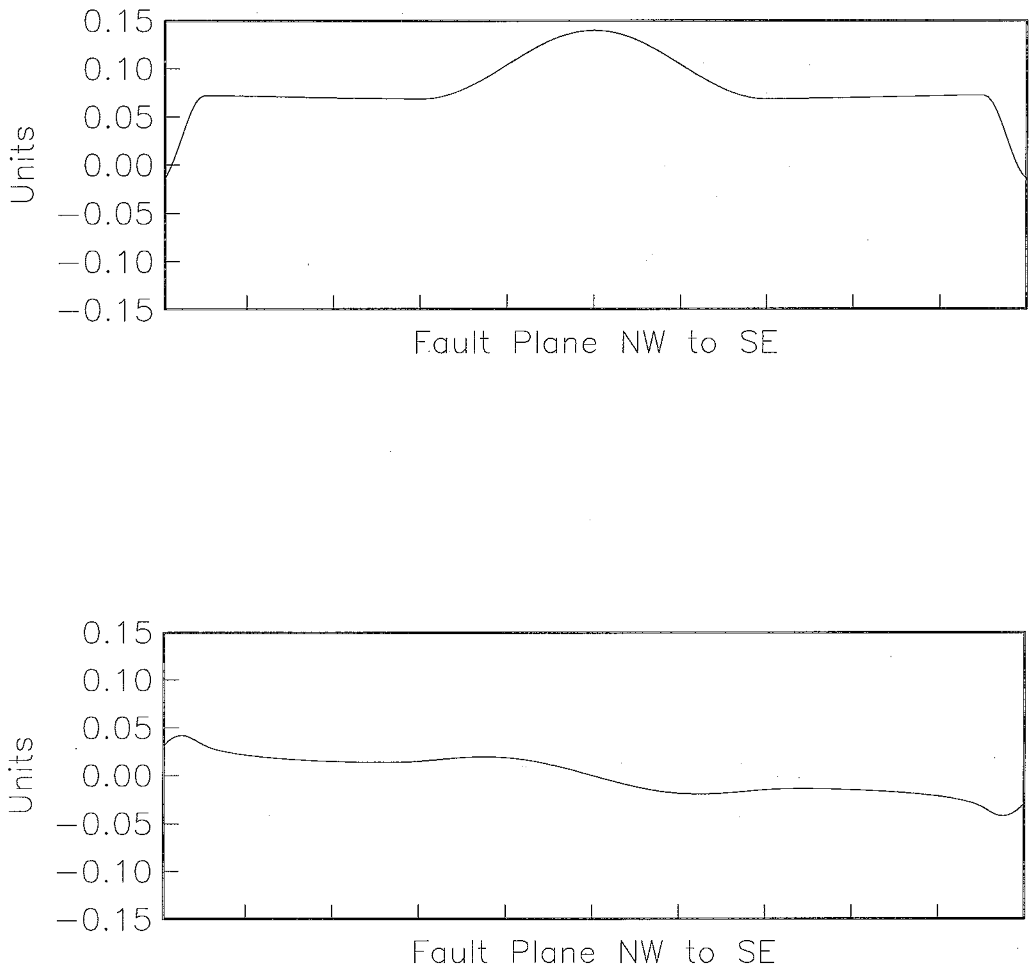


Figure 15a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The weak gap has elevated shear displacement and has small lobes of twisting near the strength transition.

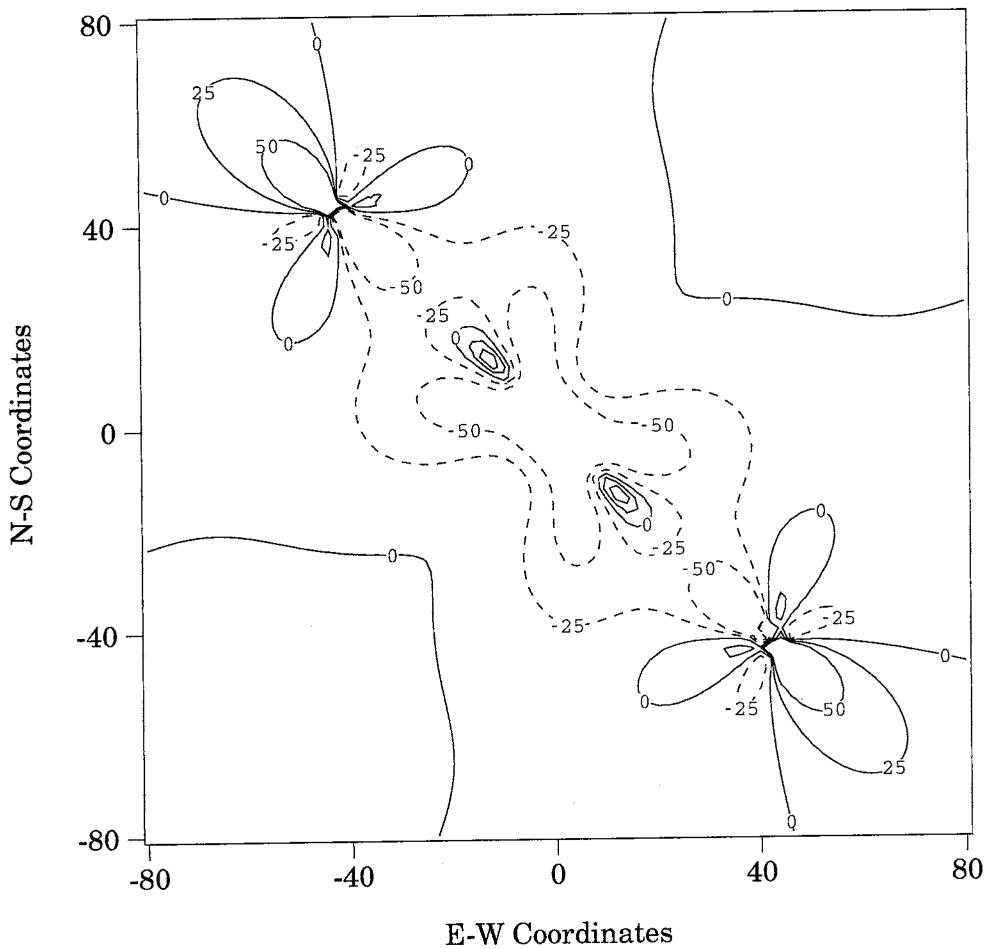


Figure 15b: The shear traction change parallel to a weak gap (Figure 9(B)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

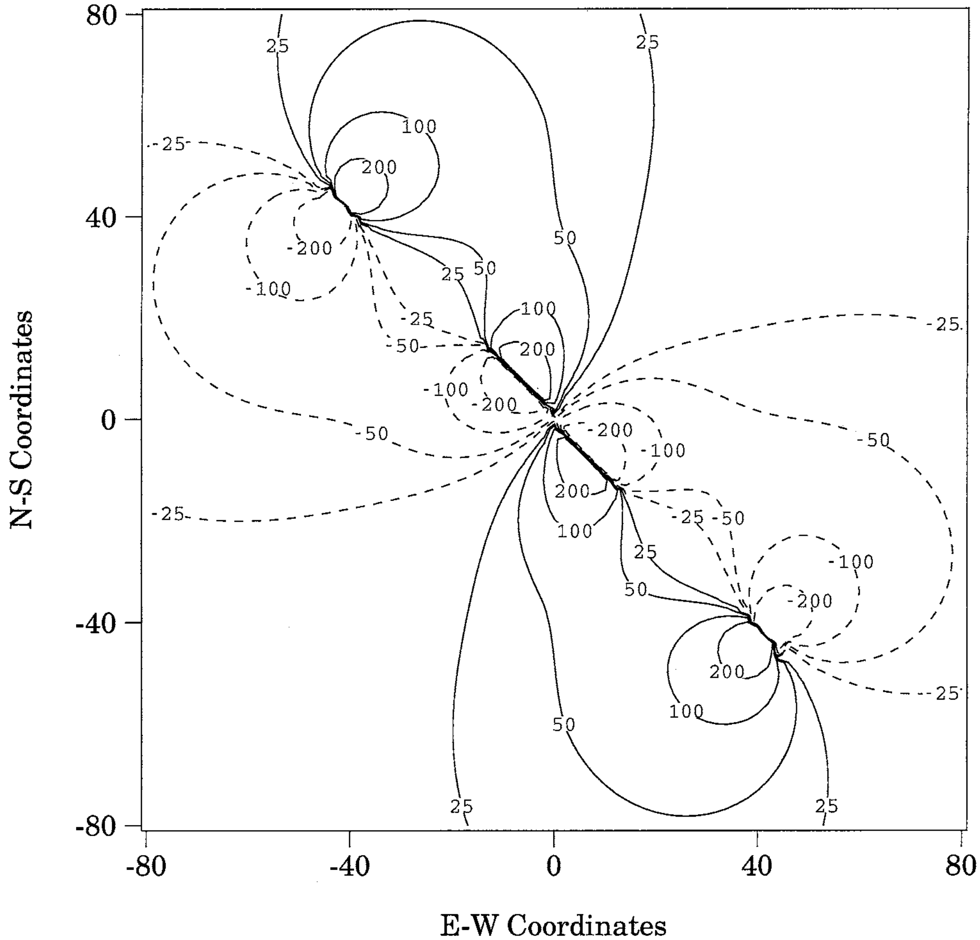
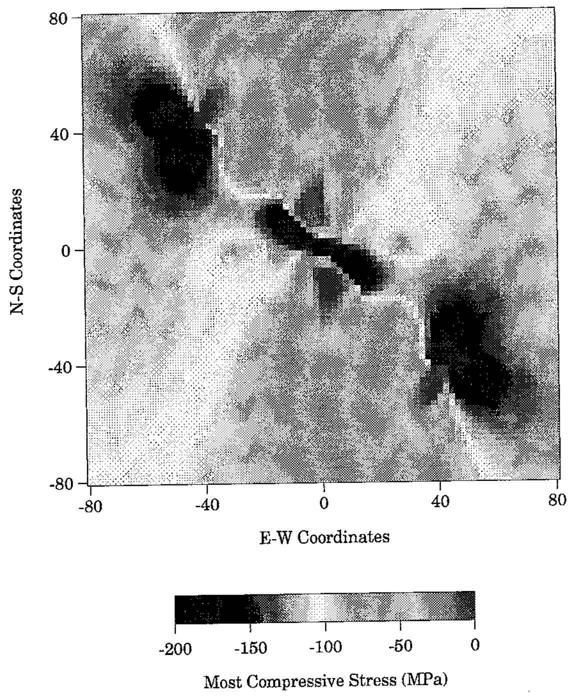


Figure 15c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



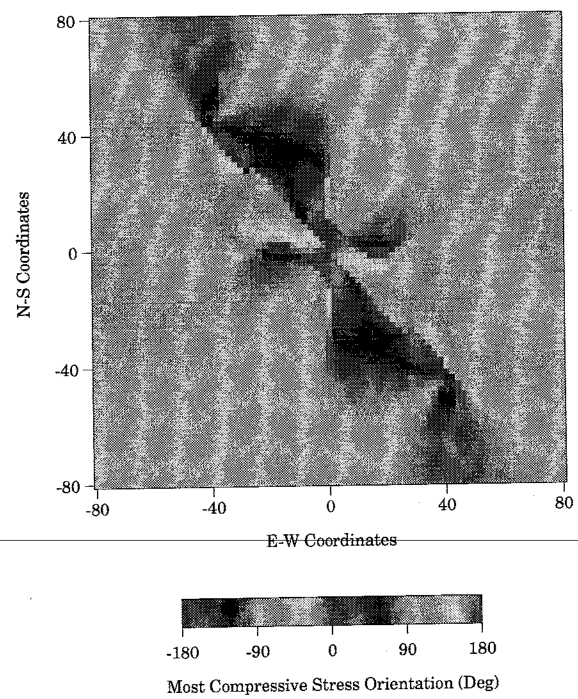


Figure 15d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

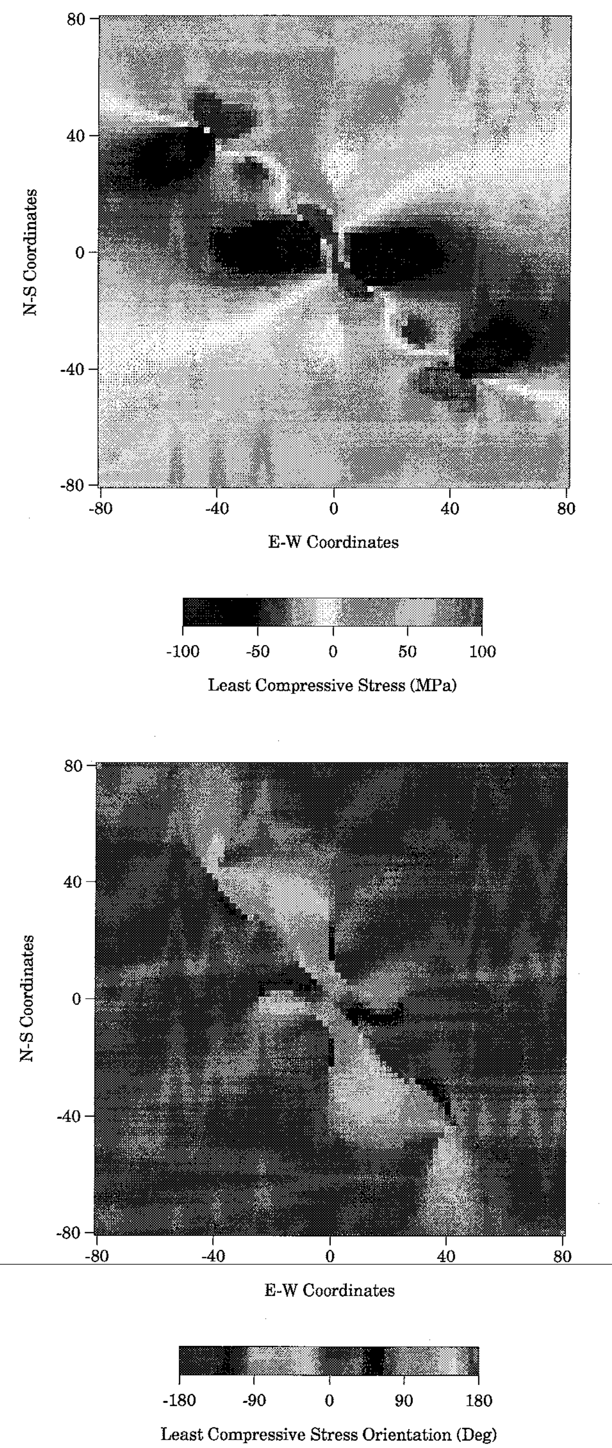
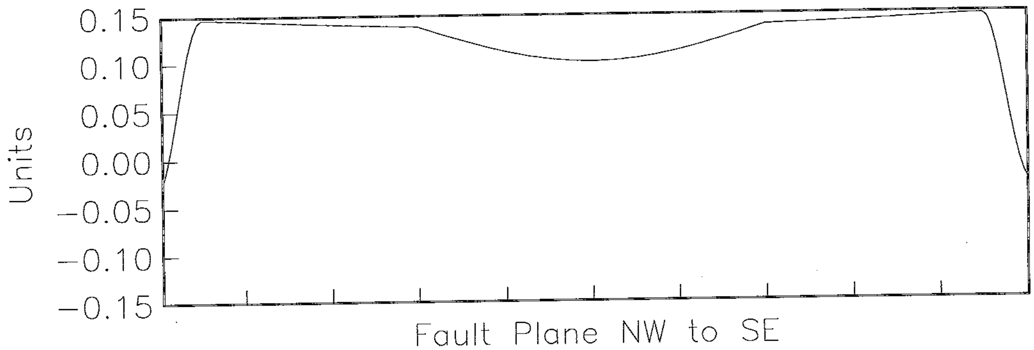


Figure 15e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a weak gap (Figure 9(B)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



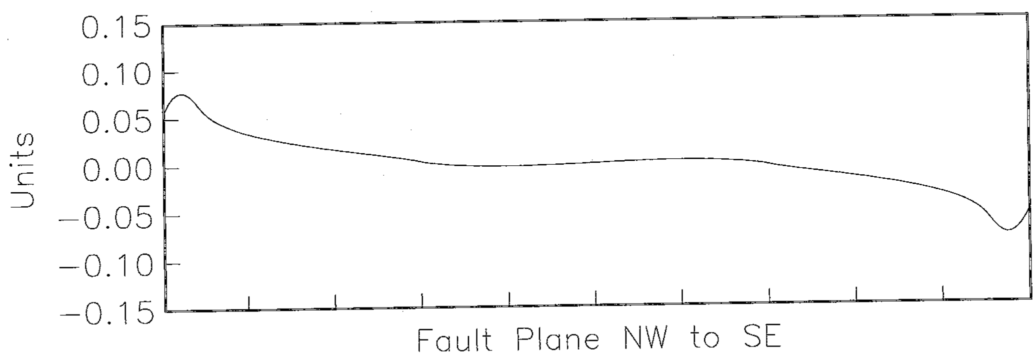


Figure 16a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The strong gap has lowered shear displacement and has small lobes of twisting near the strength transition.

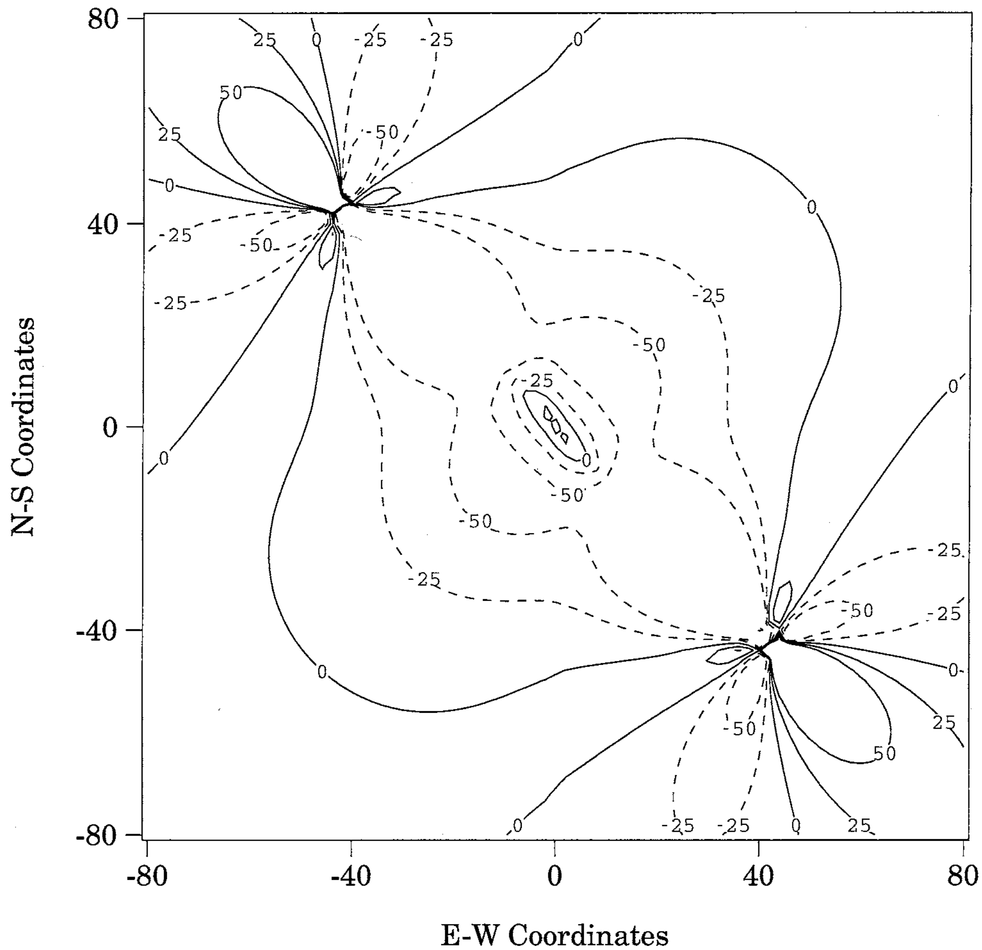


Figure 16b: The shear traction change parallel to a strong gap (Figure 9(C)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

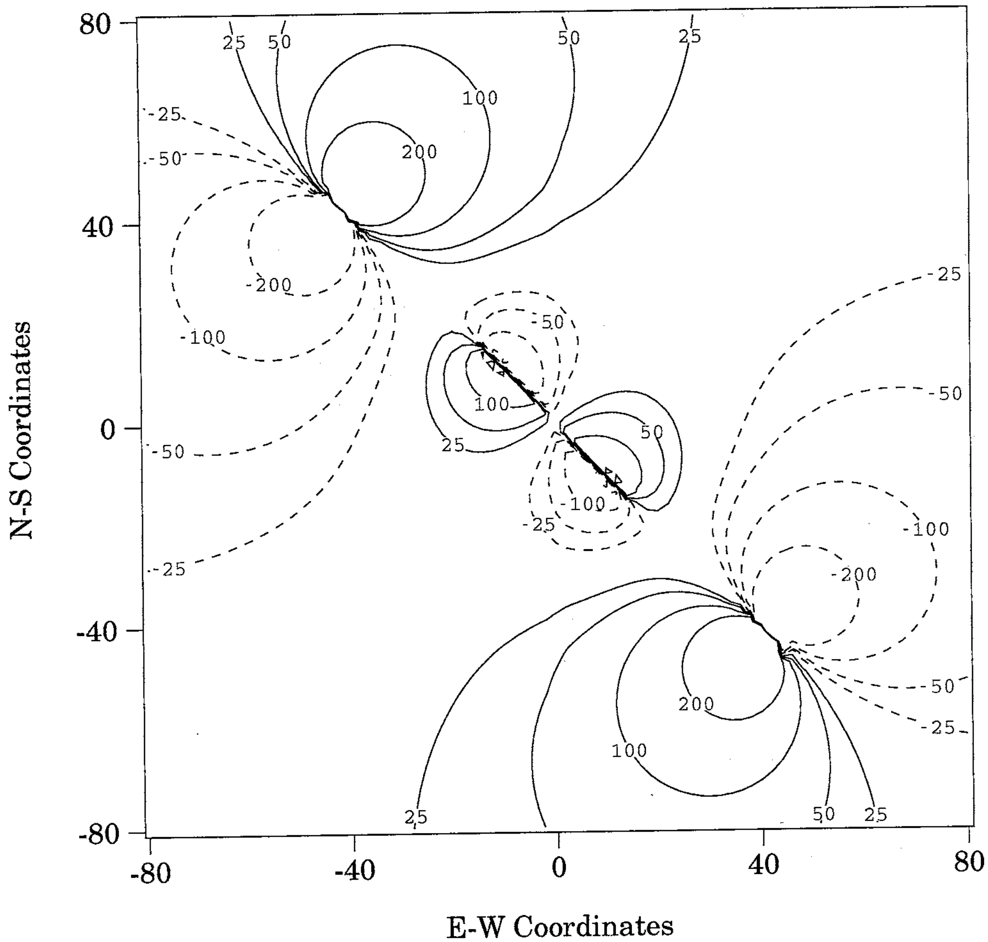


Figure 16c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

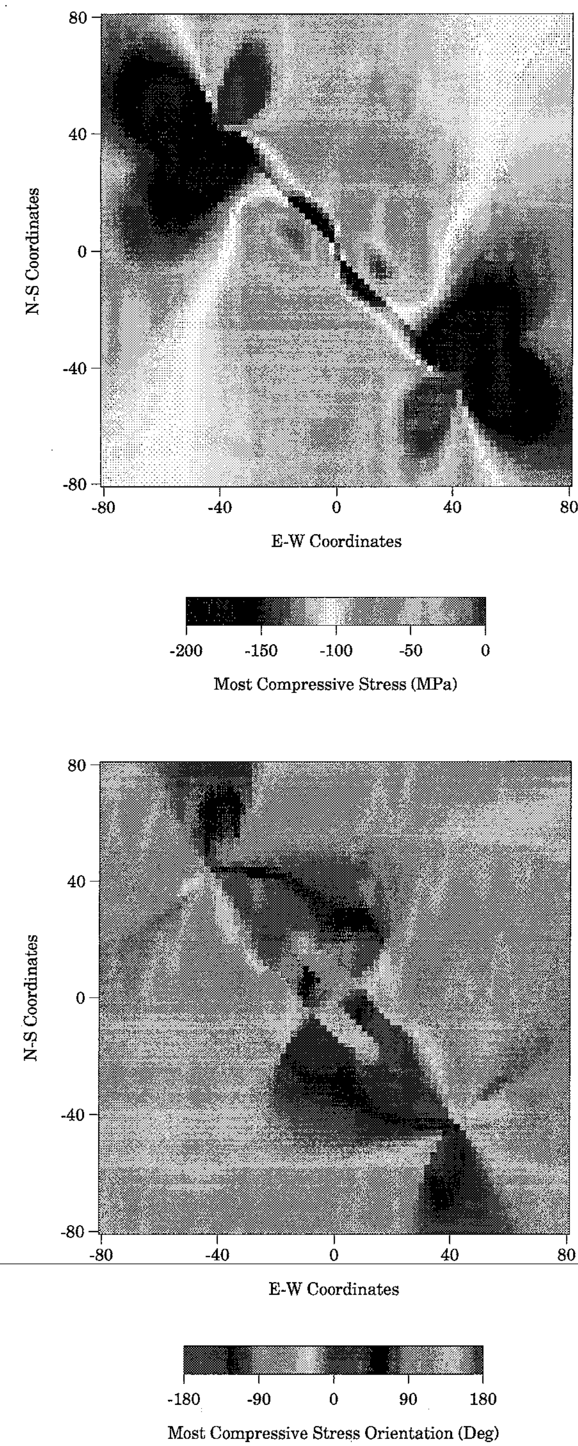


Figure 16d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

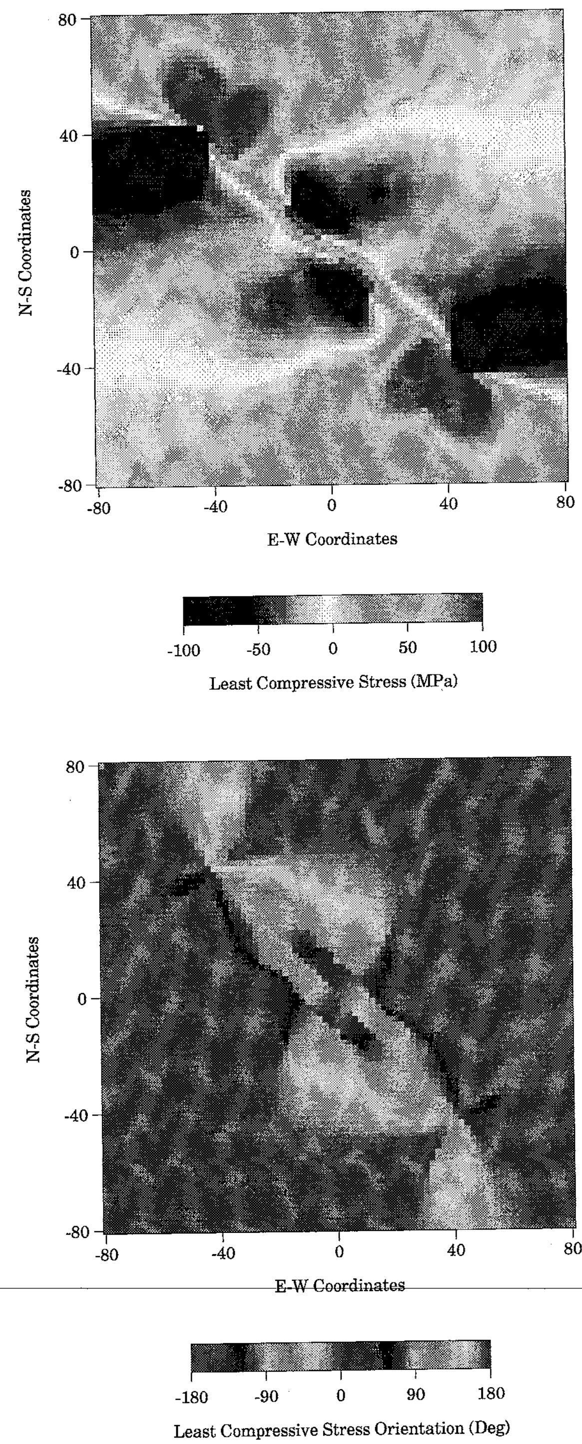


Figure 16e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(C)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

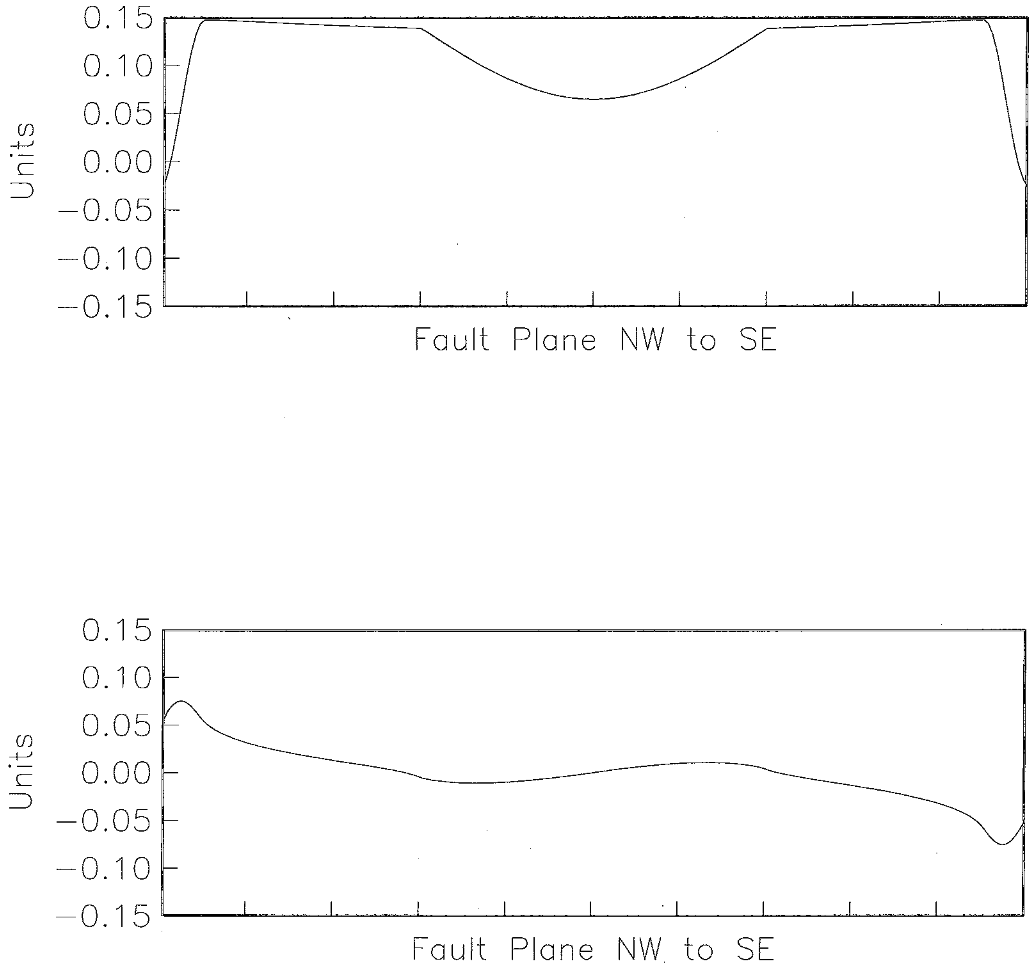


Figure 17a: The scale-independent shear displacement (top) and normal displacement (bottom) in coordinate units arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40). The values for the shear displacement are for one side of the fault and therefore are exactly half of the relative movement. The strong gap has lowered shear displacement and has small lobes of twisting near the strength transition.

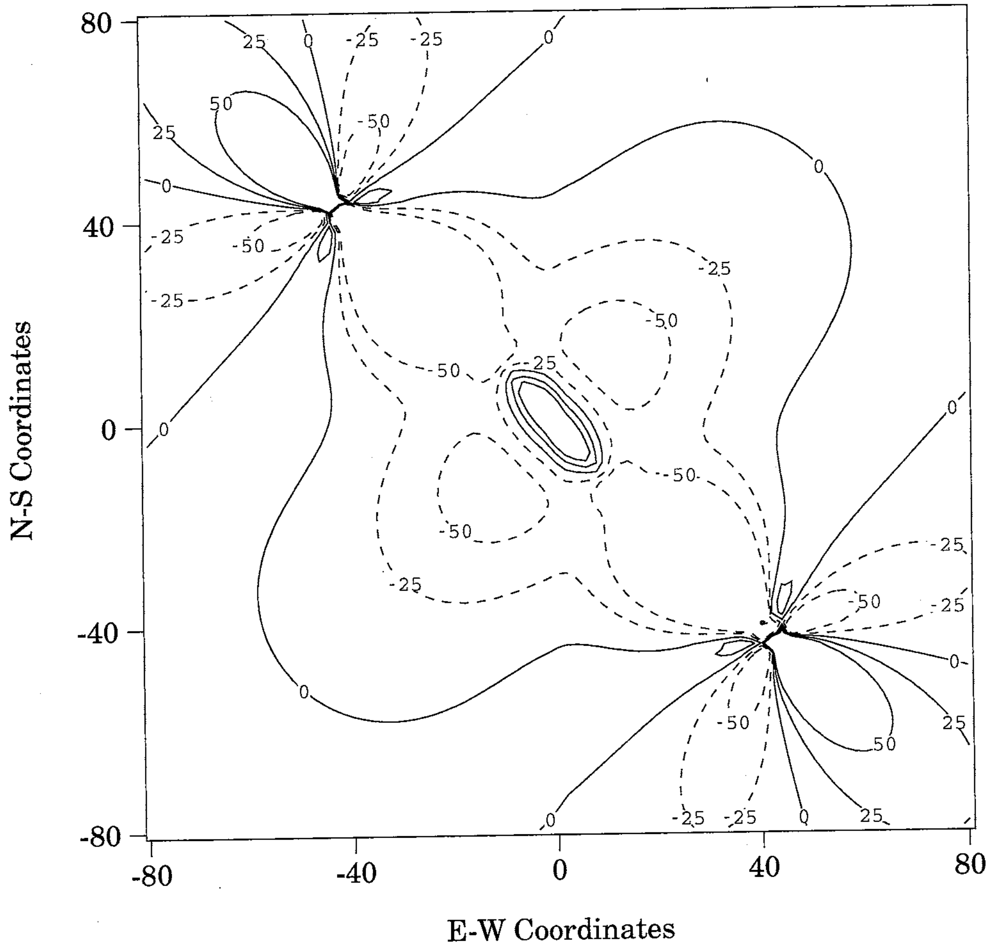


Figure 17b: The shear traction change parallel to a strong gap (Figure 9(D)). Areas with negative values represent stress dissipation whereas positive values represent stress accumulation. These values arise from the the relaxation of 100 MPa of north-south oriented compressional stress. The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

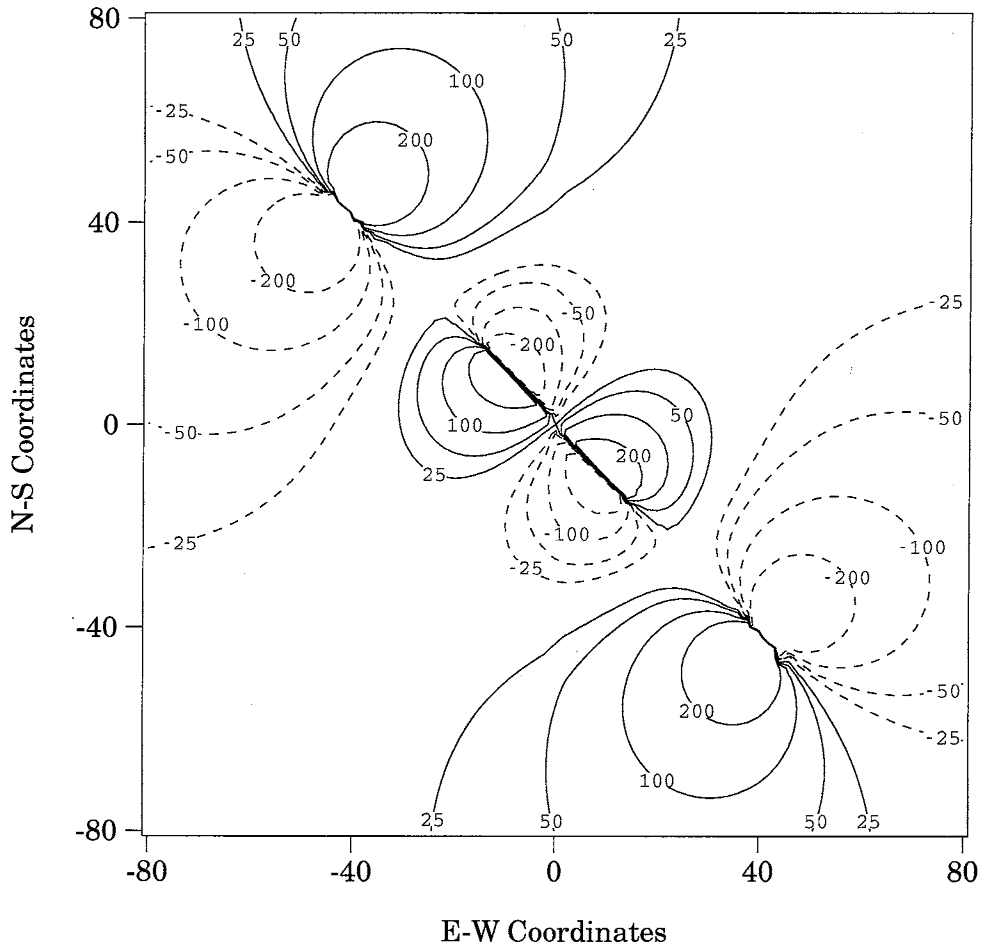


Figure 17c: The volumetric stress arising from the the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

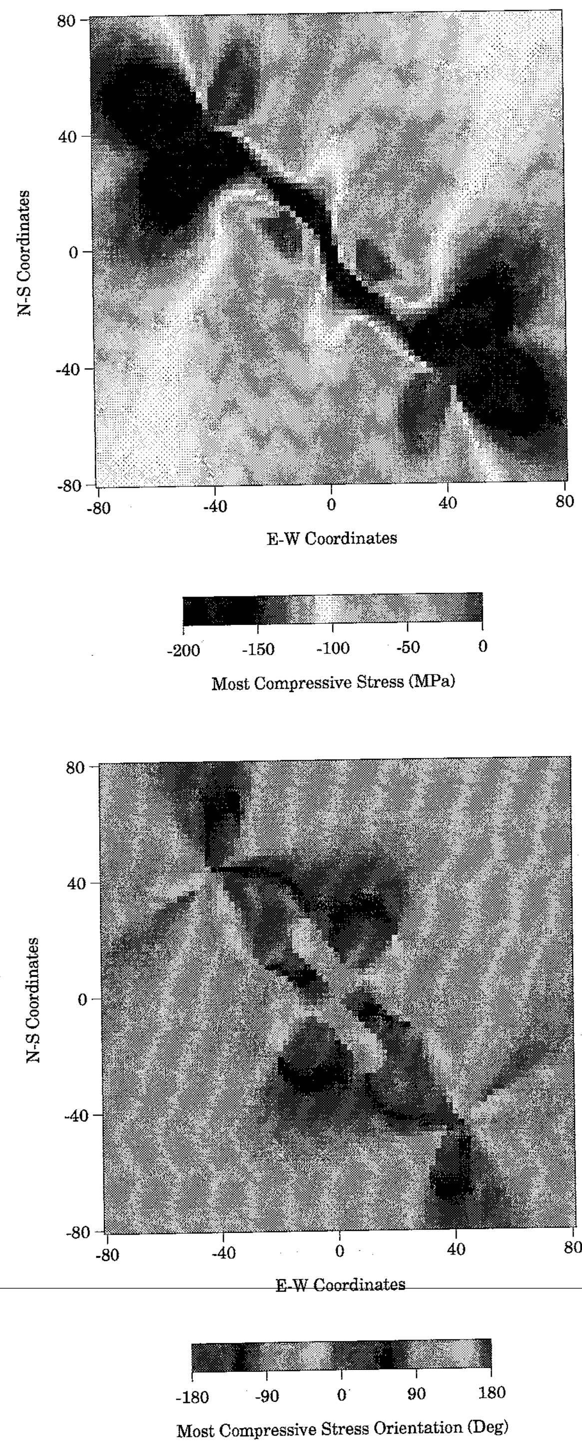
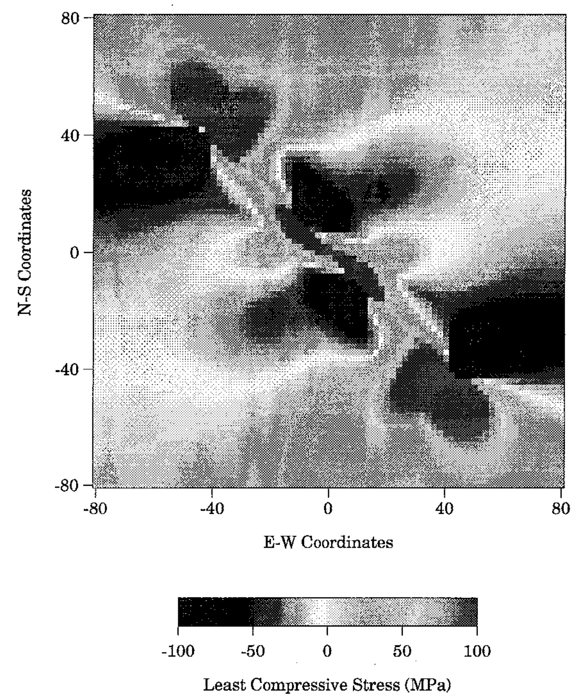


Figure 17d: The most compressive stress magnitude (upper) and orientation (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).



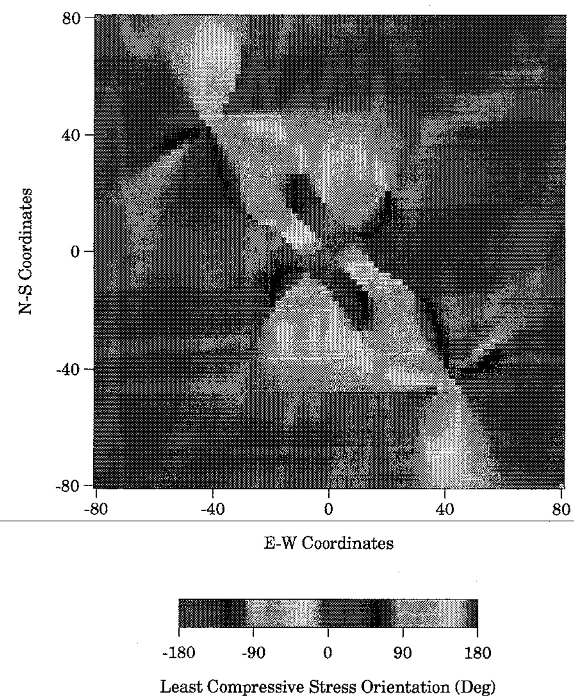


Figure 17e: The least compressive stress magnitude in MPa (upper) and orientation in degrees counter-clockwise with respect to east (lower) arising from the relaxation of 100 MPa of north-south oriented compressional stress upon a strong gap (Figure 9(D)). The X-Y coordinates for the gap range from (-20,20) to (20,-20) along the fault, which extends from the coordinates of (-40,40) to (40,-40).

Appendix A: Program Listings

- main.c
- adjbv.c
- bcalc.c
- bcond.c
- coeff.c
- elmtplt.c
- field_plt.c
- field_primer
- help.c
- influence.c
- instructions.c
- lubksb.c
- ludcmp.c
- nrutil.c
- output.c
- prcpl_stress.c
- print_input.c
- recips.c
- solve_lu.c
- somig.c
- specin.c
- specpt.c
- stressrot.c
- uplot.c

Appendix A References

Crouch, S.L. and A.M. Starfield, *Boundary Element Methods in Solid Mechanics,* Cambridge University Press, 1990.

Press, William H., Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery, *Numberical Recipes in C: The Art of Scientific Computing,* Cambridge University Press, 1992.

```
/*
  SMUDGE -    Stress
              Modeling
              Utilizing a
              DBI-algorithm* in
              Geophysical
              Elastostatics

              * Algorithm developed by Croach & Starfield
*/
#include <malloc.h>
#include <math.h>
#include <strings.h>
#include <stdio.h>
#include "smudge.h"
#define Pi 3.14159265359
#include "nrutil.h"

char be_file[30],oe_file[30],*con_file,destdir[80],command[90];
float g,poisson,young_mod,pr1,pr2,pr3,con;

/* global variables */
float *sinbet,*cosbet,*xm,*ym,*a,*r,*angle,
  **C,*P,*A,*B,*U,*S,*X,*Y,*xbeg,*ybeg,*xend,*yend;
FILE *logfile;

int main (argc,argv)
int argc;
char *argv[];
{
  int numbe=0,n,npoint,*numpb,numbs,numos,memsum,pagesize,noignore=1;
  char* sdate();
  struct griddata grid;
  struct boundpoints be;
  struct specpoints oe;
  FILE *unixpipe;

 /* create a logfile */
  logfile = fopen("logfile","w");

 /* log the start date */
  fprintf(logfile,"%s",sdate());

 /* default input file */
  con_file = "control";

 /* parse command line arguments */
  for(n=1;n<argc;n++) {
    if(!strncmp(argv[n],"-h",2)) {
```

```
            help();
        } else if(!strncmp(argv[n],"-noignore",4)) {
            noignore = 0;
        } else if(!strncmp(argv[n],"-in",3)) {
            /* if control file is specified on command line, then use it */
            con_file = argv[++n];
        }
    }
    fprintf(stderr,"Using input file '%s'...\n",con_file);

/* allocate memory for values stored as lists */
    P = vector(0,4);
    A = vector(0,5);
    B = vector(0,5);
    U = vector(0,10);
    S = vector(0,15);

/* read in the controlling file */
    instructions(&grid,&numbs,&numos);

/* create the destination directory */
    sprintf(command,"echo 'if (! -d %s) mkdir %s' | csh",destdir,destdir);
    system(command);
    fprintf(stderr,"The destination of the output is in directory %s.\n",destdir);

/* size of memory allocation for boundary element inputs. Increase by
     values of 1K bytes */
    pagesize = getpagesize()/sizeof(float);
    sprintf(command,"awk '{total += $1}; END {print total}' %s",be_file);
    unixpipe = popen(command,"r");
    fscanf(unixpipe,"%d",&memsum);
    if((pclose(unixpipe))!=0)
            prob_rpt("\n***Problem in closing unix pipe.***\n");
    while(memsum>pagesize) pagesize += pagesize;

/* allocate some memory */
    sinbet = vector(0,pagesize);
    cosbet = vector(0,pagesize);
    a = vector(0,pagesize);
    r = vector(0,2*pagesize);
    xm = vector(0,pagesize);
    ym = vector(0,pagesize);
    angle = vector(0,pagesize);

/* establish constants */
    g = 0.5*young_mod/(1.0+poisson);
    pr1 = 1.0 - 2.0*poisson;
    pr2 = 2.0*(1.0-poisson);
    pr3 = 3.0 - 4.0*poisson;
```

```
  con = 1.0/(4.0*Pi*(1.0-poisson));

/* Define locations, sizes, orientations, and boundary conditions of
     all boundary elements */
  bcond(&numbe,numbs,grid);

/* output the input */
  print_input(numbe);

/* allocate memory for system of linear equations */
  X = vector(0,4*numbe);
  Y = vector(0,4*numbe);
  C = matrix(0,4*numbe,0,4*numbe);

/* allocate memory for boundary element results */
  be.sigs = vector(0,2*numbe);
  be.sign = vector(0,2*numbe);
  be.us = vector(0,2*numbe);
  be.un = vector(0,2*numbe);
  be.ux = vector(0,2*numbe);
  be.uy = vector(0,2*numbe);

/* adjust stress boundary values to account for initial stress. */
  adjbv(numbe);

/* compute influence coefficients and set up system of algebraic equations */
  influence(numbe);

/* release some memory */
  free_vector(A,0,5);
  free_vector(B,0,5);

/* solve system of algebraic equations */
  fprintf(stderr,"Solving system of equations using LU Decomposition...\n");
  solve_lu(numbe);

/* release some memory */
  free_matrix(C,0,4*numbe,0,4*numbe);
  free_vector(Y,0,4*numbe);

/* calculate boundary displacements and stresses */
  fprintf(stderr,"Calculating boundary displacements and stresses...\n");
  bcalc(numbe,be);

/* allocate memory for other element input */
  xbeg = (numos<15) ? vector(0,15) : vector(0,numos+1);
  ybeg = (numos<15) ? vector(0,15) : vector(0,numos+1);
  xend = (numos<15) ? vector(0,15) : vector(0,numos+1);
  yend = (numos<15) ? vector(0,15) : vector(0,numos+1);
```

```
  numpb = (numos<15) ? ivector(0,15) : ivector(0,numos+1);

/* input specified element information */
  memsum = specin(numpb,numos);

/* compute displacements and stresses at specified points in the body. */
  if(numos>0) {

    /* allocate memory for other element output*/
      oe.ux = vector(0,memsum);
      oe.uy = vector(0,memsum);
      oe.xp = vector(0,memsum);
      oe.yp = vector(0,memsum);
      oe.sigxx = vector(0,memsum);
      oe.sigyy = vector(0,memsum);
      oe.sigxy = vector(0,memsum);

     /* do the calculations */
      fprintf(stderr,
          "Calculating specified point displacements and stresses...\n");
      fprintf(logfile,"        Ignored %d points!\n",
          (specpt(&npoint,numbe,oe,numpb,numos,noignore)));
  }

/* release some memory */
  free_vector(r,0,2*pagesize);
  free_vector(sinbet,0,pagesize);
  free_vector(cosbet,0,pagesize);
  free_vector(P,0,4);
  free_vector(U,0,10);
  free_vector(S,0,15);
  free_vector(X,0,2*numbe);

/* allocate some memory */
  be.sigma1 = vector(0,2*numbe);
  be.sigma3 = vector(0,2*numbe);
  be.theta = vector(0,2*numbe);
  oe.sigma1 = vector(0,npoint);
  oe.sigma3 = vector(0,npoint);
  oe.theta1 = vector(0,npoint);
  oe.theta3 = vector(0,npoint);

/* compute principal stresse for specified points */
  fprintf(stderr,"Calculating principle stresses...\n");
  prcpl_stress(numbe,npoint,oe,be);

/* release some memory */
  free_vector(a,0,pagesize);
```

```
/* output all final information */
  fprintf(stderr,"Outputing Information to disk...\n");
  output(numbe,npoint,grid,oe,be);

/* release some memory (what's left) */
  free_vector(be.sigs,0,2*numbe);
  free_vector(be.sign,0,2*numbe);
  free_vector(be.us,0,2*numbe);
  free_vector(be.un,0,2*numbe);
  free_vector(be.ux,0,2*numbe);
  free_vector(be.uy,0,2*numbe);
  free_vector(be.sigma1,0,2*numbe);
  free_vector(be.sigma3,0,2*numbe);
  free_vector(be.theta,0,2*numbe);
  free_vector(be.sigma1,0,npoint);
  free_vector(be.sigma3,0,npoint);
  free_vector(oe.theta1,0,npoint);
  free_vector(oe.theta3,0,npoint);

/* log the completion date */
  fprintf(logfile,"%s",sdate());

/* close the logfile */
  if((fclose(logfile))!=0)
         prob_rpt("\n***Problem in closing the log file.***\n");

/* clean up temporary stuff */
  sprintf(command,"mv logfile %s",destdir);
  system(command);
  sprintf(command,"echo 'if (-e AUTOGRID) rm AUTOGRID' | csh");
  system(command);
  return(0);
} /* end program */
```

(adjbv.c)

```c
static float sqrarg;
#define SQR(a) (sqrarg=(a),sqrarg*sqrarg)

/* global variables */
extern float *cosbet,*sinbet,*P,*r,g;

/* adjust stress boundary values to account for initial stress. */
void adjbv(numbe)

int numbe;

{
  int n;
  float sinb,cosb,sigs;

  for(n=1;n<=numbe;n++) {
    cosb = cosbet[n];
    sinb = sinbet[n];
    sigs = (P[2]-P[0])*sinb*cosb+P[1]*(SQR(cosb)-SQR(sinb));
    r[2*n-1] = 0.5*(r[2*n-1]-sigs)/g;   /* positive side of couple element */
    cosb = -cosbet[n];
    sinb = -sinbet[n];
    sigs = (P[2]-P[0])*sinb*cosb+P[1]*(SQR(cosb)-SQR(sinb));
    r[2*n] = 0.5*(r[2*n]-sigs)/g;       /* negative side of couple element */
  } /* end n loop */
} /* end adjbv() */
```

```
#include <math.h>
#include "smudge.h"
static float sqrarg;
#define SQR(a) (sqrarg=(a),sqrarg*sqrarg)

/* global variables */
extern float g,*cosbet,*sinbet,*X,*r,*P;

/* calculate boundary displacements and stresses */
void bcalc(numbe,be)

struct boundpoints be;
int numbe;
{
  int i,in,is,id,ic,ib,ia;
  float cosbi,sinbi;

/* do for all boundary elements */
  for(i=1;i<=numbe;i++) {
    in = 2*i;          /* negative side of couple element */
    is = in - 1;       /* positive side of couple element */
    id = 4*i;          /* id = U_N(+,-)         */
    ic = id-1;         /* ic = U_S(-)           */
    ib = ic-1;         /* ib = U_S(+)           */
    ia = ib-1;         /* ia = Sigma_N(+,-) */

/* do positive side of couple element */
    cosbi = cosbet[i];
    sinbi = sinbet[i];
    be.us[is] = X[ib];
    be.un[is] = X[id];
    be.ux[is] = be.us[is]*cosbi-be.un[is]*sinbi;
    be.uy[is] = be.us[is]*sinbi+be.un[is]*cosbi;
    be.sign[is] = 2.0*g*X[ia];
    be.sigs[is] = 2.0*g*r[is];
    be.sigs[is] += (P[2]-P[0])*sinbi*cosbi+P[1]*(SQR(cosbi)-SQR(sinbi));
    be.sign[is] += P[0]*SQR(sinbi)-2.0*P[1]*sinbi*cosbi+P[2]*SQR(cosbi);

/* do negative side of couple element */
    cosbi = -cosbet[i];
    sinbi = -sinbet[i];
    be.us[in] = X[ic];
    be.un[in] = -X[id];
    be.ux[in] = be.us[in]*cosbi-be.un[in]*sinbi;
    be.uy[in] = be.us[in]*sinbi+be.un[in]*cosbi;
    be.sigs[in] = 2.0*g*r[in];
    be.sign[in] = 2.0*g*X[ia];
    be.sigs[in] += (P[2]-P[0])*sinbi*cosbi+P[1]*(SQR(cosbi)-SQR(sinbi));
    be.sign[in] += P[0]*SQR(sinbi)-2.0*P[1]*sinbi*cosbi+P[2]*SQR(cosbi);
```

```
    } /* end of i loop */
} /* end bcalc() */
```

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "smudge.h"
static float sqrarg;
#define SQR(a) (sqrarg=(a),sqrarg*sqrarg)

/* global variables */
extern float *sinbet,*cosbet,*xm,*ym,*a,*r;
extern char be_file[30];
extern double sqrt();
extern char destdir[80];

/* Define locations, sizes, orientations, and boundary conditions of
     all boundary elementes */
void bcond(numbe,numbs,grid)

int *numbe,numbs;
struct griddata grid;
{
  int ne,m,n,num;
  float xbeg,ybeg,xend,yend,bvs,bvn,delx,dely,sw;
  FILE *data,*plotxypipe;
  char filename[100];

/* open a pipe to plotxy */
  sprintf(filename,"%s/plotxy.model",destdir);
  plotxypipe = fopen(filename,"w");

/* initialize plotxy plot */
  fprintf(plotxypipe,"file *\n");
  fprintf(plotxypipe,"output model.ps\n");
  fprintf(plotxypipe,"mode 2\n");
  fprintf(plotxypipe,"frame none\n");
  fprintf(plotxypipe,"color 0\n");
  fprintf(plotxypipe,"xlim 5.650 %f %f\n",
     grid.xorigin,grid.xorigin+grid.length);
  fprintf(plotxypipe,"ylim 5.698 %f %f\n",
     grid.yorigin,grid.yorigin+grid.length);
  fprintf(plotxypipe,"read %d\n",numbs*3);

/* open data file */
  data = fopen(be_file,"r");

/* check to see if the file is open, if not exit */
  if(data==NULL) error_rpt("Error in opening up boundary data file.");

/* set the number of boundary elements to zero */
  *numbe = 0;
```

```
/* read in the boundary element information */
  for(n=1;n<=numbs;n++) {
    fscanf(data,"%d%f%f%f%f%f%f",&num,&xbeg,&ybeg,&xend,&yend,&bvs,&bvn);
    fprintf(plotxypipe,"%f %f\n%f %f\n999999. 999999.\n",xbeg,ybeg,xend,yend);

   /* calculate boundary element x, y, and radius lengths */
    delx = (xend-xbeg)/num;
    dely = (yend-ybeg)/num;
    sw = sqrt((double)(SQR(delx)+SQR(dely)));

    for(ne=1;ne<=num;ne++) { /* seperate segment into 'num' boundary elements */
      m = ++*numbe;
      xm[m] = xbeg+0.5*(2.0*(float)ne-1.0)*delx;
      ym[m] = ybeg+0.5*(2.0*(float)ne-1.0)*dely;
      a[m] = 0.5*sw;
      sinbet[m] = (sw>0.0) ? dely/sw : 0.0;
      cosbet[m] = (sw>0.0) ? delx/sw : 0.0;
      r[2*m-1] = bvs;
      r[2*m] = bvn;
    } /* end ne loop */
  } /* end n loop */

/* close the data file */
  if(fclose(data)!=0)
    prob_rpt("***Problem in closing boundary element input file.***");

/* finish the plotxy plot */
  fprintf(plotxypipe,"plot 1.529 4.513\n");
  fprintf(plotxypipe,"stop\n");

/* close the plotxy pipe */
  if(fclose(plotxypipe)!=0)
    prob_rpt("***Problem in closing plotxy pipe.***");
} /* end of bcond() */
```

(coeff.c)

```
#include <math.h>
static float sqrarg;
#define SQR(a) (sqrarg=(a),sqrarg*sqrarg)
#define Pi 3.14159265359
#undef TEST

/* global variables */
extern float poisson,young_mod,*A,*B;
extern float g,pr1,pr2,pr3,con;

void coeff(xi,yi,xj,yj,aj,cosb,sinb,cosbi,sinbi)

float xi,yi,xj,yj,aj,cosb,sinb;
float cosbi,sinbi;

{
  float cxb,cxbpa,cxbma,cyb,cosg,sing,r1s,r2s,fl1,fl2,tb[6],
        asst,asnt,anst,annt,bsst,bsnt,bnst,bnnt;
  extern float Log();

/* establish local constants */
  cxb = (xi-xj)*(cosb)+(yi-yj)*(sinb);
  cxbpa = cxb+aj;
  cxbma = cxb-aj;
  cyb = -(xi-xj)*(sinb)+(yi-yj)*(cosb);
  cosg = cosbi*(cosb)+sinbi*(sinb);
  sing = sinbi*(cosb)-cosbi*(sinb);
  r1s = SQR(cxbma)+SQR(cyb);
  r2s = SQR(cxbpa)+SQR(cyb);
  fl1 = r1s>0.0 ? 0.5*log((double)r1s) : 0.0 ;
  fl2 = r2s>0.0 ? 0.5*log((double)r2s) : 0.0 ;
  tb[2] = -con*(fl1-fl2);
  tb[3] = (cyb==0.0) ? ((fabs((double)cxb)<aj) ? con*Pi : 0.0 ) :
     con*(atan(cxbpa/cyb)-atan(cxbma/cyb));
 #ifdef TEST
  tb[1] = -cyb*tb[3]+con*(cxbma*fl1-cxbpa*fl2)+con*2*aj;;
 #else
  tb[1] = -cyb*tb[3]+con*(cxbma*fl1-cxbpa*fl2);
 #endif
  tb[4] = con*(cyb/r1s-cyb/r2s);
  tb[5] = con*(cxbma/r1s-cxbpa/r2s);

  asst = pr2*cosg*tb[3]+pr1*sing*tb[2]+cyb*(sing*tb[4]+cosg*tb[5]);
  asnt = -pr1*cosg*tb[2]+pr2*sing*tb[3]+cyb*(cosg*tb[4]-sing*tb[5]);
  anst = -pr2*sing*tb[3]+pr1*cosg*tb[2]+cyb*(cosg*tb[4]-sing*tb[5]);
  annt = pr1*sing*tb[2]+pr2*cosg*tb[3]-cyb*(sing*tb[4]+cosg*tb[5]);

 #ifdef TEST
  bsst = pr3*cosg*tb[1]+cyb*(sing*tb[2]-cosg*tb[3])+2*aj*cosg*con;
```

```
#else
  bsst = pr3*cosg*tb[1]+cyb*(sing*tb[2]−cosg*tb[3]);
#endif
  bsnt = pr3*sing*tb[1]+cyb*(cosg*tb[2]+sing*tb[3]);
#ifdef TEST
  bnst = −pr3*sing*tb[1]+cyb*(cosg*tb[2]+sing*tb[3])−2*aj*sing*con;
#else
  bnst = −pr3*sing*tb[1]+cyb*(cosg*tb[2]+sing*tb[3]);
#endif
  bnnt = pr3*cosg*tb[1]−cyb*(sing*tb[2]−cosg*tb[3]);

  /*
    These values are the actual analytic solutions, however they are not
    used by Croach and Starfield and are overbearing except when the
    length of the boundary elements (aj) is extremely small.

    tb[1] = −cyb*tb[3]+con*(cxbma*fl1−cxbpa*fl2)+con*2*aj;
    bsst = pr3*cosg*tb[1]+cyb*(sing*tb[2]−cosg*tb[3])+2*aj*cosg/g*con;
    bnst = −pr3*sing*tb[1]+cyb*(cosg*tb[2]+sing*tb[3])−2*aj*sing/g*con;
  */

  A[0] = asst;
  A[1] = asnt;
  A[2] = anst;
  A[3] = annt;

  B[0] = bsst;
  B[1] = bsnt;
  B[2] = bnst;
  B[3] = bnnt;

} /* end of coeff() */
```

(elmtplt.c)

```
#include <stdio.h>
#include <string.h>
#include <math.h>

void minmax(min,max,benum,value)
float *value,*min,*max;
int benum;
{
  int i;
  char yorn[1];

  for(i=1;i<=benum;i++) {
    if((*max)<value[i]) *max = value[i];
    if((*min)>value[i]) *min = value[i];
  }
  fprintf(stderr,"A Y-axis range from %f to %f has been chosen.\n",*min,*max);
  fprintf(stderr,"\tAccept (y/n)? ");
  scanf("%s",yorn);
  if(*yorn!='y') {
    fprintf(stderr,"Enter the desired minimum and maximum: ");
    scanf("%f%f",min,max);
  }
}

void uplot(plotname,dec)
int dec;
char plotname[30];
{
  FILE *xypipe,*input;
  int numbe,i,j,count=0,benum,num,type,xlab=0,ylab=0,xframe=1,yframe=1;
  float *beux,*beuy,*beus,*beun,*besign,*besigs,max=-100000,min=100000.,
    xinch,yinch;
  char junk[80],plotprog[80],xlabel[80],ylabel[80],query[10],tempstr[80];
  extern float *vector();

  input = fopen("Bdry_Elements","r");
  fgets(junk,80,input);
  fgets(junk,80,input);
  fgets(junk,80,input);
  fgets(junk,80,input);
  fgets(junk,80,input);
  fscanf(input,"\n");

  sprintf(plotprog,"wc Bdry_Elements");
  xypipe = popen(plotprog,"r");
  fscanf(xypipe,"%d",&numbe);
  pclose(plotprog);
  numbe-=5;
  fprintf(stderr,"\n%d boundary segments...\n",numbe);
```

```
  beus = vector(0,numbe);
  beun = vector(0,numbe);
  beux = vector(0,numbe);
  beuy = vector(0,numbe);
  besign = vector(0,numbe);
  besigs = vector(0,numbe);

  sprintf(plotprog,"/home/griffy/purbut/bin/plotxy.dyn -Kb %f\n",
    (numbe)*4/1000.+.1);

  benum = 1;
  count = 1;
  for(i=1;i<=numbe;i++) {
    fscanf(input,"%d%f%f%f%f%f%f",&num,&beus[benum],&beun[benum],
         &beux[benum],&beuy[benum],&besigs[benum],&besign[benum]);
    if(count++==dec) {
         benum++;
         count=1;
    }
  }
  benum--;
  fclose(input);
  fprintf(stderr,"%d boundary elements...\n",benum);

  fprintf(stderr,"\tEnter the x and y in inches: ");
  scanf("%f%f",&xinch,&yinch);

/* query for a label along the y-axis */
  fprintf(stderr,"\tDo you want an x-label (y/n): ");
  scanf("%s",query);
  if(strncasecmp(query,"y",1)==0) {
    xlab = 1;
    fprintf(stderr,"\t\tEnter the x-label (end string with \"><\"): ");
    scanf("%s",xlabel);
    strcpy(tempstr," ");
    while(strncmp(tempstr,"><",2)!=0) {
      scanf("%s",tempstr);
      if(strncmp(tempstr,"><",2)!=0) {
           strcat(xlabel," ");
           strcat(xlabel,tempstr);
      }
    }
    printf("\nThe X-label is: \"%s\"\n",xlabel);
  }

/* query for a label along the y-axis */
  fprintf(stderr,"\tDo you want a y-label (y/n): ");
  scanf("%s",query);
```

```
if(strncasecmp(query,"y",1)==0) {
  ylab = 1;
  fprintf(stderr,"\t\tEnter the y-label (end string with \"><\"): ");
  scanf("%s",ylabel);
  strcpy(tempstr," ");
  while(strncmp(tempstr,"><",2)!=0) {
    scanf("%s",tempstr);
    if(strncmp(tempstr,"><",2)!=0) {
        strcat(ylabel," ");
        strcat(ylabel,tempstr);
    }
  }
  printf("\nThe Y-label is: \"%s\"\n",ylabel);
}

/* query for x-axis numbering */
 fprintf(stderr,"\tDo you want a x-axis numbering (y/n): ");
 scanf("%s",query);
 if(strncasecmp(query,"y",1)==0) xframe = 0;

/* query for y-axis numbering */
 fprintf(stderr,"\tDo you want a y-axis numbering (y/n): ");
 scanf("%s",query);
 if(strncasecmp(query,"y",1)==0) yframe = 0;

/* query for type of plot */
 type = 0;
 fflush(stdin);
 fprintf(stderr,"Plot:\n\t1)Us\n\t2)Un\n\t3)Ux\n\t4)Uy\n\t5)Sigma_S\n\t6)Sigma_N\t: ");
 scanf("%d",&type);

 xypipe = popen(plotprog,"w");
 fflush(stdout);
 fflush(stderr);
 fprintf(xypipe,"file *\n");
 fprintf(xypipe,"output %s\n",plotname);
 fprintf(xypipe,"frame on\n");
 fprintf(xypipe,"mode 1\n");
 if(xframe&&yframe) {
   fprintf(stderr,"\nNo x-axis or y-axis numbering...\n");
   fprintf(xypipe,"frame -xnum -ynum \n");
 } else if(xframe) {
   fprintf(stderr,"\nNo x-axis numbering...\n");
   fprintf(xypipe,"frame -xnum \n");
 } else if(yframe) {
   fprintf(stderr,"\nNo y-axis numbering...\n");
   fprintf(xypipe,"frame -ynum \n");
 }
 fprintf(xypipe,"xlim %f 1 %d\n",xinch,benum/2);
```

```
switch (type) {
        case 1:
          fprintf(stderr,"\nPlotting Us\n");
          minmax(&min,&max,benum,beus);
        fprintf(xypipe,"ylim %f %f %f\n",yinch,min,max);
        fprintf(xypipe,"read %d\n",benum/2);
        for(i=1;i<=benum;i+=2) fprintf(xypipe,"%f\n",beus[i]);
          break;
        case 2:
          fprintf(stderr,"\nPlotting Un\n");
          minmax(&min,&max,benum,beun);
        fprintf(xypipe,"ylim %f %f %f\n",yinch,min,max);
        fprintf(xypipe,"read %d\n",benum/2);
        for(i=1;i<=benum;i+=2) fprintf(xypipe,"%f\n",beun[i]);
          break;
        case 3:
          fprintf(stderr,"\nPlotting Ux\n");
          minmax(&min,&max,benum,beux);
        fprintf(xypipe,"ylim %f %f %f\n",yinch,min,max);
        fprintf(xypipe,"read %d\n",benum/2);
        for(i=1;i<=benum;i+=2) fprintf(xypipe,"%f\n",beux[i]);
          break;
        case 4:
          fprintf(stderr,"\nPlotting Uy\n");
          minmax(&min,&max,benum,beuy);
        fprintf(xypipe,"ylim %f %f %f\n",yinch,min,max);
        fprintf(xypipe,"read %d\n",benum/2);
        for(i=1;i<=benum;i+=2) fprintf(xypipe,"%f\n",beuy[i]);
          break;
        case 5:
          fprintf(stderr,"\nPlotting Sigma_S\n");
          minmax(&min,&max,benum,besigs);
        fprintf(xypipe,"ylim %f %f %f\n",yinch,min,max);
        fprintf(xypipe,"read %d\n",benum/2);
        for(i=1;i<=benum;i+=2) fprintf(xypipe,"%f\n",besigs[i]);
          break;
        case 6:
          fprintf(stderr,"\nPlotting Sigma_N\n");
          minmax(&min,&max,benum,besign);
        fprintf(xypipe,"ylim %f %f %f\n",yinch,min,max);
        fprintf(xypipe,"read %d\n",benum/2);
        for(i=1;i<=benum;i+=2) fprintf(xypipe,"%f\n",besign[i]);
          break;
     default:
          fprintf(stderr,"****Input range is (0-6)****\n");
          break;
}
if(xlab) fprintf(xypipe,"xlabel %s\n",xlabel);
if(ylab) fprintf(xypipe,"ylabel %s\n",ylabel);
```

```
  /* fprintf(xypipe,"character 0.0\n");*/
  fprintf(xypipe,"plot 1.529 4.513\n");
  fprintf(xypipe,"stop\n");
  pclose(xypipe);
}
```

main

```
main (argc,argv)
int argc;
char *argv[];

{
  int n,dec;
  char plotname[30];

  dec = 1;
  for(n=1;n<argc;n++)
   if(!strncmp(argv[n],"-dec",4)) dec = (int)(strtod(argv[++n],(char **)NULL));

  fprintf(stderr,"Enter the filename for the plot: ");
  scanf("%s",plotname);
  uplot(plotname,dec);
}
```

(field_plt.c)

```
#include <stdio.h>
#include <math.h>

void uplot(xmag,ymag,plotname,x0,x1,y0,y1,umax)
float xmag,ymag,x0,x1,y0,y1,umax;
char plotname[30];
{
  FILE *xypipe,*input,*dpipe;
  int numbe,numoe,i,j,count=0,benum,xnum,ynum;
  unsigned int flag;
  float *oxp,*oyp,*oux,*ouy,tol;
  char junk[20],plotprog[80];
  extern float *vector();

/* open the input file */
  input = fopen("primer.dat","r");
  if(input==NULL) {
    fprintf(stderr,"\n***\tError opening input file primer.dat ***\n");
    fprintf(stderr,
      "\t     To create this file, use the csh-script field_primer.\n");
    exit();
  }

/* find out how many boundary elements there are */
  fscanf(input,"%d",&numbe);
  fgets(junk,20,input);

/* find out how many specified elements there are */
  fscanf(input,"%d",&numoe);
  fgets(junk,20,input);

/* open a pipe to plotxy */
  sprintf(plotprog,"/home/griffy/purbut/bin/plotxy.dyn -Kb %f\n",
    (3*numoe+2*numbe)*4/1000.+.1);

/* allocate space for the original grid and the displacements */
  oxp = vector(0,2*numoe);
  oyp = vector(0,2*numoe);
  oux = vector(0,2*numoe);
  ouy = vector(0,2*numoe);

/* read in the data 'parallel' to the Y-axis*/
  xnum = 0;
  for(i=1;i<=numoe;i++) {
    fscanf(input,"%f%f%f%f",&oxp[i],&oyp[i],&oux[i],&ouy[i]);
    xnum++;
    if(oxp[i]!=oxp[i-1]) xnum++;
  }
```

```
/* read in the data 'parallel' to the X-axis*/
  ynum = 0;
  for(i=numoe+1;i<=2*numoe;i++) {
    fscanf(input,"%f%f%f%f",&oxp[i],&oyp[i],&oux[i],&ouy[i]);
    ynum++;
    if(oyp[i]!=oyp[i-1]) ynum++;
  }
  fclose(input);

/* output grid information to plotxy */
  xypipe = popen(plotprog,"w");
  fprintf(xypipe,"file *\n");
  fprintf(xypipe,"output %s\n",plotname);
  fprintf(xypipe,"frame on\n");
  fprintf(xypipe,"mode 2\n");
  fprintf(xypipe,"xlim 5.650 %f %f\n",x0,x1);
  fprintf(xypipe,"ylim 5.698 %f %f\n",y0,y1);
  fprintf(xypipe,"read %d\n",xnum+ynum);

/* do the lines 'parallel' to the Y-axis first */
  flag = 0;
  for(i=1;i<=numoe;i++) {
    if(oxp[i]!=oxp[i+1]) {
      fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
      fprintf(xypipe,"999. 999.\n");
        flag = 0;
    } else {
        if(flag) {
          tol = ((oxp[i]+oux[i]*xmag)-(oxp[i-1]+oux[i-1]*xmag))*
                ((oxp[i]+oux[i]*xmag)-(oxp[i-1]+oux[i-1]*xmag))+
                ((oyp[i]+ouy[i]*ymag)-(oyp[i-1]+ouy[i-1]*ymag))*
                ((oyp[i]+ouy[i]*ymag)-(oyp[i-1]+ouy[i-1]*ymag));
        } else {
          tol = 0.;
          flag = 1;
        }
      if(sqrt(tol)<=umax) {
        fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
      } else {
        fprintf(xypipe,"999. 999.\n");
      }
    }
  }

/* do the lines 'parallel' to the X-axis second */
  flag = 0;
  for(i=numoe+1;i<=2*numoe;i++) {
    if(oyp[i]!=oyp[i+1]) {
      fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
```

```
        fprintf(xypipe,"999. 999.\n");
          flag = 0;
      } else {
          if(flag) {
            tol = ((oxp[i]+oux[i]*xmag)-(oxp[i-1]+oux[i-1]*xmag))*
                  ((oxp[i]+oux[i]*xmag)-(oxp[i-1]+oux[i-1]*xmag))+
                  ((oyp[i]+ouy[i]*ymag)-(oyp[i-1]+ouy[i-1]*ymag))*
                  ((oyp[i]+ouy[i]*ymag)-(oyp[i-1]+ouy[i-1]*ymag));
          } else {
            tol = 0.;
            flag = 1;
          }
        if(sqrt(tol)<=umax) {
          fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
        } else {
          fprintf(xypipe,"999. 999.\n");
        }
      }
    }

/*
  sprintf(plotprog,"head -%d Disp.gph | tail -%d",numbe+2,numbe);
  dpipe = popen(plotprog,"r");
  for(i=1;i<=numoe;i++)
    fscanf(dpipe,"%f%f%f%f",&oxp[i],&oyp[i],&oux[i],&ouy[i]);
  if((pclose(dpipe))!=0)
          prob_rpt("\n***Problem in closing unix pipe.***\n");

  fprintf(xypipe,"save\nread %d\n",numbe);
  for(i=1;i<=numbe;i+=2)
    fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
  fprintf(xypipe,"999.999 999.999\n");
  for(i=2;i<=numbe;i+=2)
    fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
*/

/* final plot information */
  fprintf(xypipe,"note (1.40 -2.0 in) X magnification: %5.2f\n",xmag);
  fprintf(xypipe,"note (1.40 -1.5 in) Y magnification: %5.2f\n",ymag);
  fprintf(xypipe,"plot 1.529 4.513\n");
  fprintf(xypipe,"stop\n");
  if((pclose(xypipe))!=0)
          prob_rpt("\n***Problem in closing plotxy pipe.***\n");

}

void main()

{
```

```
  int n;
  float xmag=1.,ymag=1.,x0,x1,y0,y1,umax;
  char plotname[30];

/* Define plot properties */
  fprintf(stderr,"Enter the filename for the plot: ");
  scanf("%s",plotname);
  fprintf(stderr,"\n");
  fprintf(stderr,"Enter the limts of the graph (X0 X1 Y0 Y1):");
  scanf("%f%f%f%f",&x0,&x1,&y0,&y1);
  fprintf(stderr,"\nEnter X-magnification factor: ");
  scanf("%f",&xmag);
  fprintf(stderr,"\nEnter Y-magnification factor: ");
  scanf("%f",&ymag);
  fprintf(stderr,"\nEnter Maximum Displacement Value: ");
  scanf("%f",&umax);
  uplot(xmag,ymag,plotname,x0,x1,y0,y1,umax);
}
```

(field_primer)

```
#!/bin/csh
# This csh script takes the displacement output file from SMUDGE and
# modifies it so that a mesh displacement field can be plotted using
# field_plt

if (! -e Disp.gph) then
  echo "The SMUDGE produced Disp.gph file must be present to run this script."
  exit();
endif

# rm temporary files if they should exist
if (-e primer1.tmp) rm primer1.tmp
if (-e primer2.tmp) rm primer2.tmp
set count = 0
set step = `grep elements logfile | awk '{print $4}'`

# get the number of boundary elements
set num = `head -1 Disp.gph | awk '{print $1}'`
@ num += 3

# place the specified element data into the temporary files
tail +$num Disp.gph > primer1.tmp  # data parallel to the Y-axis
sort -bn primer1.tmp > primer2.tmp # data parallel to the X-axis

# place information header onto the output file
if (-e primer.dat) rm primer.dat
head -2 Disp.gph > primer.dat

# correctly sort the data for plotxy
while($count <= $step)
  set endn = $count
  @ endn *= $step
  set start = $endn
  @ start += 1
  echo $count of $step complete...
  tail +$start primer2.tmp | head -$step | sort -bn +1 >> primer.dat
  @ count += 1
end
cat primer1.tmp >> primer.dat

# rm temporary files
rm primer1.tmp
rm primer2.tmp
```

```
/* display man page for SMUDGE */

char *path = SMUDGEPATH;

void help()
{
  void exit();
  char command[240];


  sprintf(command,
    "/usr/bin/nroff -man %s/smudge.man | /usr/local/bin/less",path);
  system(command);
  exit(1);
}
```

```
#include "smudge.h"
#undef TEST

/* global variables */
extern float **C,*A,*B,*Y,*r,*xm,*ym,*a,*r,*sinbet,*cosbet;

/* compute influence coefficients and set up system of algebraic equations

   This system is determined from:

  A) sum[ Bss*Sigma_S] = -sum[ Bsn*Sigma_N] + sum[ Ass*Us] + sum[ Asn*Un]
  B) sum[ Bns*Sigma_S] = -sum[ Bnn*Sigma_N] + sum[ Ans*Us] + sum[ Ann*Un]

          summed from i = 1 to N, where the B and A matrices are of
             dimension ij and the SIGMA and U vectors are of dimension
             j. Depending on the known parameters, these equations can
             be arranged so that a system y = Cx is formed, where y are
             the knowns, C is the matrix of the influence coefficients
             A and B, and x are the unknowns.

*/

void influence(numbe)

int numbe;

{
   int i,ia,ib,ic,id,j,ja,jb,jc,jd;
   float aj,xi,xj,yi,yj,cosbi,cosbj,sinbi,sinbj;

   for(i=1;i<=numbe;i++) {
      id = 4*i;
      ic = id-1;
      ib = ic-1;
      ia = ib-1;
      Y[id] = 0.0;
      Y[ic] = 0.0;
      Y[ib] = 0.0;
      Y[ia] = 0.0;

   /* location and orientation of middle point of ith boundary element */
      xi = xm[i];
      yi = ym[i];

      for(j=1;j<=numbe;j++) {
         jd = 4*j;    /* U_n(+,-)     */
         jc = jd-1;   /* U_s(-)       */
         jb = jc-1;   /* U_s(+)       */
         ja = jb-1;   /* Sigma_N(+,-) */
```

```
/* location of middle point of jth boundary element */
    xj = xm[j];
    yj = ym[j];
    aj = a[j];

/* do calculation for positive side of double couple element */
    cosbi = cosbet[i];
    sinbi = sinbet[i];
    cosbj = cosbet[j];
    sinbj = sinbet[j];
    coeff(xi,yi,xj,yj,aj,cosbj,sinbj,cosbi,sinbi);
    C[ia][ja] = -B[1];
    C[ia][jb] = A[0];
    C[ia][jc] = 0.;
    C[ia][jd] = A[1];
    C[ic][ja] = -B[3];
    C[ic][jb] = A[2];
    C[ic][jc] = 0.;
    C[ic][jd] = A[3];
    Y[ia] += B[0]*r[2*i-1];
    Y[ic] += B[2]*r[2*i-1];

/* do calculation for negative side of double couple element */
    cosbi = -cosbet[i];
    sinbi = -sinbet[i];
    cosbj = -cosbet[j];
    sinbj = -sinbet[j];
    coeff(xi,yi,xj,yj,aj,cosbj,sinbj,cosbi,sinbi);
    C[ib][ja] = -B[1];
    C[ib][jb] = 0.;
    C[ib][jc] = A[0];
    C[ib][jd] = -A[1];
    C[id][ja] = -B[3];
    C[id][jb] = 0.;
    C[id][jc] = A[2];
    C[id][jd] = -A[3];
    Y[ib] += B[0]*r[2*i];
    Y[id] += B[2]*r[2*i];

  }  /* end j loop */
 } /* end i loop */
} /* end influence() */
```

```
#include <stdio.h>
#include "smudge.h"

extern float poisson,young_mod;
extern float *P;
extern char ps_file[30],be_file[30],oe_file[30],*con_file,destdir[80];
extern FILE *logfile;

void autogrid(xorigin,yorigin,length,step)

float xorigin,yorigin,length;
int step;

{
  float lenstep;
  int i;
  FILE *outfile;

/* open the temporary output file AUTOGRID */
  outfile = fopen("AUTOGRID","w");

/* check to see whether the file is actually open */
  if(outfile==NULL) error_rpt("Error in opening up AUTOGRID file.");

/* create the AUTOGRID file with the same format as specified point file */
  lenstep = length/step;
  for(i=0;i<=step;i++) {
    fprintf(outfile,"%f %f %f %f %d\n",xorigin,yorigin+i*lenstep,
         xorigin+length,yorigin+i*lenstep,step);
  }
  fprintf(outfile,"\n Origin: %f %f",xorigin,yorigin);
  fprintf(outfile,"\n Length: %f",length);
  fprintf(outfile,"\n Spacing: %d",step);
  fprintf(outfile,"\n Number of elements: %d",step+1);

/* close the AUTOGRID file */
  if((fclose(outfile))!=0)
    prob_rpt("\n***Problem in closing AUTOGRID file.***\n");

/* output information to the logfile */
  fprintf(logfile,"\t\tOrigin: %f %f\n",xorigin,yorigin);
  fprintf(logfile,"\t\tLength: %f\n",length);
  fprintf(logfile,"\t\tSpacing: %d\n",step);
  fprintf(logfile,"\t\tNumber of elements: %d\n",step+1);

} /* end autogrid() */

void instructions(grid,numbs,numos)
```

```
struct griddata grid;
int *numbs,*numos;

{
  char title[80],junk[80];
  FILE *data;
  float xorigin,yorigin,length,vp,vs,rho;
  int count=0,step;

/* open the instruction file */
  data = fopen(con_file,"r");

/* check to see whether the file is actually open */
  if(data==NULL) error_rpt("Error in opening up instruction file.");

/* load in the instructions */
  fgets(title,80,data);
  fgets(junk,80,data);
  fscanf(data,"%d%d\n",numbs,numos);
  fgets(junk,80,data);
  fscanf(data,"%f%f%f\n",&vp,&vs,&rho);
  poisson = (vp*vp-2*vs*vs)/2./(vp*vp-vs*vs);
  young_mod = (rho*vs*vs/(vp*vp-vs*vs)*(3.*vp*vp-4*vs*vs))/1000000.0;
  fgets(junk,80,data);
  fscanf(data,"%f%f%f\n",&P[0],&P[1],&P[2]);            /* Field Stress Values */
  fgets(junk,80,data);
  fscanf(data,"%s\n",be_file);          /* boundary element input file */
  fgets(junk,80,data);
  fscanf(data,"%s\n",oe_file);          /* specified point input file */

/* check to see if an automatic gridding was specified */
  if(!strcmp(oe_file,"AUTOGRID"))  {
    count = fscanf(data,"%f%f%f%d\n",&xorigin,&yorigin,&length,&step);
    *numos = step+1;
    if(count!=4) error_rpt(
        "Correct AUTOGRID usage is: Xorigin Yorigin Length Step\n");
  } /* end if */
```

```
/* load in some more instructions */
  fgets(junk,80,data);
  fscanf(data,"%s",destdir); /* output directory */

/* Be cautious just for sport and check for file closure */
  if((fclose(data))!=0)
    prob_rpt("\n***Problem in closing control file.***\n");

/* output what was read to a logfile for confirmation purposes */
  fprintf(logfile,"%s",title);
  fprintf(logfile,"       Boundary Element input file is %s.\n",be_file);
```

```
fprintf(logfile,"      Other Element input file is %s.\n",oe_file);

if(!strcmp(oe_file,"AUTOGRID")) {
   autogrid(xorigin,yorigin,length,step);
} else {
  /*
      If the other elements were created by any other grid program, copy
         the statistics as if it were AUTOGRIDed.  This makes my plotting
      programs work better.  The format for these values is:

          Origin: xorigin yorigin
          Length: length
          Spacing: spacing
          Number of elements: spacing+1

      where xorigin and yorigin are the lower left coordinates of the
      grid, length is the number how long a side of the square is, spacing
      is the number of grid squares on a side, and number of elements is
      the the spacing plus one (it takes n+1 nodes to create n line
      segments).
  */

   sprintf(junk,"tail -4 %s",oe_file);
   data = popen(junk,"r");
   count = fscanf(data,"%s%f%f",junk,&xorigin,&yorigin);

   /* be retentive about grid.c output confirmation */
   if((count==3)&&(!strncmp(junk,"Origin:",7))) {
      fscanf(data,"%s%f",junk,&length);
      fscanf(data,"%s%d",junk,&step);
      /* output what was read to a logfile for confirmation purposes */
      fprintf(logfile,"\t\tOrigin: %f %f\n",xorigin,yorigin);
      fprintf(logfile,"\t\tLength: %f\n",length);
      fprintf(logfile,"\t\tSpacing: %d\n",step);
      fprintf(logfile,"\t\tNumber of elements: %d\n",step+1);
   } /* end if */

   /* close the unix pipe */
   if((pclose(data))!=0)
            prob_rpt("\n***Problem in closing unix pipe.***\n");
} /* end else */

/* keep track of grid stuff */
 grid.xorigin = xorigin;
 grid.yorigin = yorigin;
 grid.length = length;
 grid.step = step;

/* output what was read to a logfile for confirmation purposes */
```

```
    fprintf(logfile,"        The target directory is %s.\n",destdir);
    fprintf(logfile,"        The number of boundary line segments = %d.\n",*numbs);
    fprintf(logfile,"        The number of other line segments = %d.\n",*numos);
    fprintf(logfile,"        P-wave Velocity = %f m/s\n",vp);
    fprintf(logfile,"        S-wave Velocity = %f m/s\n",vs);
    fprintf(logfile,"        Density = %f kg/m^3\n",rho);
    fprintf(logfile,"        Poisson's Ratio = %f\n",poisson);
    fprintf(logfile,"        Young's Modulus = %f MPa\n",young_mod);
    fprintf(logfile,"        XX Component of Field Stress = %f MPa\n",P[0]);
    fprintf(logfile,"        XY Component of Field Stress = %f MPa\n",P[1]);
    fprintf(logfile,"        YY Component of Field Stress = %f MPa\n",P[2]);
    fflush(logfile);
} /* end instructions() */
```

```
void lubksb(a,n,indx,b)

float **a,*b;
int n,*indx;
{
	int i,ii=0,ip,j;
	float sum;

	for (i=1;i<=n;i++) {
		ip=indx[i];
		sum=b[ip];
		b[ip]=b[i];
		if (ii)
			for (j=ii;j<=i-1;j++) sum -= a[i][j]*b[j];
		else if (sum) ii=i;
		b[i]=sum;
	}
	for (i=n;i>=1;i--) {
		sum=b[i];
		for (j=i+1;j<=n;j++) sum -= a[i][j]*b[j];
		b[i]=sum/a[i][i];
	}
}
/* (C) Copr. 1986-92 Numerical Recipes Software .-)-4'%10(9p#3D0>4^c. */
```

```
#include <math.h>
#undef NRANSI
/*#include <local/nrutil.h>*/
#include "nrutil.h"
#define TINY 1.0e-20;

/* COND is defined if the condition number should be calculated */
#undef COND
#ifdef COND
#include <stdio.h>
#define PRECISION 1.0e-6
#endif

void ludcmp(a,n,indx,d)

float **a,*d;
int n,*indx;
{
        int i,imax,j,k;
        float big,dum,sum,temp,small;
        float *vv;

        vv=vector(1,n);
        *d=1.0;
        for (i=1;i<=n;i++) {
                big=0.0;
                small=1./TINY;
                for (j=1;j<=n;j++) {
                        if ((temp=fabs(a[i][j])) > big) big=temp;
#ifdef COND
                        if ((temp=fabs(a[i][j])) < small) small=temp;
#endif
                }
                if (big == 0.0) nrerror("Singular matrix in routine ludcmp");
                vv[i]=1.0/big;
#ifdef COND
                if ((temp=fabs(small/big)) <= PRECISION) printf(
                  "Matrix ill-conditioned {i = %d, cond# = %f}...\n"
                  ,i,temp);
#endif
        }
        for (j=1;j<=n;j++) {
                for (i=1;i<j;i++) {
                        sum=a[i][j];
                        for (k=1;k<i;k++) sum -= a[i][k]*a[k][j];
                        a[i][j]=sum;
                }
                big=0.0;
                for (i=j;i<=n;i++) {
```

```
				sum=a[i][j];
				for (k=1;k<j;k++)
					sum -= a[i][k]*a[k][j];
				a[i][j]=sum;
				if ( (dum=vv[i]*fabs(sum)) >= big) {
					big=dum;
					imax=i;
				}
			}
			if (j != imax) {
				for (k=1;k<=n;k++) {
					dum=a[imax][k];
					a[imax][k]=a[j][k];
					a[j][k]=dum;
				}
				*d = -(*d);
				vv[imax]=vv[j];
			}
			indx[j]=imax;
			if (a[j][j] == 0.0) a[j][j]=TINY;
			if (j != n) {
				dum=1.0/(a[j][j]);
				for (i=j+1;i<=n;i++) a[i][j] *= dum;
			}
		}
		free_vector(vv,1,n);
}
#undef TINY
#undef NRANSI
/* (C) Copr. 1986-92 Numerical Recipes Software .-)-4'%10(9p#3D0>4^c. */
```

```
#include <malloc.h>
#include <stdio.h>
#include <math.h>
#include "smudge.h"
#define BIG 1.0e20
#define SMALL 1.0e-8
#define Pi 3.14159265359

/* report a problem and exit */
void error_rpt(error_text)
char error_text[];
{
 void exit();

 fprintf(stderr,"\nRun-time error...\n");
 fprintf(stderr,"        %s\n",error_text);
 fprintf(stderr,"...now exiting to system...\n");
 exit(1);
} /* end error_rpt() */

/* report a problem and return to program */
void prob_rpt(error_text)
char error_text[];
{
 fprintf(stderr,"        %s\n",error_text);
} /* end prob_rpt */


void nrerror(error_text)
char error_text[];
{
        void exit();

        fprintf(stderr,"Numerical Recipes run-time error...\n");
        fprintf(stderr,"%s\n",error_text);
        fprintf(stderr,"...now exiting to system...\n");
        exit(1);
}


float *vector(nl,nh)
int nl,nh;
{
        float *v;

        v=(float *)malloc((unsigned) (nh-nl+1)*sizeof(float));
        if (!v) nrerror("allocation failure in vector()");
        return v-nl;
```

```
}

char *cvector(nl,nh)
int nl,nh;
{
	char *v;

	v=(char *)malloc((unsigned) (nh-nl+1)*sizeof(char));
	if (!v) nrerror("allocation failure in cvector()");
	return v-nl;
}

int *ivector(nl,nh)
int nl,nh;
{
	int *v;

	v=(int *)malloc((unsigned) (nh-nl+1)*sizeof(int));
	if (!v) nrerror("allocation failure in ivector()");
	return v-nl;
}

double *dvector(nl,nh)
int nl,nh;
{
	double *v;

	v=(double *)malloc((unsigned) (nh-nl+1)*sizeof(double));
	if (!v) nrerror("allocation failure in dvector()");
	return v-nl;
}


float **matrix(nrl,nrh,ncl,nch)
int nrl,nrh,ncl,nch;
{
	int i;
	float **m;

	m=(float **) malloc((unsigned) (nrh-nrl+1)*sizeof(float*));
	if (!m) nrerror("allocation failure 1 in matrix()");
	m -= nrl;

	for(i=nrl;i<=nrh;i++) {
		m[i]=(float *) malloc((unsigned) (nch-ncl+1)*sizeof(float));
		if (!m[i]) nrerror("allocation failure 2 in matrix()");
		m[i] -= ncl;
	}
```

```
	return m;
}

double **dmatrix(nrl,nrh,ncl,nch)
int nrl,nrh,ncl,nch;
{
	int i;
	double **m;

	m=(double **) malloc((unsigned) (nrh-nrl+1)*sizeof(double*));
	if (!m) nrerror("allocation failure 1 in dmatrix()");
	m -= nrl;

	for(i=nrl;i<=nrh;i++) {
		m[i]=(double *) malloc((unsigned) (nch-ncl+1)*sizeof(double));
		if (!m[i]) nrerror("allocation failure 2 in dmatrix()");
		m[i] -= ncl;
	}
	return m;
}

int **imatrix(nrl,nrh,ncl,nch)
int nrl,nrh,ncl,nch;
{
	int i,**m;

	m=(int **)malloc((unsigned) (nrh-nrl+1)*sizeof(int*));
	if (!m) nrerror("allocation failure 1 in imatrix()");
	m -= nrl;

	for(i=nrl;i<=nrh;i++) {
		m[i]=(int *)malloc((unsigned) (nch-ncl+1)*sizeof(int));
		if (!m[i]) nrerror("allocation failure 2 in imatrix()");
		m[i] -= ncl;
	}
	return m;
}
```

```
float **submatrix(a,oldrl,oldrh,oldcl,oldch,newrl,newcl)
float **a;
int oldrl,oldrh,oldcl,oldch,newrl,newcl;
{
	int i,j;
	float **m;

	m=(float **) malloc((unsigned) (oldrh-oldrl+1)*sizeof(float*));
	if (!m) nrerror("allocation failure in submatrix()");
```

```
	m -= newrl;

	for(i=oldrl,j=newrl;i<=oldrh;i++,j++) m[j]=a[i]+oldcl-newcl;

	return m;
}



int free_vector(v,nl,nh)
float *v;
int nl,nh;
{
	return(free((char*) (v+nl)));
}

void free_ivector(v,nl,nh)
int *v,nl,nh;
{
	free((char*) (v+nl));
}

void free_dvector(v,nl,nh)
double *v;
int nl,nh;
{
	free((char*) (v+nl));
}



void free_matrix(m,nrl,nrh,ncl,nch)
float **m;
int nrl,nrh,ncl,nch;
{
	int i;

	for(i=nrh;i>=nrl;i--) free((char*) (m[i]+ncl));
	free((char*) (m+nrl));
}

void free_dmatrix(m,nrl,nrh,ncl,nch)
double **m;
int nrl,nrh,ncl,nch;
{
	int i;

	for(i=nrh;i>=nrl;i--) free((char*) (m[i]+ncl));
	free((char*) (m+nrl));
```

```
}

void free_imatrix(m,nrl,nrh,ncl,nch)
int **m;
int nrl,nrh,ncl,nch;
{
        int i;

        for(i=nrh;i>=nrl;i--) free((char*) (m[i]+ncl));
        free((char*) (m+nrl));
}


void free_submatrix(b,nrl,nrh,ncl,nch)
float **b;
int nrl,nrh,ncl,nch;
{
        free((char*) (b+nrl));
}


float **convert_matrix(a,nrl,nrh,ncl,nch)
float *a;
int nrl,nrh,ncl,nch;
{
        int i,j,nrow,ncol;
        float **m;

        nrow=nrh-nrl+1;
        ncol=nch-ncl+1;
        m = (float **) malloc((unsigned) (nrow)*sizeof(float*));
        if (!m) nrerror("allocation failure in convert_matrix()");
        m -= nrl;
        for(i=0,j=nrl;i<=nrow-1;i++,j++) m[j]=a+ncol*i-ncl;
        return m;
}

void free_convert_matrix(b,nrl,nrh,ncl,nch)
float **b;
int nrl,nrh,ncl,nch;
{
        free((char*) (b+nrl));
}


char *sdate()
{
```

```
    FILE *datepipe;
    char datestring[30];

    datepipe = popen("date","r");
    fgets(datestring,30,datepipe);
    if((pclose(datepipe))!=0)
          prob_rpt("\n***Problem in closing date pipe.***\n");
    return(datestring);
}

float arctan2(y,x)
float y,x;

{
    extern int signbit();
    extern double atan2();

    float answer;
    answer = (x!=0) ? atan2(y,x) : Pi/2.*signbit(y);
    return(answer);
}

float Log(x)
float x;

{
    float answer;
    answer = (x>0.) ? (float)log((double)x) : -BIG;
    printf("Ans = %f\n",answer);
    return(answer);
}
```

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "smudge.h"
#define Pi 3.14159265359

/* global variables */
extern float *xm,*ym;
extern char destdir[80];

void output(numbe,npoint,grid,oe,be)
struct griddata grid;
struct boundpoints be;
struct specpoints oe;
int numbe,npoint;
{
  extern double cos(),sin();
  int i,m;
  FILE *plotxypipe,*data;
  char filename[100];
  float phi,xi;

 /*
   print boundary displacements and stresses.
 */
   sprintf(filename,"%s/Bdry_Elements",destdir);
   fprintf(stderr,"\tFile: Bdry_Elements\n",destdir);
   data = fopen(filename,"w");
   fprintf(data,
     "\nDisplacements and Stresses at Midpoints of Boundary Elements.\n");
   fprintf(data,
     "\n Element       US          UN          UX          UY        Sigma-S     Sigma-N\n");
   fprintf(data,
     "---------  ---------  ---------  ---------  ---------  ---------  ---------\n");
   for(i=1;i<=2*numbe;i++)
     fprintf(data,"%5d      %9.6f  %9.6f  %9.6f  %9.6f  %9.6f  %9.6f\n",
            i,be.us[i],be.un[i],be.ux[i],be.uy[i],be.sigs[i],be.sign[i]);
   if((fclose(data))!=0)
     prob_rpt("\n***Problem in closing 'Bdry_Elements' file.***\n");

 /*
   print other element displacements and stresses
 */
   sprintf(filename,"%s/Spec_Elements",destdir);
   fprintf(stderr,"\tFile: Spec_Elements\n",destdir);
   data = fopen(filename,"w");
   fprintf(data,
     "\nDisplacements and Stresses at Specified Points in the Body.\n");
   fprintf(data,
```

```
    "\n  Point      X Coord     Y Coord       UX          UY         Sigma-XX    Sigma-YY    Sigma-XY\n"
  fprintf(data,
    "---------  ---------  ---------  ---------  ---------  ---------  ---------  ---------\n");
  for(i=1;i<=npoint;i++)
    fprintf(data,
      "%5d        %6.2f    %6.2f      %9.6f  %9.6f  %7.2f     %7.2f     %7.2f\n",
      i,oe.xp[i],oe.yp[i],oe.ux[i],oe.uy[i],
        oe.sigxx[i],oe.sigyy[i],oe.sigxy[i]);
  if((fclose(data))!=0)
    prob_rpt("\n***Problem in closing 'Spec_Elements' file.***\n");

/*
  plot the boundary element and specified point dispacements
*/
  sprintf(filename,"%s/Disp.gph",destdir);
  fprintf(stderr,"\tFile: Disp.gph\n",destdir);
  data = fopen(filename,"w");
  fprintf(data,"%d boundary elements.\n",2*numbe);
  fprintf(data,"%d specified points.\n",npoint);
  for(m=1;m<=numbe;m++) {
    fprintf(data,"%f %f %f %f\n",xm[m],ym[m],be.ux[2*m-1],be.uy[2*m-1]);
    fprintf(data,"%f %f %f %f\n",xm[m],ym[m],be.ux[2*m],be.uy[2*m]);
  }
  for(m=1;m<=npoint;m++) fprintf(data,"%f %f %f %f\n",
    oe.xp[m],oe.yp[m],oe.ux[m],oe.uy[m]);
  if((fclose(data))!=0)
    prob_rpt("\n***Problem in closing 'Disp.gph' file.***\n");

/*
  print the boundary element and specified point stress tensor
*/
/*
  sprintf(filename,"%s/Stress.gph",destdir);
  fprintf(stderr,"\tFile: Stress.gph\n",destdir);
  data = fopen(filename,"w");
  fprintf(data,"%d specified points.\n",npoint);
  for(m=1;m<=npoint;m++) fprintf(data,"%f %f %f %f %f\n",
    oe.xp[m],oe.yp[m],oe.sigxx[m],oe.sigyy[m],oe.sigxy[m]);
  if((fclose(data))!=0)
    prob_rpt("\n***Problem in closing 'Stress.gph' file.***\n");
*/

/*
  print the specified point principal stresses into a usable file
  for use with Spyglass Transform
*/
  sprintf(filename,"%s/Sigma_Spy",destdir);
  fprintf(stderr,"\tFile: Sigma_Spy\n",destdir);
  data = fopen(filename,"w");
```

```
fprintf(data,"%d specified points.\n",npoint);
fprintf(data,"%d boundary elements.\n",2*numbe);
for(m=1;m<=npoint;m++) {
   phi = oe.theta1[m]*180./Pi;
   xi = oe.theta3[m]*180./Pi;
   fprintf(data,"%f %f %f %f %f %f\n",
      oe.xp[m],oe.yp[m],oe.sigma1[m],oe.sigma3[m],phi,xi);
}
for(m=1;m<=numbe;m++) {
   phi = be.theta[m]*180./Pi;
   if(phi>90.) {
        while(phi>90.) phi −= 180.;
   } else {
        while(phi<−90.) phi += 180.;
   }
   xi = be.theta[m]*180./Pi+90;
   if(xi>90.) {
        while(xi>90.) xi −= 180.;
   } else {
        while(xi<−90.) xi += 180.;
   }
   fprintf(data,"%f %f %f %f %f %f\n",
    xm[m],ym[m],be.sigma1[m],be.sigma3[m],phi,xi);
}
fprintf(data,"XPos Ypos Sigma1 Sigma3 Sig1dir Sig3dir\n");
if((fclose(data))!=0)
   prob_rpt("\n***Problem in closing 'Sigma_Spy' file.***\n");

/* create an orientation vector quick plot using plotxy */
sprintf(filename,"%s/plotxy.dir3",destdir);
fprintf(stderr,"\tFile: plotxy.dir3\n",destdir);
plotxypipe = fopen(filename,"w");
fprintf(plotxypipe,"file *\n");
fprintf(plotxypipe,"output Sigma1dir.ps\n");
fprintf(plotxypipe,"mode 2\n");
fprintf(plotxypipe,"frame none\n");
fprintf(plotxypipe,"color 0\n");
fprintf(plotxypipe,"xlim 5.650 %f %f\n",
   grid.xorigin,grid.xorigin+grid.length);
fprintf(plotxypipe,"ylim 5.698 %f %f\n",
   grid.yorigin,grid.yorigin+grid.length);
fprintf(plotxypipe,"read %d\n",npoint*3);
/* check the trig out */
for(m=1;m<=npoint;m++) fprintf(plotxypipe,"%f %f\n%f %f\n999999. 999999.\n",
   oe.xp[m]+sin(oe.theta3[m]),oe.yp[m]−cos(oe.theta3[m]),
   oe.xp[m]−sin(oe.theta3[m]),oe.yp[m]+cos(oe.theta3[m]));
fprintf(plotxypipe,"plot 1.529 4.513\n");
fprintf(plotxypipe,"stop\n");
if((fclose(plotxypipe))!=0)
```

```
    prob_rpt("\n***Problem in closing 'plotxy.dir2' file.***\n");

/* create the orientation vector quick plot using plotxy */
  sprintf(filename,"%s/plotxy.dir1",destdir);
  fprintf(stderr,"\tFile: plotxy.dir1\n",destdir);
  plotxypipe = fopen(filename,"w");
  fprintf(plotxypipe,"file *\n");
  fprintf(plotxypipe,"output Sigma3dir.ps\n");
  fprintf(plotxypipe,"mode 2\n");
  fprintf(plotxypipe,"frame none\n");
  fprintf(plotxypipe,"color 0\n");
  fprintf(plotxypipe,"xlim 5.650 %f %f\n",
    grid.xorigin,grid.xorigin+(float)grid.length);
  fprintf(plotxypipe,"ylim 5.698 %f %f\n",
    grid.yorigin,grid.yorigin+(float)grid.length);
  fprintf(plotxypipe,"read %d\n",npoint*3);
  for(m=1;m<=npoint;m++) fprintf(plotxypipe,"%f %f\n%f %f\n999999. 999999.\n",
    oe.xp[m]-sin(oe.theta1[m]),oe.yp[m]-cos(oe.theta1[m]),
    oe.xp[m]+sin(oe.theta1[m]),oe.yp[m]+cos(oe.theta1[m]));
  fprintf(plotxypipe,"plot 1.529 4.513\n");
  fprintf(plotxypipe,"stop\n");
  if((fclose(plotxypipe))!=0)
    prob_rpt("\n***Problem in closing 'plotxy.dir1' file.***\n");

} /* end output() */
```

```
#include <math.h>
#include "smudge.h"
#include <stdio.h>
#define Pi 3.14159265359

/* global variables */
extern float *angle;

/* calculate the principle stresses */
int prcpl_stress(numbe,npoint,oe,be)
int numbe,npoint;
struct specpoints oe;
struct boundpoints be;
{
  float sigmabarn,sigmasmax,u1,u2,uu;
  int i;
  extern float arctan2();

/* calculate the stress for the boundary elements */
  for(i=1;i<=numbe;i++) {
    be.theta[i] = 0.5*arctan2(0.,be.sign[i]-be.sigs[i])-angle[i]/180.*Pi;
    if(be.sign[i]<be.sigs[i]) {
        be.sigma1[i] = be.sign[i];
        be.sigma3[i] = be.sigs[i];
    } else {
        be.sigma1[i] = be.sigs[i];
        be.sigma3[i] = be.sign[i];
    } /* end of if statement */
  } /* end of i loop */

/* calculate the stress for the specified points by a SVD */
  for(i=1;i<=npoint;i++) {
    sigmabarn = 0.5*(oe.sigxx[i] + oe.sigyy[i]);
    sigmasmax = 0.5*sqrt((oe.sigxx[i]+oe.sigyy[i])*(oe.sigxx[i]+oe.sigyy[i])+
        4.*(oe.sigxy[i]*oe.sigxy[i]-oe.sigxx[i]*oe.sigyy[i]));
    oe.sigma1[i] = sigmabarn - sigmasmax; /* most compressive stress */
    oe.sigma3[i] = sigmabarn + sigmasmax; /* least compressive stress */

    u1 = 1.;
    u2 = (oe.sigma1[i]-oe.sigxx[i])/oe.sigxy[i];
    uu = sqrt(u1*u1+u2*u2);
    u1 = u1/uu;
    u2 = u2/uu;
    oe.theta1[i] = arctan2(u2,u1);

    u1 = 1.;
    u2 = (oe.sigma3[i]-oe.sigxx[i])/oe.sigxy[i];
    uu = sqrt(u1*u1+u2*u2);
    u1 = u1/uu;
```

```
        u2 = u2/uu;
        oe.theta3[i] = arctan2(u2,u1);

        if ((oe.sigma1[i] <= 1.05*oe.sigma3[i])&&
                (oe.sigma1[i] >= 0.95*oe.sigma3[i])) {
            oe.theta1[i] = 666./180.*Pi;
            oe.theta3[i] = oe.theta1[i];
                printf("%f %f\n",oe.sigma1[i],oe.sigma3[i]);
        }
    } /* end of i loop */
} /* end of prcpl_strees() */
```

(print_input.c)

```c
#include <stdio.h>
#include <math.h>
#include <string.h>

/* global variables */
extern char *somig();
extern float *sinbet,*cosbet,*xm,*ym,*a,*r,*angle;
extern char destdir[80];

void print_input(numbe)

int numbe;
{
  int m;
  float size;
  FILE *data;
  char filename[100];
  extern void prob_rpt();

/* open the Bdry_input filename */
  sprintf(filename,"%s/Bdry_input",destdir);
  data = fopen(filename,"w");


/* output information about the data */
  fprintf(data,"\nBoundary Element Data\n");
  fprintf(data,"\n              Center     Center\n");
  fprintf(data," Element      X          Y         Length");
  fprintf(data,"      Angle          Sigma-S\n");
  fprintf(data,"-------   -------- ---------- -------- ");
  fprintf(data,"----------- -----------\n");

  for(m=1;m<=numbe;m++) {
    size = 2.0 * a[m];
    if(cosbet[m]!=0.0&&sinbet[m]!=0.0) {
      angle[m] = 180.0*atan2(sinbet[m],cosbet[m])/3.14159265359;
    } else {
      angle[m] = 0.;
    } /* end if */
    fprintf(data,"%5d+      %4.4f    %4.4f    %4.4f    %3.3f     %9.6f\n",
        m,xm[m],ym[m],size,angle[m],r[2*m-1]);
    fprintf(data,"%5d-      %4.4f    %4.4f    %4.4f    %3.3f     %9.6f\n",
        m,xm[m],ym[m],size,180.+angle[m],r[2*m]);
  } /* end m loop */

 /* close the file */
   if((fclose(data))!=0)
         prob_rpt("\n***Problem in closing 'Bdry_input' file.***\n");
```

```
} /* end print_input() */
```

```
float* matXvect(mat,row,col,vect)
/* matrix multiplied by a vector */

float **mat,*vect;
int row,col;

{
  int i,j;
  float *ans,*vector();

  ans = vector(0,row+1);
  for(i=1;i<=row;i++) {
    ans[i] = 0.0;
    for(j=1;j<=col;j++) ans[i] += mat[i][j]*vect[j];
  } /* end i loop */
  return(ans);
} /* end matXvect() */

float* vectXmat(mat,row,col,vect)
/* vector multiplied by a matrix */

float **mat,*vect;
int row,col;

{
  int i,j;
  float *ans,*vector();

  ans = vector(0,row+1);
  for(i=1;i<=row;i++) {
    ans[i] = 0.0;
    for(j=1;j<=col;j++) ans[i] += vect[j]*mat[j][i];
  } /* end i loop */
  return(ans);
} /* end vectXmat() */

float** matXmat(mat1,mat2,row,col,depth)
/* matrix multiplied by a matrix */

float **mat1,**mat2;
int row,col;

{
  int i,j,k;
  float **ans,**matrix();

  ans = matrix(0,row+1,0,depth+1);
  for(i=1;i<=row;i++) {
    for(k=1;k<=depth;k++) {
```

```
          ans[i][k] = 0.0;
          for(j=1;j<=col;j++) ans[i][k] += mat1[i][j]*mat2[j][k];
       } /* end k loop */
    } /* end i loop */
    return(ans);
} /* end matXmat() */

float** transpose(mat,row,col)
/* matrix transpose */

float **mat;
int row,col;

{
    int i,j;
    float **ans,**matrix();

    ans = matrix(0,col+1,0,row+1);
    for(i=1;i<=row;i++) {
       for(j=1;j<=col;j++) {
            ans[j][i] = mat[i][j];
       } /* end j loop */
    } /* end i loop */
    return(ans);
} /* end transpose() */
```

```
/* Solve the system (Y = CX) using LU Decomposition.   */
#undef DEBUG

#ifdef DEBUG
#include <stdio.h>
#endif

/* global variables */
extern float *X,*Y,**C;

void solve_lu(numbe)

int numbe;
{
  float **Cinv,*col,d,*vector(),**matrix(),*matXvect();
  int *indx,i,j,*ivector(),n;

/* we have a 4*numbe system of equations */
  n = 4*numbe;

#ifdef DEBUG
/* Print out C matrix when debugging. */
  printf("\nC Matrix\n");
  for(i=1;i<=n;i++) {
    for(j=1;j<=n;j++) printf("%f ",C[i][j]);
    printf("\n");
  }
/* Print out Y vector when debugging. */
  printf("\nY Vector\n");
  for(i=1;i<=n;i++) printf("%f\n",Y[i]);
  fflush(stdout);
#endif

/* allocate memory for tempory variables */
  Cinv = matrix(0,n+1,0,n+1);
  indx = ivector(0,n+1);
  col = vector(0,n+1);

/* perform the LU-Decomposition */
  ludcmp(C,n,indx,&d);
  for(j=1;j<=n;j++) {
    for(i=1;i<=n;i++) col[i] = 0.0;
    col[j] = 1.0;
    lubksb(C,n,indx,col);
    for(i=1;i<=n;i++) Cinv[i][j] = col[i];
  }
  X = matXvect(Cinv,n,n,Y);

/* release memory for the temporary variables */
```

```
   free_matrix(Cinv,0,n+1,0,n+1);
   free_vector(col,0,n+1);
   free_ivector(indx,0,n+1);

#ifdef DEBUG
/* Print out X vector when debugging. */
   printf("\nX Vector\n");
   for(i=1;i<=n;i++) printf("%f\n",X[i]);
   fflush(stdout);
#endif
} /* end solve_lu() */
```

```
#include <math.h>
static float sqrarg;
#define SQR(a) (sqrarg=(a),sqrarg*sqrarg)
#define Pi 3.14159265359
#undef TEST

/* global variables */
extern float *S,*U;
extern float pr1,pr2,pr3,con;

void somig(x,y,xj,yj,a,cosb,sinb)

float x,y,xj,yj,a,cosb,sinb;

{
  float cxb,cxbpa,cxbma,cyb,cosb2,sinb2,cos2b,sin2b,r1s,r2s,fl1,fl2,tb[8],
         uxust,uxunt,uxsst,uxsnt,uyust,uyunt,uysst,uysnt,
           sxxust,sxxunt,sxxsst,sxxsnt,syyust,syyunt,syysst,syysnt,
           sxyust,sxyunt,sxysst,sxysnt;

/* initialize local variables */
  cxb = (x-xj)*cosb+(y-yj)*sinb;
  cxbpa = cxb+a;
  cxbma = cxb-a;
  cyb = -(x-xj)*sinb+(y-yj)*cosb;
  cosb2 = SQR(cosb);
  sinb2 = SQR(sinb);
  cos2b = cosb2-sinb2;
  sin2b = 2.0*cosb*sinb;
  r1s = SQR(cxbma)+SQR(cyb);
  r2s = SQR(cxbpa)+SQR(cyb);
  fl1 = r1s>0.0 ? 0.5*log((double)r1s) : 0.0 ;
  fl2 = r2s>0.0 ? 0.5*log((double)r2s) : 0.0 ;
  /*
  fl1 = 0.5*Log((double)r1s);
  fl2 = 0.5*Log((double)r2s);
  */
  tb[2] = -con*(fl1-fl2);
  tb[3] = (cyb==0.0) ? ((fabs((double)cxb)<a) ? con*Pi : 0.0 ) :
           con*(atan(cxbpa/cyb)-atan(cxbma/cyb));
 #ifdef TEST
  tb[1] = -cyb*tb[3]+con*(cxbma*fl1-cxbpa*fl2)+2*a*con;
 #else
  tb[1] = -cyb*tb[3]+con*(cxbma*fl1-cxbpa*fl2);
 #endif
  tb[4] = con*(cyb/r1s-cyb/r2s);
  tb[5] = con*(cxbma/r1s-cxbpa/r2s);
  tb[6] = -con*((SQR(cxbma)-SQR(cyb))/SQR(r1s)-(SQR(cxbpa)-SQR(cyb))/SQR(r2s));
  tb[7] = -con*2.0*cyb*(cxbma/SQR(r1s)-cxbpa/SQR(r2s));
```

```
    uxust = pr1*sinb*tb[2]-pr2*cosb*tb[3]+cyb*(sinb*tb[4]-cosb*tb[5]);
    uxunt = pr1*cosb*tb[2]+pr2*sinb*tb[3]-cyb*(cosb*tb[4]+sinb*tb[5]);
#ifdef TEST
    uxsst = pr3*cosb*tb[1]-cyb*(sinb*tb[2]+cosb*tb[3])+2*con*a*cosb;
#else
    uxsst = pr3*cosb*tb[1]-cyb*(sinb*tb[2]+cosb*tb[3]);
#endif
    uxsnt = -pr3*sinb*tb[1]+cyb*(cosb*tb[2]-sinb*tb[3]);
    uyust = -pr1*cosb*tb[2]-pr2*sinb*tb[3]-cyb*(cosb*tb[4]+sinb*tb[5]);
    uyunt = pr1*sinb*tb[2]-pr2*cosb*tb[3]-cyb*(sinb*tb[4]-cosb*tb[5]);
#ifdef TEST
    uysst = pr3*sinb*tb[1]+cyb*(cosb*tb[2]-sinb*tb[3])+2*con*a*cosb;
#else
    uysst = pr3*sinb*tb[1]+cyb*(cosb*tb[2]-sinb*tb[3]);
#endif
    uysnt = pr3*cosb*tb[1]+cyb*(sinb*tb[2]+cosb*tb[3]);

    U[0] = uxust;
    U[1] = uxunt;
    U[2] = uxsst;
    U[3] = uxsnt;
    U[4] = uyust;
    U[5] = uyunt;
    U[6] = uysst;
    U[7] = uysnt;

    sxxust = 2.0*cosb2*tb[4]+sin2b*tb[5]-cyb*(cos2b*tb[6]-sin2b*tb[7]);
    sxxunt = -tb[5]-cyb*(sin2b*tb[6]+cos2b*tb[7]);
    sxxsst = -tb[2]-pr2*(cos2b*tb[2]-sin2b*tb[3])+cyb*(cos2b*tb[4]+sin2b*tb[5]);
    sxxsnt = -tb[3]+pr1*(sin2b*tb[2]+cos2b*tb[3])+cyb*(sin2b*tb[4]-cos2b*tb[5]);

    syyust = 2.0*sinb2*tb[4]-sin2b*tb[5]+cyb*(cos2b*tb[6]-sin2b*tb[7]);
    syyunt = -tb[5]+cyb*(sin2b*tb[6]+cos2b*tb[7]);
    syysst = -tb[2]+pr2*(cos2b*tb[2]-sin2b*tb[3])-cyb*(cos2b*tb[4]+sin2b*tb[5]);
    syysnt = -tb[3]-pr1*(sin2b*tb[2]+cos2b*tb[3])-cyb*(sin2b*tb[4]-cos2b*tb[5]);

    sxyust = sin2b*tb[4]-cos2b*tb[5]-cyb*(sin2b*tb[6]+cos2b*tb[7]);
    sxyunt = cyb*(cos2b*tb[6]-sin2b*tb[7]);
    sxysst = -pr2*(sin2b*tb[2]+cos2b*tb[3])+cyb*(sin2b*tb[4]-cos2b*tb[5]);
    sxysnt = -pr1*(cos2b*tb[2]-sin2b*tb[3])-cyb*(cos2b*tb[4]+sin2b*tb[5]);

/* SigmaXX */
    S[0] = sxxust;
    S[1] = sxxunt;
    S[2] = sxxsst;
    S[3] = sxxsnt;

/* SigmaYY */
```

```
    S[4]  = syyust;
    S[5]  = syyunt;
    S[6]  = syysst;
    S[7]  = syysnt;

/* SigmaXY */
    S[8]  = sxyust;
    S[9]  = sxyunt;
    S[10] = sxysst;
    S[11] = sxysnt;

} /* end of somig() */
```

```c
/* Input the specified point information */
#include <stdio.h>
#include <string.h>
#include "smudge.h"

/* global variables */
extern char oe_file[30];
extern float *xbeg,*ybeg,*xend,*yend;

int specin(numpb,numos)
int *numpb,numos;

{
  FILE *data;
  int n,memsum=0;

/* Open data file.  If file does not exist report error and exit. */
  data = fopen(oe_file,"r");
  if(data==NULL) error_rpt("Error in opening up other element data file.");

/* read in the specified point grid */
  for(n=1;n<=numos;n++) {
    fscanf(data,"%f%f%f%f%d",&xbeg[n],&ybeg[n],&xend[n],&yend[n],&numpb[n]);
    memsum += numpb[n];
    memsum += 2;
  }

/* close the input file */
  if((fclose(data))!=0)
      prob_rpt("\n***Problem in closing other element input file.***\n");

/* return the ammount of memory needed for the specified elements */
  return(memsum);
}
```

(specpt.c)

```
#include <math.h>
#include "smudge.h"
static float sqrarg;
#define SQR(a) (sqrarg=(a),sqrarg*sqrarg)

/* global variables */
extern void somig();
extern float g,*xbeg,*ybeg,*xend,*yend,*sinbet,*cosbet,*xm,*ym,*a,*r,
  *X,*P,*A,*B,*U,*S;

/* compute displacements and stresses at specified points in the body. */
int specpt(npoint,numbe,oe,numpb,numos,noignore)

int *npoint,numbe,*numpb,numos,noignore;
struct specpoints oe;
{
  float aj,xj,yj,cosbj,sinbj,delx,dely,usj,unj,ssj,snj,xp,yp;
  int nump,ni,n,j,ignore=0,ja,jb,jc,jd;

  *npoint = 1;
  for(n=1;n<=numos;n++) {
    nump=numpb[n]+1;

    /* calculate boundary element x and y lengths */
    delx = (xend[n]-xbeg[n])/nump;
    dely = (yend[n]-ybeg[n])/nump;

    /* adjust the number of points for some reason*/
    if(numpb[n]>0) nump++;

    /* if points are coincident, don't be redundant */
    if(SQR(delx)+SQR(dely)==0.0) nump = 1;

    for(ni=1;ni<=nump;ni++) { /* loop over all of the specified points */
      xp = xbeg[n]+(float)(ni-1)*delx;
      yp = ybeg[n]+(float)(ni-1)*dely;
      oe.xp[*npoint] = xp;         /* x location of the specified point */
      oe.yp[*npoint] = yp;         /* y location of the specified point */
      oe.ux[*npoint] = 0.0;        /* x displacement at the specified point */
      oe.uy[*npoint] = 0.0;        /* y displacement at the specified point */
      oe.sigxx[*npoint] = P[0];    /* Regional Sigma_XX */
      oe.sigxy[*npoint] = P[1];    /* Regional Sigma_XY */
      oe.sigyy[*npoint] = P[2];    /* Regional Sigma_YY */

      for(j=1;j<=numbe;j++){  /* loop over all of the elements */
        /* location of middle point of jth boundary element */
        xj = xm[j];
        yj = ym[j];
        aj = a[j];
```

```
      /* if a specified point coincides with a boundary element, ignore it
         unless noignore is true */
      if(((SQR(xp-xj)+SQR(yp-yj))>=4.0*aj*aj)||noignore) {
        jd = 4*j;    /* U_n(+)       */
        jc = jd-1;   /* U_s(-)       */
        jb = jc-1;   /* U_s(+)       */
        ja = jb-1;   /* Sigma_N(+) */
      /* do calculation for positive side of couple element */
        cosbj = cosbet[j];
        sinbj = sinbet[j];
        somig(xp,yp,xj,yj,aj,cosbj,sinbj);
        snj = X[ja];  /* Sigma_N(+) */
        usj = X[jb];  /* U_S(+) */
        unj = X[jd];  /* U_N(+) */
        ssj = r[2*j-1]; /* Specified Strength */

        oe.ux[*npoint] -= U[0]*usj+U[1]*unj+U[2]*ssj+U[3]*snj;
        oe.uy[*npoint] -= U[4]*usj+U[5]*unj+U[6]*ssj+U[7]*snj;
        oe.sigxx[*npoint] -= 2.0*g*(S[0]*usj+S[1]*unj+S[2]*ssj+S[3]*snj);
        oe.sigyy[*npoint] -= 2.0*g*(S[4]*usj+S[5]*unj+S[6]*ssj+S[7]*snj);
        oe.sigxy[*npoint] -= 2.0*g*(S[8]*usj+S[9]*unj+S[10]*ssj+S[11]*snj);

      /* do calculation for negative side of couple element */
        cosbj = -cosbet[j];
        sinbj = -sinbet[j];
        somig(xp,yp,xj,yj,aj,cosbj,sinbj);
        snj = X[ja];    /* Sigma_N(-) */
        usj = X[jc];    /* U_S(-) */
        unj = -X[jd];   /* U_N(-) */
        ssj = r[2*j];   /* Specified Strength */

        oe.ux[*npoint] -= U[0]*usj+U[1]*unj+U[2]*ssj+U[3]*snj;
        oe.uy[*npoint] -= U[4]*usj+U[5]*unj+U[6]*ssj+U[7]*snj;
        oe.sigxx[*npoint] -= 2.0*g*(S[0]*usj+S[1]*unj+S[2]*ssj+S[3]*snj);
        oe.sigyy[*npoint] -= 2.0*g*(S[4]*usj+S[5]*unj+S[6]*ssj+S[7]*snj);
        oe.sigxy[*npoint] -= 2.0*g*(S[8]*usj+S[9]*unj+S[10]*ssj+S[11]*snj);

      } else {
      /* ignore this element as it coincides with a boundary element */
        ignore++;
        *npoint-=1;
        break;
      } /* end if statement */
    } /* end j loop */
    *npoint+=1; /* increase the number of elements for next value of loop*/
  } /* end ni loop */
} /* end n loop */
*npoint-=1; /* correct for element increase at end of n loop */
```

```
    return(ignore); /* the number of points ignored */
} /* end of specpt() */
```

```
#include <stdio.h>
#include <math.h>
main()
{

   float **mat1,**mat2,**mat3,**ans1,**ans2;
   float xp,yp,ux,uy,rotation,sigxx,sigxy,sigyy,regionalxx,regionalyy,
      regionalxy,angle;
   FILE *input,*output;
   char temp[120],outfile[40];
   int i,point;
   extern float** matrix(),**matXmat();

/* allocate space for variables */
   mat1 = matrix(0,3,0,3);
   mat2 = matrix(0,3,0,3);
   mat3 = matrix(0,3,0,3);
   ans1 = matrix(0,3,0,3);
   ans2 = matrix(0,3,0,3);

/* Enter the output filename */
   fprintf(stderr,"Enter the output filename: ");
   scanf("%s",outfile);

/* Enter the Regional Stress Tensor */
   fprintf(stderr,"Enter SigmaXX: ");
   scanf("%f",&regionalxx);
   fprintf(stderr,"Enter SigmaYY: ");
   scanf("%f",&regionalyy);
   fprintf(stderr,"Enter SigmaXY: ");
   scanf("%f",&regionalxy);

/* Enter the plane along which the stress tensor is calculated */
   fprintf(stderr,"Enter the orientation of the plane: ");
   scanf("%f",&angle);

/* set up rotation matrix */
   rotation = 3.14159265*angle/180.;
   mat1[1][1] = cos(rotation);
   mat1[1][2] = sin(rotation);
   mat1[2][1] = -sin(rotation);
   mat1[2][2] = cos(rotation);

/* set up rotation matrix transpose */
   mat3[1][1] = mat1[1][1];
   mat3[1][2] = mat1[2][1];
   mat3[2][1] = mat1[1][2];
   mat3[2][2] = mat1[2][2];
```

```
/* read in the header stuff */
  input = fopen("Spec_Elements","r");
  fgets(temp,120,input);
  fgets(temp,120,input);
  fgets(temp,120,input);
  fgets(temp,120,input);
  fgets(temp,120,input);

/* open output file */
  output = fopen(outfile,"w");

/* read in tensors and remove specified regional tensor */
  while(fscanf(input,"%d%f%f%f%f%f%f%f",&point,&xp,&yp,&ux,&uy,
     &sigxx,&sigyy,&sigxy)!=EOF) {
/* load in the stress tensor */
     mat2[1][1] = sigxx-regionalxx;
     mat2[1][2] = sigxy-regionalxy;
     mat2[2][1] = sigxy-regionalxy;
     mat2[2][2] = sigyy-regionalyy;

/* rotate that tensor */
     ans1 = matXmat(mat1,mat2,2,2,2);
     ans2 = matXmat(ans1,mat3,2,2,2);

/* output it to file */
     fprintf(output,"%f %f %f %f %f %f\n",xp,yp,ans2[1][1],ans2[1][2],
          ans2[2][1],ans2[2][2]);
   } /* end of while loop */

/* close the output file */
   if((fclose(output))!=0) fprintf(stderr,"Error closing %s.",outfile);
   if((fclose(input))!=0) fprintf(stderr,"Error closing Spec_Elements.");
} /* end of stressrot() */
```

```
#include <stdio.h>
#include <math.h>

void uplot(type,xmag,ymag,plotname,x0,x1,y0,y1,dec)
int type,dec;
float xmag,ymag,x0,x1,y0,y1;
char plotname[30];
{
  FILE *xypipe,*input;
  int numbe,numoe,i,count=0,benum,oenum;
  float *bxp,*byp,*bux,*buy;
  float *oxp,*oyp,*oux,*ouy;
  char junk[20],plotprog[80];
  extern float *vector();

/* open the input file */
  input = fopen("Disp.gph","r");
  fscanf(input,"%d",&numbe);
  fgets(junk,20,input);
  fscanf(input,"%d",&numoe);
  fgets(junk,20,input);

/* allocate memory for the boundary elements */
  bxp = vector(0,numbe);
  byp = vector(0,numbe);
  bux = vector(0,numbe);
  buy = vector(0,numbe);

/* allocate memory for the specified points */
  oxp = vector(0,numoe);
  oyp = vector(0,numoe);
  oux = vector(0,numoe);
  ouy = vector(0,numoe);

/* read in the boundary element data */
  benum = 1;
  count = 1;
  for(i=1;i<=numbe;i++) {
    fscanf(input,"%f%f%f%f",&bxp[benum],&byp[benum],&bux[benum],&buy[benum]);
    if(count++==dec) {
        benum++;
        count=1;
    }
  }
  benum--;

/* read in the specified point data */
  oenum = 1;
  count = 1;
```

```
for(i=1;i<=numoe;i++) {
  fscanf(input,"%f%f%f%f",&oxp[oenum],&oyp[oenum],&oux[oenum],&ouy[oenum]);
  if(count++==dec) {
      oenum++;
      count=1;
  }
}
oenum--;
fclose(input);

/* open a pipe to plotxy */
sprintf(plotprog,"/home/griffy/purbut/bin/plotxy.dyn -Kb %f\n",
  (numbe+numoe)*4/1000.+.1);
xypipe = popen(plotprog,"w");

fprintf(xypipe,"file *\n");
fprintf(xypipe,"output %s\n",plotname);
fprintf(xypipe,"frame on\n");
fprintf(xypipe,"mode 2\n");
fprintf(xypipe,"xlim 5.650 %f %f\n",x0,x1);
fprintf(xypipe,"ylim 5.698 %f %f\n",y0,y1);
fprintf(xypipe,"read %d\n",(benum+oenum)*3);
if(type==0) {
  for(i=1;i<=benum;i++) {
    fprintf(xypipe,"%f %f\n",bxp[i],byp[i]);
    fprintf(xypipe,"%f %f\n",bxp[i]+bux[i]*xmag,byp[i]);
    fprintf(xypipe,"999. 999.\n");
  }
  for(i=1;i<=oenum;i++) {
    fprintf(xypipe,"%f %f\n",oxp[i],oyp[i]);
    fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,oyp[i]);
    fprintf(xypipe,"999. 999.\n");
  }
  fprintf(xypipe,"note (1.40 -1.5 in) X magnification: %5.2f\n",xmag);
} else if(type==1) {
  for(i=1;i<=benum;i++) {
    fprintf(xypipe,"%f %f\n",bxp[i],byp[i]);
    fprintf(xypipe,"%f %f\n",bxp[i],buy[i]*ymag+byp[i]);
    fprintf(xypipe,"999. 999.\n");
  }
  for(i=1;i<=oenum;i++) {
    fprintf(xypipe,"%f %f\n",oxp[i],oyp[i]);
    fprintf(xypipe,"%f %f\n",oxp[i],ouy[i]*ymag+oyp[i]);
    fprintf(xypipe,"999. 999.\n");
  }
  fprintf(xypipe,"note (1.40 -1.5 in) Y magnification: %5.2f\n",ymag);
} else if(type==2) {
  for(i=1;i<=benum;i++) {
    fprintf(xypipe,"%f %f\n",bxp[i],byp[i]);
```

```
      fprintf(xypipe,"%f %f\n",bxp[i]+bux[i]*xmag,buy[i]*ymag+byp[i]);
      fprintf(xypipe,"999. 999.\n");
    }
    for(i=1;i<=oenum;i++) {
      fprintf(xypipe,"%f %f\n",oxp[i],oyp[i]);
      fprintf(xypipe,"%f %f\n",oxp[i]+oux[i]*xmag,ouy[i]*ymag+oyp[i]);
      fprintf(xypipe,"999. 999.\n");
    }
    fprintf(xypipe,"note (1.40 -2.0 in) X magnification: %5.2f\n",xmag);
    fprintf(xypipe,"note (1.40 -1.5 in) Y magnification: %5.2f\n",ymag);
  }

/* plot dots where the particle that was displaced originated from */
  fprintf(xypipe,"symbol 19 .03 \n");
  fprintf(xypipe,"read %d\n",(benum+oenum));
  for(i=1;i<=benum;i++)
    fprintf(xypipe,"%f %f\n",bxp[i],byp[i]);
  for(i=1;i<=oenum;i++)
    fprintf(xypipe,"%f %f\n",oxp[i],oyp[i]);
  fprintf(xypipe,"symbol -1 0 \n");

  fprintf(xypipe,"plot 1.529 4.513\n");
  fprintf(xypipe,"stop\n");
  if((pclose(xypipe))!=0)
          prob_rpt("\n***Problem in closing plotxy pipe.***\n");
}
```

main

```
main (argc,argv)
int argc;
char *argv[];

{
  int type=-1,n,dec;
  float xmag=1.,ymag=1.,x0,x1,y0,y1;
  char plotname[30];

  dec = 1;
  for(n=1;n<argc;n++)
   if(!strncmp(argv[n],"-dec",4)) dec = (int)(strtod(argv[++n],(char **)NULL));

  printf("Enter the filename for the plot: ");
  scanf("%s",plotname);
  printf("\n");
  while(type<0) {
    printf("Enter the limts of the graph (X0 X1 Y0 Y1):");
    scanf("%f%f%f%f",&x0,&x1,&y0,&y1);
    printf("\nPlot Ux(0), Uy(1), or both(2)? ");
    scanf("%d",&type);
    printf("\n");
```

```
    switch(type) {
        case 0 :
        printf("Enter X-magnification factor: ");
        scanf("%f",&xmag);
        printf("\n");
          break;
        case 1 :
        printf("Enter Y-magnification factor: ");
        scanf("%f",&ymag);
        printf("\n");
          break;
        case 2 :
        printf("Enter X-magnification factor: ");
        scanf("%f",&xmag);
        printf("\n");
        printf("Enter Y-magnification factor: ");
        scanf("%f",&ymag);
        printf("\n");
          break;
        default :
          type = -1;
          break;
    }
  }
  uplot(type,xmag,ymag,plotname,x0,x1,y0,y1,dec);
}
```

Appendix B: Sample Execution

Execution Script

Instruction File

- Instructions

Fault Definition File

- ZeroStrength

Output Files

- Zero_Test/Bdry_Elements
- Zero_Test/Bdry_input
- Zero_Test/Disp.gph
- Zero_Test/Sigma_Spy
- Zero_Test/Spec_Elements
- Zero_Test/logfile
- Zero_Test/plotxy.dir1 *(condensed)*
- Zero_Test/plotxy.dir3 *(condensed)*
- Zero_Test/plotxy.model *(condensed)*

(script)

```
griffy> smudge -in Instructions
Using input file 'Instructions'...
The destination of the output is in directory Zero\_Test.
Solving system of equations using LU Decomposition...
Calculating boundary displacements and stresses...
Calculating specified point displacements and stresses...
Calculating principle stresses...
Outputing Information to disk...
        File: Bdry\_Elements
        File: Spec\_Elements
        File: Disp.gph
        File: Sigma\_Spy
        File: plotxy.dir3
        File: plotxy.dir1
griffy>
```

```
-45 degree fault with strength 00 under 100 MPa N-S compression
number of boundary segments, number of other segment
1 0
Vp Vs Density (Scott 1992, pg 11-12)
6100 3600 2750
Field stress values (XX, XY, YY)
0.0 0.0 -100.0
Boundary Element Input File
ZeroStrength
Other Element Input File
AUTOGRID -80 -80 160 20
Output Destination Directory
Zero_Test
```

```
40 20 -20 -20 20  00. 00.
# A forty element fault extending from (20,-20) to (-20,20).
# Both the positive and negative sides  of the fault have zero strength.
```

Displacements and Stresses at Midpoints of Boundary Elements.

Element	US	UN	UX	UY	Sigma-S	Sigma-N
1	0.062206	0.030939	-0.065864	0.022109	-0.000002	-50.000484
2	0.062206	-0.030939	0.022109	-0.065864	-0.000002	-50.000484
3	0.065372	0.023128	-0.062579	0.029871	-0.000002	-49.999531
4	0.065372	-0.023128	0.029871	-0.062579	-0.000002	-49.999531
5	0.065648	0.018907	-0.059790	0.033051	-0.000002	-49.999920
6	0.065648	-0.018907	0.033051	-0.059790	-0.000002	-49.999920
7	0.065369	0.016039	-0.057564	0.034882	-0.000002	-49.999973
8	0.065369	-0.016039	0.034882	-0.057564	-0.000002	-49.999973
9	0.064920	0.013867	-0.055711	0.036100	-0.000002	-49.998554
10	0.064920	-0.013867	0.036100	-0.055711	-0.000002	-49.998554
11	0.064419	0.012118	-0.054120	0.036982	-0.000002	-50.002243
12	0.064419	-0.012118	0.036982	-0.054120	-0.000002	-50.002243
13	0.063912	0.010654	-0.052726	0.037659	-0.000002	-49.999512
14	0.063912	-0.010654	0.037659	-0.052726	-0.000002	-49.999512
15	0.063422	0.009393	-0.051488	0.038204	-0.000002	-49.999680
16	0.063422	-0.009393	0.038204	-0.051488	-0.000002	-49.999680
17	0.062960	0.008284	-0.050377	0.038662	-0.000002	-49.999573
18	0.062960	-0.008284	0.038662	-0.050377	-0.000002	-49.999573
19	0.062531	0.007292	-0.049372	0.039060	-0.000002	-50.001171
20	0.062531	-0.007292	0.039060	-0.049372	-0.000002	-50.001171
21	0.062140	0.006391	-0.048458	0.039421	-0.000002	-49.999306
22	0.062140	-0.006391	0.039421	-0.048458	-0.000002	-49.999306
23	0.061786	0.005563	-0.047623	0.039756	-0.000002	-49.999466
24	0.061786	-0.005563	0.039756	-0.047623	-0.000002	-49.999466
25	0.061472	0.004793	-0.046857	0.040078	-0.000002	-50.000488
26	0.061472	-0.004793	0.040078	-0.046857	-0.000002	-50.000488
27	0.061197	0.004071	-0.046152	0.040394	-0.000002	-49.999836
28	0.061197	-0.004071	0.040394	-0.046152	-0.000002	-49.999836
29	0.060962	0.003387	-0.045502	0.040711	-0.000002	-49.999832
30	0.060962	-0.003387	0.040711	-0.045502	-0.000002	-49.999832
31	0.060766	0.002733	-0.044901	0.041035	-0.000002	-50.000652
32	0.060766	-0.002733	0.041035	-0.044901	-0.000002	-50.000652
33	0.060610	0.002103	-0.044345	0.041371	-0.000002	-49.999908
34	0.060610	-0.002103	0.041371	-0.044345	-0.000002	-49.999908
35	0.060493	0.001490	-0.043828	0.041721	-0.000002	-49.998547
36	0.060493	-0.001490	0.041721	-0.043828	-0.000002	-49.998547
37	0.060415	0.000889	-0.043348	0.042091	-0.000002	-50.001896
38	0.060415	-0.000889	0.042091	-0.043348	-0.000002	-50.001896
39	0.060375	0.000296	-0.042901	0.042483	-0.000002	-50.000092
40	0.060376	-0.000296	0.042483	-0.042901	-0.000002	-50.000092
41	0.060376	-0.000296	-0.042483	0.042901	-0.000002	-49.999027
42	0.060376	0.000296	0.042901	-0.042483	-0.000002	-49.999027
43	0.060415	-0.000889	-0.042091	0.043348	-0.000002	-50.000549
44	0.060415	0.000889	0.043348	-0.042091	-0.000002	-50.000549

45	0.060493	-0.001490	-0.041721	0.043828	-0.000002	-49.999088
46	0.060493	0.001490	0.043828	-0.041721	-0.000002	-49.999088
47	0.060610	-0.002103	-0.041371	0.044344	-0.000002	-50.000820
48	0.060610	0.002103	0.044344	-0.041371	-0.000002	-50.000820
49	0.060766	-0.002733	-0.041035	0.044901	-0.000002	-50.000168
50	0.060766	0.002733	0.044901	-0.041035	-0.000002	-50.000168
51	0.060962	-0.003387	-0.040711	0.045502	-0.000002	-49.999649
52	0.060962	0.003387	0.045502	-0.040711	-0.000002	-49.999649
53	0.061197	-0.004071	-0.040394	0.046152	-0.000002	-49.999866
54	0.061197	0.004071	0.046152	-0.040394	-0.000002	-49.999866
55	0.061472	-0.004793	-0.040078	0.046857	-0.000002	-49.999722
56	0.061472	0.004793	0.046857	-0.040078	-0.000002	-49.999722
57	0.061786	-0.005563	-0.039756	0.047623	-0.000002	-50.001022
58	0.061786	0.005563	0.047623	-0.039756	-0.000002	-50.001022
59	0.062140	-0.006391	-0.039420	0.048458	-0.000002	-49.999844
60	0.062140	0.006391	0.048458	-0.039420	-0.000002	-49.999844
61	0.062531	-0.007292	-0.039060	0.049372	-0.000002	-50.000172
62	0.062531	0.007292	0.049372	-0.039060	-0.000002	-50.000172
63	0.062960	-0.008284	-0.038662	0.050377	-0.000002	-49.999001
64	0.062960	0.008284	0.050377	-0.038662	-0.000002	-49.999001
65	0.063422	-0.009393	-0.038204	0.051488	-0.000002	-50.000153
66	0.063422	0.009393	0.051488	-0.038204	-0.000002	-50.000153
67	0.063912	-0.010654	-0.037659	0.052726	-0.000002	-49.999939
68	0.063912	0.010654	0.052726	-0.037659	-0.000002	-49.999939
69	0.064419	-0.012118	-0.036982	0.054120	-0.000002	-49.999802
70	0.064419	0.012118	0.054120	-0.036982	-0.000002	-49.999802
71	0.064920	-0.013867	-0.036100	0.055711	-0.000002	-50.000622
72	0.064920	0.013867	0.055711	-0.036100	-0.000002	-50.000622
73	0.065369	-0.016039	-0.034882	0.057564	-0.000002	-49.999901
74	0.065369	0.016039	0.057564	-0.034882	-0.000002	-49.999901
75	0.065648	-0.018907	-0.033051	0.059790	-0.000002	-50.000397
76	0.065648	0.018907	0.059790	-0.033051	-0.000002	-50.000397
77	0.065372	-0.023128	-0.029871	0.062579	-0.000002	-49.999680
78	0.065372	0.023128	0.062579	-0.029871	-0.000002	-49.999680
79	0.062206	-0.030939	-0.022109	0.065864	-0.000002	-49.999886
80	0.062206	0.030939	0.065864	-0.022109	-0.000002	-49.999886

Boundary Element Data

Element	Center X	Center Y	Length	Angle	Sigma-S
1+	19.5000	-19.5000	1.4142	135.000	0.000000
1-	19.5000	-19.5000	1.4142	315.000	0.000000
2+	18.5000	-18.5000	1.4142	135.000	0.000000
2-	18.5000	-18.5000	1.4142	315.000	0.000000
3+	17.5000	-17.5000	1.4142	135.000	0.000000
3-	17.5000	-17.5000	1.4142	315.000	0.000000
4+	16.5000	-16.5000	1.4142	135.000	0.000000
4-	16.5000	-16.5000	1.4142	315.000	0.000000
5+	15.5000	-15.5000	1.4142	135.000	0.000000
5-	15.5000	-15.5000	1.4142	315.000	0.000000
6+	14.5000	-14.5000	1.4142	135.000	0.000000
6-	14.5000	-14.5000	1.4142	315.000	0.000000
7+	13.5000	-13.5000	1.4142	135.000	0.000000
7-	13.5000	-13.5000	1.4142	315.000	0.000000
8+	12.5000	-12.5000	1.4142	135.000	0.000000
8-	12.5000	-12.5000	1.4142	315.000	0.000000
9+	11.5000	-11.5000	1.4142	135.000	0.000000
9-	11.5000	-11.5000	1.4142	315.000	0.000000
10+	10.5000	-10.5000	1.4142	135.000	0.000000
10-	10.5000	-10.5000	1.4142	315.000	0.000000
11+	9.5000	-9.5000	1.4142	135.000	0.000000
11-	9.5000	-9.5000	1.4142	315.000	0.000000
12+	8.5000	-8.5000	1.4142	135.000	0.000000
12-	8.5000	-8.5000	1.4142	315.000	0.000000
13+	7.5000	-7.5000	1.4142	135.000	0.000000
13-	7.5000	-7.5000	1.4142	315.000	0.000000
14+	6.5000	-6.5000	1.4142	135.000	0.000000
14-	6.5000	-6.5000	1.4142	315.000	0.000000
15+	5.5000	-5.5000	1.4142	135.000	0.000000
15-	5.5000	-5.5000	1.4142	315.000	0.000000
16+	4.5000	-4.5000	1.4142	135.000	0.000000
16-	4.5000	-4.5000	1.4142	315.000	0.000000
17+	3.5000	-3.5000	1.4142	135.000	0.000000
17-	3.5000	-3.5000	1.4142	315.000	0.000000
18+	2.5000	-2.5000	1.4142	135.000	0.000000
18-	2.5000	-2.5000	1.4142	315.000	0.000000
19+	1.5000	-1.5000	1.4142	135.000	0.000000
19-	1.5000	-1.5000	1.4142	315.000	0.000000
20+	0.5000	-0.5000	1.4142	135.000	0.000000
20-	0.5000	-0.5000	1.4142	315.000	0.000000
21+	-0.5000	0.5000	1.4142	135.000	0.000000
21-	-0.5000	0.5000	1.4142	315.000	0.000000
22+	-1.5000	1.5000	1.4142	135.000	0.000000

22-	-1.5000	1.5000	1.4142	315.000	0.000000
23+	-2.5000	2.5000	1.4142	135.000	0.000000
23-	-2.5000	2.5000	1.4142	315.000	0.000000
24+	-3.5000	3.5000	1.4142	135.000	0.000000
24-	-3.5000	3.5000	1.4142	315.000	0.000000
25+	-4.5000	4.5000	1.4142	135.000	0.000000
25-	-4.5000	4.5000	1.4142	315.000	0.000000
26+	-5.5000	5.5000	1.4142	135.000	0.000000
26-	-5.5000	5.5000	1.4142	315.000	0.000000
27+	-6.5000	6.5000	1.4142	135.000	0.000000
27-	-6.5000	6.5000	1.4142	315.000	0.000000
28+	-7.5000	7.5000	1.4142	135.000	0.000000
28-	-7.5000	7.5000	1.4142	315.000	0.000000
29+	-8.5000	8.5000	1.4142	135.000	0.000000
29-	-8.5000	8.5000	1.4142	315.000	0.000000
30+	-9.5000	9.5000	1.4142	135.000	0.000000
30-	-9.5000	9.5000	1.4142	315.000	0.000000
31+	-10.5000	10.5000	1.4142	135.000	0.000000
31-	-10.5000	10.5000	1.4142	315.000	0.000000
32+	-11.5000	11.5000	1.4142	135.000	0.000000
32-	-11.5000	11.5000	1.4142	315.000	0.000000
33+	-12.5000	12.5000	1.4142	135.000	0.000000
33-	-12.5000	12.5000	1.4142	315.000	0.000000
34+	-13.5000	13.5000	1.4142	135.000	0.000000
34-	-13.5000	13.5000	1.4142	315.000	0.000000
35+	-14.5000	14.5000	1.4142	135.000	0.000000
35-	-14.5000	14.5000	1.4142	315.000	0.000000
36+	-15.5000	15.5000	1.4142	135.000	0.000000
36-	-15.5000	15.5000	1.4142	315.000	0.000000
37+	-16.5000	16.5000	1.4142	135.000	0.000000
37-	-16.5000	16.5000	1.4142	315.000	0.000000
38+	-17.5000	17.5000	1.4142	135.000	0.000000
38-	-17.5000	17.5000	1.4142	315.000	0.000000
39+	-18.5000	18.5000	1.4142	135.000	0.000000
39-	-18.5000	18.5000	1.4142	315.000	0.000000
40+	-19.5000	19.5000	1.4142	135.000	0.000000
40-	-19.5000	19.5000	1.4142	315.000	0.000000

```
80 boundary elements.
462 specified points.
19.500000 -19.500000 -0.065864 0.022109
19.500000 -19.500000 0.022109 -0.065864
18.500000 -18.500000 -0.062579 0.029871
18.500000 -18.500000 0.029871 -0.062579
17.500000 -17.500000 -0.059790 0.033051
17.500000 -17.500000 0.033051 -0.059790
16.500000 -16.500000 -0.057564 0.034882
16.500000 -16.500000 0.034882 -0.057564
15.500000 -15.500000 -0.055711 0.036100
15.500000 -15.500000 0.036100 -0.055711
14.500000 -14.500000 -0.054120 0.036982
14.500000 -14.500000 0.036982 -0.054120
13.500000 -13.500000 -0.052726 0.037659
13.500000 -13.500000 0.037659 -0.052726
12.500000 -12.500000 -0.051488 0.038204
12.500000 -12.500000 0.038204 -0.051488
11.500000 -11.500000 -0.050377 0.038662
11.500000 -11.500000 0.038662 -0.050377
10.500000 -10.500000 -0.049372 0.039060
10.500000 -10.500000 0.039060 -0.049372
9.500000 -9.500000 -0.048458 0.039421
9.500000 -9.500000 0.039421 -0.048458
8.500000 -8.500000 -0.047623 0.039756
8.500000 -8.500000 0.039756 -0.047623
7.500000 -7.500000 -0.046857 0.040078
7.500000 -7.500000 0.040078 -0.046857
6.500000 -6.500000 -0.046152 0.040394
6.500000 -6.500000 0.040394 -0.046152
5.500000 -5.500000 -0.045502 0.040711
5.500000 -5.500000 0.040711 -0.045502
4.500000 -4.500000 -0.044901 0.041035
4.500000 -4.500000 0.041035 -0.044901
3.500000 -3.500000 -0.044345 0.041371
3.500000 -3.500000 0.041371 -0.044345
2.500000 -2.500000 -0.043828 0.041721
2.500000 -2.500000 0.041721 -0.043828
1.500000 -1.500000 -0.043348 0.042091
1.500000 -1.500000 0.042091 -0.043348
0.500000 -0.500000 -0.042901 0.042483
0.500000 -0.500000 0.042483 -0.042901
-0.500000 0.500000 -0.042483 0.042901
-0.500000 0.500000 0.042901 -0.042483
-1.500000 1.500000 -0.042091 0.043348
-1.500000 1.500000 0.043348 -0.042091
-2.500000 2.500000 -0.041721 0.043828
-2.500000 2.500000 0.043828 -0.041721
-3.500000 3.500000 -0.041371 0.044344
```

```
-26.666666 48.000000 0.000253 -0.015055
-19.047619 48.000000 0.000754 -0.019461
-11.428572 48.000000 0.000420 -0.021794
-3.809524 48.000000 0.000053 -0.021550
3.809524 48.000000 0.000125 -0.019523
11.428572 48.000000 0.000617 -0.016667
19.047619 48.000000 0.001368 -0.013612
26.666668 48.000000 0.002243 -0.010710
34.285713 48.000000 0.003144 -0.008149
41.904762 48.000000 0.003998 -0.006010
49.523811 48.000000 0.004753 -0.004304
57.142857 48.000000 0.005376 -0.002991
64.761902 48.000000 0.005857 -0.002009
72.380951 48.000000 0.006203 -0.001292
80.000000 48.000000 0.006427 -0.000778
-80.000000 56.000000 -0.005719 -0.000768
-72.380951 56.000000 -0.005301 -0.001344
-64.761902 56.000000 -0.004680 -0.002194
-57.142857 56.000000 -0.003818 -0.003438
-49.523811 56.000000 -0.002713 -0.005224
-41.904762 56.000000 -0.001447 -0.007684
-34.285713 56.000000 -0.000240 -0.010805
-26.666666 56.000000 0.000570 -0.014235
-19.047619 56.000000 0.000749 -0.017206
-11.428572 56.000000 0.000424 -0.018892
-3.809524 56.000000 0.000002 -0.018969
3.809524 56.000000 -0.000180 -0.017715
11.428572 56.000000 -0.000019 -0.015650
19.047619 56.000000 0.000433 -0.013244
26.666668 56.000000 0.001078 -0.010821
34.285713 56.000000 0.001821 -0.008580
41.904762 56.000000 0.002583 -0.006624
49.523811 56.000000 0.003303 -0.004991
57.142857 56.000000 0.003943 -0.003675
64.761902 56.000000 0.004480 -0.002644
72.380951 56.000000 0.004909 -0.001852
80.000000 56.000000 0.005233 -0.001255
-80.000000 64.000000 -0.004421 -0.001387
-72.380951 64.000000 -0.003912 -0.002089
-64.761902 64.000000 -0.003248 -0.003058
-57.142857 64.000000 -0.002432 -0.004373
-49.523811 64.000000 -0.001504 -0.006105
-41.904762 64.000000 -0.000565 -0.008275
-34.285713 64.000000 0.000220 -0.010777
-26.666666 64.000000 0.000669 -0.013317
-19.047619 64.000000 0.000696 -0.015432
-11.428572 64.000000 0.000398 -0.016679
-3.809524 64.000000 0.000005 -0.016856
3.809524 64.000000 -0.000260 -0.016060
```

```
11.428572 64.000000 -0.000281 -0.014572
19.047619 64.000000 -0.000049 -0.012707
26.666668 64.000000 0.000385 -0.010722
34.285713 64.000000 0.000952 -0.008800
41.904762 64.000000 0.001583 -0.007051
49.523811 64.000000 0.002220 -0.005531
57.142857 64.000000 0.002822 -0.004256
64.761902 64.000000 0.003359 -0.003215
72.380951 64.000000 0.003818 -0.002384
80.000000 64.000000 0.004194 -0.001731
-80.000000 72.000000 -0.003356 -0.001978
-72.380951 72.000000 -0.002828 -0.002755
-64.761902 72.000000 -0.002199 -0.003767
-57.142857 72.000000 -0.001490 -0.005056
-49.523811 72.000000 -0.000753 -0.006643
-41.904762 72.000000 -0.000075 -0.008497
-34.285713 72.000000 0.000433 -0.010497
-26.666666 72.000000 0.000678 -0.012424
-19.047619 72.000000 0.000631 -0.013989
-11.428572 72.000000 0.000363 -0.014935
-3.809524 72.000000 0.000016 -0.015130
3.809524 72.000000 -0.000262 -0.014607
11.428572 72.000000 -0.000372 -0.013524
19.047619 72.000000 -0.000284 -0.012081
26.666668 72.000000 -0.000012 -0.010472
34.285713 72.000000 0.000398 -0.008846
41.904762 72.000000 0.000895 -0.007310
49.523811 72.000000 0.001431 -0.005925
57.142857 72.000000 0.001966 -0.004721
64.761902 72.000000 0.002469 -0.003704
72.380951 72.000000 0.002922 -0.002862
80.000000 72.000000 0.003314 -0.002178
-80.000000 80.000000 -0.002506 -0.002506
-72.380951 80.000000 -0.002005 -0.003311
-64.761902 80.000000 -0.001445 -0.004310
-57.142857 80.000000 -0.000856 -0.005520
-49.523811 80.000000 -0.000287 -0.006931
-41.904762 80.000000 0.000196 -0.008494
-34.285713 80.000000 0.000525 -0.010103
-26.666666 80.000000 0.000649 -0.011597
-19.047619 80.000000 0.000566 -0.012790
-11.428572 80.000000 0.000326 -0.013522
-3.809524 80.000000 0.000025 -0.013707
3.809524 80.000000 -0.000239 -0.013353
11.428572 80.000000 -0.000388 -0.012550
19.047619 80.000000 -0.000386 -0.011428
26.666668 80.000000 -0.000231 -0.010126
34.285713 80.000000 0.000054 -0.008763
41.904762 80.000000 0.000432 -0.007430
```

```
49.523811 80.000000 0.000867 -0.006189
57.142857 80.000000 0.001324 -0.005075
64.761902 80.000000 0.001775 -0.004104
72.380951 80.000000 0.002199 -0.003276
80.000000 80.000000 0.002583 -0.002583
```

```
462 specified points.
80 boundary elements.
-80.000000 -80.000000 -103.379753 3.379753 -90.000000 88.323555
-72.380951 -80.000000 -103.199036 3.982277 -89.637886 0.361945
-64.761902 -80.000000 -102.774971 4.552246 -89.201714 0.798277
-57.142857 -80.000000 -102.030724 5.038261 -88.699524 1.300480
-49.523811 -80.000000 -100.887894 5.376717 -88.151894 1.848092
-41.904762 -80.000000 -99.279037 5.495464 -87.597183 2.402794
-34.285713 -80.000000 -97.167870 5.322533 -87.098610 2.901405
-26.666666 -80.000000 -94.579048 4.803886 -86.752518 3.247486
-19.047619 -80.000000 -91.642685 3.936253 -86.695053 3.304931
-11.428572 -80.000000 -88.650421 2.819172 -87.093826 2.906183
-3.809524 -80.000000 -86.099915 1.711136 -88.093925 1.906029
3.809524 -80.000000 -84.650314 1.024418 -89.685028 0.314715
11.428572 -80.000000 -84.884583 1.155590 88.446991 -1.553068
19.047619 -80.000000 -86.932182 2.184975 86.852074 -3.147907
26.666668 -80.000000 -90.301086 3.749828 85.965240 -4.034748
34.285713 -80.000000 -94.145859 5.280521 85.859436 -4.140582
41.904762 -80.000000 -97.698303 6.338253 86.332497 -3.667518
49.523811 -80.000000 -100.511337 6.759743 87.110741 -2.889231
57.142857 -80.000000 -102.462112 6.605434 87.972176 -2.027842
64.761902 -80.000000 -103.637131 6.041679 88.778137 -1.221887
72.380951 -80.000000 -104.210876 5.247143 89.460876 -0.539060
80.000000 -80.000000 -104.365944 4.365952 90.000000 -89.077019
-80.000000 -72.000000 -104.011932 3.184204 89.617805 -0.382297
-72.380951 -72.000000 -103.998260 3.952724 89.979362 -0.021075
-64.761902 -72.000000 -103.736420 4.733379 -89.560059 0.439960
-57.142857 -72.000000 -103.124435 5.471451 -89.000961 0.999044
-49.523811 -72.000000 -102.048965 6.092964 -88.356705 1.643293
-41.904762 -72.000000 -100.397110 6.505527 -87.659752 2.340230
-34.285713 -72.000000 -98.078033 6.602978 -86.970818 3.029168
-26.666666 -72.000000 -95.056915 6.278240 -86.392838 3.607166
-19.047619 -72.000000 -91.412170 5.455875 -86.091614 3.908391
-11.428572 -72.000000 -87.432068 4.165951 -86.314781 3.685203
-3.809524 -72.000000 -83.746872 2.673721 -87.361305 2.638738
3.809524 -72.000000 -81.392975 1.588436 -89.381271 0.618749
11.428572 -72.000000 -81.496933 1.668789 88.023895 -1.976107
19.047619 -72.000000 -84.438316 3.159126 85.843002 -4.157032
26.666668 -72.000000 -89.334869 5.392849 84.851662 -5.148343
34.285713 -72.000000 -94.660400 7.350773 85.046280 -4.953719
41.904762 -72.000000 -99.197441 8.398201 85.969048 -4.030950
49.523811 -72.000000 -102.417946 8.453781 87.157143 -2.842828
57.142857 -72.000000 -104.342461 7.766518 88.310066 -1.689939
64.761902 -72.000000 -105.252655 6.663788 89.282738 -0.717200
72.380951 -72.000000 -105.476974 5.414616 -89.967781 0.030671
80.000000 -72.000000 -105.289932 4.193562 -89.430122 0.570014
-80.000000 -64.000000 -104.605545 2.716286 89.154236 -0.845750
-72.380951 -64.000000 -104.805374 3.644096 89.488625 -0.511439
-64.761902 -64.000000 -104.770721 4.646984 89.946907 -0.051735
```

```
-57.142857 -64.000000 -104.373199 5.673496 -89.461487 0.538499
-49.523811 -64.000000 -103.459152 6.646641 -88.739700 1.260266
-41.904762 -64.000000 -101.860748 7.462151 -87.910156 2.089843
-34.285713 -64.000000 -99.417229 7.987991 -87.023613 2.976402
-26.666666 -64.000000 -96.007256 8.065674 -86.174889 3.825099
-19.047619 -64.000000 -91.605972 7.523079 -85.535286 4.464717
-11.428572 -64.000000 -86.406036 6.238567 -85.413704 4.586286
-3.809524 -64.000000 -81.074883 4.345413 -86.320175 3.679817
3.809524 -64.000000 -77.128403 2.616543 -88.800522 1.199471
11.428572 -64.000000 -76.789307 2.510189 87.461983 -2.538005
19.047619 -64.000000 -81.175095 4.762737 84.348564 -5.651443
26.666668 -64.000000 -88.592957 8.051140 83.364296 -6.635698
34.285713 -64.000000 -96.100891 10.449879 84.188164 -5.811834
41.904762 -64.000000 -101.778824 11.136070 85.826820 -4.173197
49.523811 -64.000000 -105.187149 10.364201 87.549065 -2.450944
57.142857 -64.000000 -106.744987 8.769829 89.004478 -0.995528
64.761902 -64.000000 -107.089722 6.905987 -89.907661 0.092352
72.380951 -64.000000 -106.757965 5.112698 -89.165497 0.834501
80.000000 -64.000000 -106.111214 3.549789 -88.702415 1.297559
-80.000000 -56.000000 -105.100151 1.886566 88.619576 -1.380413
-72.380951 -56.000000 -105.555435 2.938663 88.891975 -1.107953
-64.761902 -56.000000 -105.815407 4.145210 89.310417 -0.689603
-57.142857 -56.000000 -105.725685 5.466202 89.896767 -0.103098
-49.523811 -56.000000 -105.087296 6.831226 -89.339752 0.660264
-41.904762 -56.000000 -103.668060 8.137653 -88.409927 1.590084
-34.285713 -56.000000 -101.225677 9.250092 -87.351730 2.648259
-26.666666 -56.000000 -97.537903 9.992298 -86.240028 3.759974
-19.047619 -56.000000 -92.438019 10.121777 -85.214180 4.785827
-11.428572 -56.000000 -85.900002 9.311363 -84.566406 5.433580
-3.809524 -56.000000 -78.354721 7.300957 -84.963898 5.036097
3.809524 -56.000000 -71.612312 4.654396 -87.678169 2.321855
11.428572 -56.000000 -69.941818 3.984432 86.757080 -3.242905
19.047619 -56.000000 -76.787186 7.587292 82.036858 -7.963138
26.666668 -56.000000 -88.629562 12.578613 81.435478 -8.564521
34.285713 -56.000000 -99.331451 15.114494 83.508461 -6.491529
41.904762 -56.000000 -106.030022 14.548347 86.211334 -3.788689
49.523811 -56.000000 -108.996964 12.124947 88.537041 -1.462907
57.142857 -56.000000 -109.574265 9.140774 -89.790833 0.209298
64.761902 -56.000000 -108.944305 6.343449 -88.725967 1.274026
72.380951 -56.000000 -107.844650 4.023190 -88.125153 1.874854
80.000000 -56.000000 -106.659187 2.222565 -87.844009 2.155986
-80.000000 -48.000000 -105.420433 0.605164 88.039902 -1.960132
-72.380951 -48.000000 -106.160477 1.706562 88.207756 -1.792306
-64.761902 -48.000000 -106.777451 3.052128 88.537811 -1.462218
-57.142857 -48.000000 -107.094040 4.621891 89.067802 -0.932238
-49.523811 -48.000000 -106.859802 6.359859 89.821838 -0.178022
-41.904762 -48.000000 -105.762650 8.175262 -89.201012 0.798975
-34.285713 -48.000000 -103.463982 9.954948 -88.034103 1.965898
-26.666666 -48.000000 -99.644585 11.572380 -86.741302 3.258693
```

```
-19.047619 -48.000000 -94.019211 12.846497 -85.412086 4.587923
-11.428572 -48.000000 -86.296616 13.384399 -84.206520 5.793481
-3.809524 -48.000000 -76.264114 12.363232 -83.645645 6.354347
3.809524 -48.000000 -64.980499 9.061069 -85.645645 4.354345
11.428572 -48.000000 -59.394199 6.837322 86.116379 -3.883614
19.047619 -48.000000 -70.668251 13.059031 78.273567 -11.726434
26.666668 -48.000000 -90.918823 20.756756 79.191063 -10.808928
34.285713 -48.000000 -106.001228 21.925583 83.587646 -6.412364
41.904762 -48.000000 -112.634422 18.043266 87.666260 -2.333714
49.523811 -48.000000 -113.740433 12.740337 -89.547325 0.452712
57.142857 -48.000000 -112.417732 7.995186 -87.943138 2.056886
64.761902 -48.000000 -110.398148 4.360157 -87.169617 2.830399
72.380951 -48.000000 -108.419449 1.776104 -86.910698 3.089308
80.000000 -48.000000 -106.725708 0.017651 -86.942528 3.057470
-80.000000 -40.000000 -105.482155 -1.201263 87.464149 -2.535871
-72.380951 -40.000000 -106.511383 -0.177158 87.478981 -2.520994
-64.761902 -40.000000 -107.531387 1.180073 87.661285 -2.338728
-57.142857 -40.000000 -108.351654 2.879822 88.070961 -1.929074
-49.523811 -40.000000 -108.662888 4.883556 88.753952 -1.246022
-41.904762 -40.000000 -108.039642 7.104240 89.718994 -0.280895
-34.285713 -40.000000 -105.999573 9.443516 -89.082657 0.917350
-26.666666 -40.000000 -102.116501 11.870945 -87.754097 2.245870
-19.047619 -40.000000 -96.096420 14.481709 -86.400597 3.599401
-11.428572 -40.000000 -87.615570 17.337948 -85.033440 4.966572
-3.809524 -40.000000 -75.767426 19.744831 -83.607712 6.392295
3.809524 -40.000000 -58.890175 18.865150 -83.179207 6.820787
11.428572 -40.000000 -42.291626 13.471280 87.042999 -2.957008
19.047619 -40.000000 -61.623596 25.315842 71.824768 -18.175228
26.666668 -40.000000 -99.748535 36.412968 77.530922 -12.469069
34.285713 -40.000000 -119.021927 30.381500 85.757759 -4.242217
41.904762 -40.000000 -121.712967 19.192123 -88.977112 1.022850
49.523811 -40.000000 -118.457047 10.071930 -86.407173 3.592849
57.142857 -40.000000 -114.327911 4.010071 -85.478233 4.521766
64.761902 -40.000000 -110.808899 0.283745 -85.403625 4.596375
72.380951 -40.000000 -108.117676 -1.905231 -85.718323 4.281660
80.000000 -40.000000 -106.131737 -3.131760 -86.175262 3.824738
-80.000000 -32.000000 -105.204742 -3.562046 86.973358 -3.026654
-72.380951 -32.000000 -106.485458 -2.803185 86.785759 -3.214275
-64.761902 -32.000000 -107.919678 -1.641449 86.750740 -3.249266
-57.142857 -32.000000 -109.330902 -0.023190 86.960762 -3.039243
-49.523811 -32.000000 -110.354073 2.035271 87.502899 -2.497089
-41.904762 -32.000000 -110.392899 4.418507 88.412964 -1.587036
-34.285713 -32.000000 -108.687424 6.952549 89.613861 -0.385899
-26.666666 -32.000000 -104.567635 9.592148 -89.127274 0.872699
-19.047619 -32.000000 -97.809090 12.760468 -88.146957 1.853058
-11.428572 -32.000000 -88.653076 17.622784 -87.617943 2.382031
-3.809524 -32.000000 -76.677368 25.699776 -87.025414 2.974592
3.809524 -32.000000 -57.446915 36.265800 -84.915802 5.084192
11.428572 -32.000000 -18.852903 36.230370 -84.871918 5.128109
```

```
19.047619 -32.000000 -45.858345 60.651115 60.541943 -29.458059
26.666668 -32.000000 -129.930786 66.112015 80.060547 -9.939452
34.285713 -32.000000 -141.119339 32.945602 -87.327217 2.672801
41.904762 -32.000000 -130.432388 11.438873 -82.923492 7.076495
49.523811 -32.000000 -120.573395 0.882069 -82.221146 7.778850
57.142857 -32.000000 -113.878632 -4.020512 -82.885040 7.114958
64.761902 -32.000000 -109.545746 -6.150856 -83.917610 6.082410
72.380951 -32.000000 -106.726959 -6.915371 -84.935738 5.064263
80.000000 -32.000000 -104.856064 -7.006657 -85.818161 4.181827
-80.000000 -24.000000 -104.537598 -6.420101 86.691414 -3.308558
-72.380951 -24.000000 -105.967651 -6.178829 86.264709 -3.735291
-64.761902 -24.000000 -107.758018 -5.518272 85.937592 -4.062394
-57.142857 -24.000000 -109.809738 -4.312897 85.844452 -4.155550
-49.523811 -24.000000 -111.756073 -2.523289 86.151817 -3.848203
-41.904762 -24.000000 -112.786163 -0.310059 86.987640 -3.012345
-34.285713 -24.000000 -111.603165 1.907253 88.299690 -1.700311
-26.666666 -24.000000 -106.830811 3.668644 89.665894 -0.334378
-19.047619 -24.000000 -98.018555 5.205235 -89.840851 0.158942
-11.428572 -24.000000 -86.587997 8.573139 88.563065 -1.436989
-3.809524 -24.000000 -75.061790 18.555458 84.618698 -5.381314
3.809524 -24.000000 -62.177277 43.521553 80.929367 -9.070642
11.428572 -24.000000 -23.262428 102.511414 83.729515 -6.270483
19.047619 -24.000000 35.587372 254.188843 39.280956 -50.719040
26.666668 -24.000000 -227.919617 69.953735 -79.265770 10.734228
34.285713 -24.000000 -158.414215 -0.871132 -72.890335 17.109661
41.904762 -24.000000 -129.653244 -13.997997 -75.406082 14.593928
49.523811 -24.000000 -116.748535 -16.186222 -78.801811 11.198195
57.142857 -24.000000 -110.235924 -15.475006 -81.583214 8.416777
64.761902 -24.000000 -106.639236 -14.000305 -83.633202 6.366785
72.380951 -24.000000 -104.509323 -12.430546 -85.111320 4.888682
80.000000 -24.000000 -103.176758 -10.977722 -86.182671 3.817316
-80.000000 -16.000000 -103.508919 -9.570610 86.790154 -3.209833
-72.380951 -16.000000 -104.904549 -10.143654 86.135414 -3.864615
-64.761902 -16.000000 -106.867813 -10.401154 85.463722 -4.536273
-57.142857 -16.000000 -109.486160 -10.115925 84.935059 -5.064949
-49.523811 -16.000000 -112.571884 -9.094872 84.835876 -5.164124
-41.904762 -16.000000 -115.225670 -7.423286 85.529099 -4.470907
-34.285713 -16.000000 -115.336479 -5.832512 87.221024 -2.778988
-26.666666 -16.000000 -109.833618 -5.777000 89.496323 -0.503542
-19.047619 -16.000000 -96.877167 -8.387329 -89.356247 0.643922
-11.428572 -16.000000 -79.267380 -12.071266 86.702438 -3.297554
-3.809524 -16.000000 -65.822624 -10.131958 72.354034 -17.645969
3.809524 -16.000000 -65.377747 5.230110 51.877537 -38.122467
11.428572 -16.000000 -95.779114 42.996521 25.344282 -64.655708
19.047619 -16.000000 -356.537476 9.036407 -35.975273 54.024731
26.666668 -16.000000 -176.605377 -155.002716 -65.250870 24.749134
34.285713 -16.000000 -130.924820 -73.435837 -67.670929 22.329054
41.904762 -16.000000 -115.344971 -46.898346 -76.402084 13.597909
49.523811 -16.000000 -108.668823 -33.813553 -81.085052 8.914942
```

```
57.142857 -16.000000 -105.330093 -25.927784 -83.813202 6.186795
64.761902 -16.000000 -103.479034 -20.649746 -85.519600 4.480419
72.380951 -16.000000 -102.375206 -16.886131 -86.646324 3.353647
80.000000 -16.000000 -101.680023 -14.085575 -87.422325 2.577660
-80.000000 -8.000000 -102.297440 -12.589256 87.464455 -2.535545
-72.380951 -8.000000 -103.420349 -14.232056 86.710358 -3.289636
-64.761902 -8.000000 -105.206902 -15.886917 85.760895 -4.239093
-57.142857 -8.000000 -108.020767 -17.272957 84.704994 -5.295015
-49.523811 -8.000000 -112.190811 -17.934082 83.887230 -6.112783
-41.904762 -8.000000 -117.359673 -17.475502 84.067940 -5.932062
-34.285713 -8.000000 -120.907150 -16.397205 86.269035 -3.730926
-26.666666 -8.000000 -116.570290 -17.226860 -89.049004 0.951015
-19.047619 -8.000000 -98.287994 -23.613369 -83.303177 6.696829
-11.428572 -8.000000 -70.195259 -35.737259 -82.065521 7.934443
-3.809524 -8.000000 -53.433640 -42.271652 45.245808 -44.754196
3.809524 -8.000000 -66.760513 -35.713608 9.010546 -80.989487
11.428572 -8.000000 -81.300552 -17.518543 -9.708527 80.291466
19.047619 -8.000000 -150.533493 -35.371624 -57.143047 32.856953
26.666668 -8.000000 -163.246185 -57.371414 -89.483437 0.516585
34.285713 -8.000000 -130.412857 -59.280712 83.044991 -6.955019
41.904762 -8.000000 -112.807663 -48.801804 87.018188 -2.981821
49.523811 -8.000000 -106.256073 -37.627537 89.685081 -0.315215
57.142857 -8.000000 -103.514038 -29.175640 -89.415749 0.584180
64.761902 -8.000000 -102.166321 -23.113834 -89.198906 0.801212
72.380951 -8.000000 -101.423248 -18.717369 -89.209602 0.790458
80.000000 -8.000000 -100.979584 -15.452099 -89.288765 0.711094
-80.000000 0.000000 -101.286530 -14.805065 88.827171 -1.172842
-72.380951 0.000000 -101.984329 -17.521580 88.290588 -1.709402
-64.761902 0.000000 -103.210106 -20.839912 87.452538 -2.547442
-57.142857 0.000000 -105.468857 -24.714039 86.173752 -3.826269
-49.523811 0.000000 -109.743462 -28.709007 84.454453 -5.545539
-41.904762 0.000000 -117.454056 -31.620102 83.081306 -6.918669
-34.285713 0.000000 -128.213211 -31.876888 84.645645 -5.354340
-26.666666 0.000000 -132.348236 -30.507160 -87.224297 2.775761
-19.047619 0.000000 -110.993065 -33.347221 -71.068115 18.931889
-11.428572 0.000000 -73.389503 -40.442909 -44.289536 45.710461
-3.809524 0.000000 -57.853931 -40.496777 -2.448783 87.551315
3.809524 0.000000 -57.853962 -40.496471 -2.448977 87.550964
11.428572 0.000000 -73.389534 -40.442879 -44.289581 45.710423
19.047619 0.000000 -110.993065 -33.347252 -71.068115 18.931875
26.666668 0.000000 -132.348312 -30.507187 -87.224304 2.775679
34.285713 0.000000 -128.213226 -31.876884 84.645645 -5.354365
41.904762 0.000000 -117.454056 -31.620102 83.081306 -6.918690
49.523811 0.000000 -109.743500 -28.709015 84.454460 -5.545526
57.142857 0.000000 -105.468849 -24.714043 86.173752 -3.826291
64.761902 0.000000 -103.210075 -20.839905 87.452538 -2.547473
72.380951 0.000000 -101.984344 -17.521580 88.290588 -1.709359
80.000000 0.000000 -101.286552 -14.805069 88.827171 -1.172780
-80.000000 8.000000 -100.979607 -15.452095 -89.288765 0.711300
```

-72.380951	8.000000	-101.423256	-18.717369	-89.209602	0.790458
-64.761902	8.000000	-102.166306	-23.113842	-89.198906	0.801212
-57.142857	8.000000	-103.514038	-29.175644	-89.415749	0.584038
-49.523811	8.000000	-106.256088	-37.627533	89.685081	-0.314635
-41.904762	8.000000	-112.807663	-48.801800	87.018181	-2.981818
-34.285713	8.000000	-130.412933	-59.280708	83.044991	-6.955011
-26.666666	8.000000	-163.246155	-57.371414	-89.483421	0.516796
-19.047619	8.000000	-150.533539	-35.371635	-57.143066	32.856937
-11.428572	8.000000	-81.300392	-17.518333	-9.708630	80.291367
-3.809524	8.000000	-66.760483	-35.713547	9.010674	-80.989357
3.809524	8.000000	-53.433651	-42.271641	45.245930	-44.754086
11.428572	8.000000	-70.195297	-35.737247	-82.065521	7.934477
19.047619	8.000000	-98.287987	-23.613380	-83.303177	6.696813
26.666668	8.000000	-116.570290	-17.226845	-89.048996	0.951011
34.285713	8.000000	-120.907135	-16.397205	86.269035	-3.730959
41.904762	8.000000	-117.359650	-17.475506	84.067940	-5.932053
49.523811	8.000000	-112.190811	-17.934093	83.887230	-6.112739
57.142857	8.000000	-108.020767	-17.272957	84.704994	-5.295015
64.761902	8.000000	-105.206871	-15.886917	85.760895	-4.239093
72.380951	8.000000	-103.420334	-14.232059	86.710358	-3.289636
80.000000	8.000000	-102.297440	-12.589256	87.464455	-2.535518
-80.000000	16.000000	-101.680008	-14.085575	-87.422325	2.577687
-72.380951	16.000000	-102.375191	-16.886139	-86.646324	3.353688
-64.761902	16.000000	-103.479065	-20.649757	-85.519600	4.480404
-57.142857	16.000000	-105.330093	-25.927780	-83.813194	6.186819
-49.523811	16.000000	-108.668808	-33.813553	-81.085045	8.914960
-41.904762	16.000000	-115.344955	-46.898357	-76.402077	13.597907
-34.285713	16.000000	-130.924850	-73.435860	-67.670929	22.329075
-26.666666	16.000000	-176.605408	-155.002747	-65.250992	24.749039
-19.047619	16.000000	-356.537506	9.036743	-35.975277	54.024723
-11.428572	16.000000	-95.779129	42.996429	25.344290	-64.655708
-3.809524	16.000000	-65.377800	5.230097	51.877518	-38.122482
3.809524	16.000000	-65.822586	-10.131977	72.354019	-17.645979
11.428572	16.000000	-79.267365	-12.071262	86.702438	-3.297601
19.047619	16.000000	-96.877220	-8.387321	-89.356247	0.643866
26.666668	16.000000	-109.833603	-5.776989	89.496315	-0.503688
34.285713	16.000000	-115.336502	-5.832516	87.221024	-2.778993
41.904762	16.000000	-115.225677	-7.423286	85.529099	-4.470901
49.523811	16.000000	-112.571877	-9.094872	84.835876	-5.164129
57.142857	16.000000	-109.486145	-10.115925	84.935059	-5.064942
64.761902	16.000000	-106.867821	-10.401154	85.463722	-4.536280
72.380951	16.000000	-104.904579	-10.143654	86.135414	-3.864616
80.000000	16.000000	-103.508919	-9.570610	86.790154	-3.209863
-80.000000	24.000000	-103.176773	-10.977726	-86.182671	3.817342
-72.380951	24.000000	-104.509300	-12.430550	-85.111320	4.888677
-64.761902	24.000000	-106.639221	-14.000301	-83.633202	6.366801
-57.142857	24.000000	-110.235901	-15.475010	-81.583214	8.416785
-49.523811	24.000000	-116.748550	-16.186234	-78.801811	11.198174
-41.904762	24.000000	-129.653229	-13.998005	-75.406075	14.593919

-34.285713 24.000000 -158.414230 -0.871140 -72.890335 17.109661
-26.666666 24.000000 -227.919571 69.953781 -79.265770 10.734232
-19.047619 24.000000 35.587402 254.188904 39.280941 -50.719055
-11.428572 24.000000 -23.262459 102.511406 83.729515 -6.270481
-3.809524 24.000000 -62.177254 43.521561 80.929359 -9.070638
3.809524 24.000000 -75.061783 18.555466 84.618698 -5.381298
11.428572 24.000000 -86.587982 8.573132 88.563065 -1.436990
19.047619 24.000000 -98.018494 5.205235 -89.840851 0.158653
26.666668 24.000000 -106.830780 3.668652 89.665894 -0.334208
34.285713 24.000000 -111.603157 1.907257 88.299690 -1.700338
41.904762 24.000000 -112.786148 -0.310062 86.987640 -3.012336
49.523811 24.000000 -111.756050 -2.523293 86.151810 -3.848180
57.142857 24.000000 -109.809738 -4.312901 85.844452 -4.155539
64.761902 24.000000 -107.757996 -5.518272 85.937584 -4.062408
72.380951 24.000000 -105.967697 -6.178829 86.264709 -3.735300
80.000000 24.000000 -104.537613 -6.420101 86.691414 -3.308581
-80.000000 32.000000 -104.856064 -7.006649 -85.818161 4.181865
-72.380951 32.000000 -106.726959 -6.915371 -84.935738 5.064270
-64.761902 32.000000 -109.545761 -6.150864 -83.917610 6.082375
-57.142857 32.000000 -113.878632 -4.020515 -82.885040 7.114951
-49.523811 32.000000 -120.573395 0.882069 -82.221146 7.778853
-41.904762 32.000000 -130.432373 11.438881 -82.923485 7.076523
-34.285713 32.000000 -141.119308 32.945599 -87.327217 2.672801
-26.666666 32.000000 -129.930756 66.112000 80.060547 -9.939452
-19.047619 32.000000 -45.858315 60.651131 60.541931 -29.458069
-11.428572 32.000000 -18.852898 36.230370 -84.871910 5.128099
-3.809524 32.000000 -57.446884 36.265793 -84.915802 5.084192
3.809524 32.000000 -76.677322 25.699776 -87.025414 2.974594
11.428572 32.000000 -88.653076 17.622795 -87.617943 2.382058
19.047619 32.000000 -97.809082 12.760468 -88.146957 1.853058
26.666668 32.000000 -104.567596 9.592140 -89.127274 0.872700
34.285713 32.000000 -108.687424 6.952549 89.613861 -0.386144
41.904762 32.000000 -110.392929 4.418507 88.412964 -1.587054
49.523811 32.000000 -110.354088 2.035271 87.502907 -2.497078
57.142857 32.000000 -109.330917 -0.023190 86.960762 -3.039265
64.761902 32.000000 -107.919693 -1.641449 86.750740 -3.249264
72.380951 32.000000 -106.485466 -2.803185 86.785759 -3.214270
80.000000 32.000000 -105.204758 -3.562046 86.973358 -3.026654
-80.000000 40.000000 -106.131752 -3.131760 -86.175262 3.824739
-72.380951 40.000000 -108.117706 -1.905231 -85.718323 4.281652
-64.761902 40.000000 -110.808884 0.283745 -85.403618 4.596376
-57.142857 40.000000 -114.327919 4.010071 -85.478233 4.521772
-49.523811 40.000000 -118.457062 10.071930 -86.407173 3.592835
-41.904762 40.000000 -121.712982 19.192123 -88.977119 1.022937
-34.285713 40.000000 -119.021942 30.381500 85.757759 -4.242226
-26.666666 40.000000 -99.748535 36.412964 77.530922 -12.469069
-19.047619 40.000000 -61.623619 25.315842 71.824776 -18.175232
-11.428572 40.000000 -42.291634 13.471277 87.043007 -2.956995
-3.809524 40.000000 -58.890182 18.865152 -83.179207 6.820786

```
3.809524 40.000000 -75.767448 19.744825 -83.607712 6.392283
11.428572 40.000000 -87.615578 17.337944 -85.033440 4.966572
19.047619 40.000000 -96.096420 14.481705 -86.400597 3.599394
26.666668 40.000000 -102.116470 11.870945 -87.754097 2.245883
34.285713 40.000000 -105.999557 9.443516 -89.082657 0.917348
41.904762 40.000000 -108.039635 7.104240 89.718994 -0.280845
49.523811 40.000000 -108.662888 4.883556 88.753952 -1.246043
57.142857 40.000000 -108.351662 2.879822 88.070961 -1.929066
64.761902 40.000000 -107.531387 1.180073 87.661285 -2.338732
72.380951 40.000000 -106.511353 -0.177158 87.478981 -2.521014
80.000000 40.000000 -105.482132 -1.201267 87.464149 -2.535827
-80.000000 48.000000 -106.725708 0.017651 -86.942528 3.057469
-72.380951 48.000000 -108.419464 1.776104 -86.910698 3.089306
-64.761902 48.000000 -110.398163 4.360157 -87.169617 2.830408
-57.142857 48.000000 -112.417709 7.995186 -87.943138 2.056893
-49.523811 48.000000 -113.740395 12.740334 -89.547325 0.452657
-41.904762 48.000000 -112.634422 18.043266 87.666260 -2.333714
-34.285713 48.000000 -106.001236 21.925583 83.587646 -6.412362
-26.666666 48.000000 -90.918762 20.756756 79.191055 -10.808940
-19.047619 48.000000 -70.668274 13.059025 78.273575 -11.726428
-11.428572 48.000000 -59.394157 6.837326 86.116371 -3.883656
-3.809524 48.000000 -64.980560 9.061069 -85.645645 4.354326
3.809524 48.000000 -76.264053 12.363239 -83.645645 6.354371
11.428572 48.000000 -86.296600 13.384399 -84.206520 5.793482
19.047619 48.000000 -94.019218 12.846497 -85.412086 4.587905
26.666668 48.000000 -99.644562 11.572384 -86.741302 3.258695
34.285713 48.000000 -103.464005 9.954952 -88.034103 1.965913
41.904762 48.000000 -105.762672 8.175262 -89.201012 0.798907
49.523811 48.000000 -106.859802 6.359859 89.821838 -0.178023
57.142857 48.000000 -107.094009 4.621891 89.067795 -0.932223
64.761902 48.000000 -106.777451 3.052128 88.537811 -1.462212
72.380951 48.000000 -106.160492 1.706562 88.207756 -1.792300
80.000000 48.000000 -105.420433 0.605164 88.039902 -1.960135
-80.000000 56.000000 -106.659195 2.222565 -87.844009 2.155993
-72.380951 56.000000 -107.844650 4.023190 -88.125153 1.874843
-64.761902 56.000000 -108.944305 6.343449 -88.725967 1.274036
-57.142857 56.000000 -109.574265 9.140774 -89.790833 0.209298
-49.523811 56.000000 -108.996994 12.124954 88.537041 -1.462995
-41.904762 56.000000 -106.029984 14.548347 86.211334 -3.788689
-34.285713 56.000000 -99.331444 15.114494 83.508461 -6.491523
-26.666666 56.000000 -88.629562 12.578617 81.435478 -8.564519
-19.047619 56.000000 -76.787186 7.587292 82.036858 -7.963140
-11.428572 56.000000 -69.941864 3.984432 86.757088 -3.242915
-3.809524 56.000000 -71.612320 4.654392 -87.678169 2.321801
3.809524 56.000000 -78.354736 7.300957 -84.963898 5.036081
11.428572 56.000000 -85.899994 9.311371 -84.566406 5.433609
19.047619 56.000000 -92.438065 10.121777 -85.214188 4.785820
26.666668 56.000000 -97.537903 9.992298 -86.240028 3.759996
34.285713 56.000000 -101.225677 9.250095 -87.351730 2.648291
```

```
41.904762 56.000000 -103.668091 8.137653 -88.409927 1.590102
49.523811 56.000000 -105.087326 6.831223 -89.339752 0.660159
57.142857 56.000000 -105.725655 5.466202 89.896767 -0.103233
64.761902 56.000000 -105.815384 4.145206 89.310417 -0.689479
72.380951 56.000000 -105.555450 2.938663 88.891975 -1.107973
80.000000 56.000000 -105.100121 1.886566 88.619576 -1.380416
-80.000000 64.000000 -106.111221 3.549789 -88.702415 1.297581
-72.380951 64.000000 -106.757965 5.112698 -89.165497 0.834551
-64.761902 64.000000 -107.089722 6.905987 -89.907661 0.091906
-57.142857 64.000000 -106.745026 8.769833 89.004478 -0.995610
-49.523811 64.000000 -105.187119 10.364201 87.549057 -2.450954
-41.904762 64.000000 -101.778854 11.136066 85.826820 -4.173183
-34.285713 64.000000 -96.100891 10.449879 84.188164 -5.811834
-26.666666 64.000000 -88.592979 8.051140 83.364296 -6.635707
-19.047619 64.000000 -81.175095 4.762737 84.348557 -5.651441
-11.428572 64.000000 -76.789307 2.510189 87.461983 -2.537993
-3.809524 64.000000 -77.128418 2.616543 -88.800522 1.199446
3.809524 64.000000 -81.074898 4.345413 -86.320183 3.679814
11.428572 64.000000 -86.406021 6.238567 -85.413704 4.586287
19.047619 64.000000 -91.605957 7.523079 -85.535286 4.464722
26.666668 64.000000 -96.007217 8.065678 -86.174889 3.825114
34.285713 64.000000 -99.417206 7.987991 -87.023613 2.976400
41.904762 64.000000 -101.860794 7.462151 -87.910164 2.089844
49.523811 64.000000 -103.459152 6.646641 -88.739700 1.260233
57.142857 64.000000 -104.373199 5.673500 -89.461487 0.538631
64.761902 64.000000 -104.770691 4.646984 89.946907 -0.052004
72.380951 64.000000 -104.805389 3.644093 89.488625 -0.511283
80.000000 64.000000 -104.605560 2.716286 89.154236 -0.845732
-80.000000 72.000000 -105.289963 4.193562 -89.430122 0.570014
-72.380951 72.000000 -105.476959 5.414616 -89.967781 0.031109
-64.761902 72.000000 -105.252640 6.663788 89.282738 -0.717180
-57.142857 72.000000 -104.342476 7.766518 88.310066 -1.689931
-49.523811 72.000000 -102.417946 8.453789 87.157143 -2.842885
-41.904762 72.000000 -99.197433 8.398201 85.969048 -4.030941
-34.285713 72.000000 -94.660400 7.350773 85.046280 -4.953732
-26.666666 72.000000 -89.334885 5.392849 84.851662 -5.148345
-19.047619 72.000000 -84.438301 3.159126 85.843002 -4.157030
-11.428572 72.000000 -81.496933 1.668789 88.023895 -1.976098
-3.809524 72.000000 -81.392960 1.588436 -89.381271 0.618756
3.809524 72.000000 -83.746849 2.673721 -87.361305 2.638735
11.428572 72.000000 -87.432053 4.165951 -86.314781 3.685210
19.047619 72.000000 -91.412186 5.455875 -86.091614 3.908406
26.666668 72.000000 -95.056885 6.278240 -86.392838 3.607162
34.285713 72.000000 -98.078079 6.602978 -86.970818 3.029162
41.904762 72.000000 -100.397110 6.505527 -87.659752 2.340224
49.523811 72.000000 -102.048950 6.092960 -88.356705 1.643240
57.142857 72.000000 -103.124451 5.471451 -89.000961 0.999029
64.761902 72.000000 -103.736412 4.733376 -89.560059 0.439796
72.380951 72.000000 -103.998276 3.952724 89.979355 -0.020372
```

```
80.000000 72.000000 -104.011932 3.184204 89.617805 -0.382354
-80.000000 80.000000 -104.365959 4.365952 90.000000 -0.000000
-72.380951 80.000000 -104.210892 5.247143 89.460876 -0.539087
-64.761902 80.000000 -103.637115 6.041679 88.778137 -1.221840
-57.142857 80.000000 -102.462128 6.605434 87.972176 -2.027856
-49.523811 80.000000 -100.511307 6.759743 87.110741 -2.889236
-41.904762 80.000000 -97.698311 6.338249 86.332497 -3.667494
-34.285713 80.000000 -94.145889 5.280521 85.859436 -4.140573
-26.666666 80.000000 -90.301102 3.749828 85.965240 -4.034750
-19.047619 80.000000 -86.932167 2.184975 86.852074 -3.147911
-11.428572 80.000000 -84.884613 1.155590 88.446991 -1.553069
-3.809524 80.000000 -84.650299 1.024418 -89.685028 0.314715
3.809524 80.000000 -86.099915 1.711140 -88.093925 1.906085
11.428572 80.000000 -88.650421 2.819172 -87.093826 2.906186
19.047619 80.000000 -91.642654 3.936253 -86.695053 3.304920
26.666668 80.000000 -94.579025 4.803886 -86.752518 3.247481
34.285713 80.000000 -97.167854 5.322533 -87.098602 2.901393
41.904762 80.000000 -99.279053 5.495464 -87.597191 2.402814
49.523811 80.000000 -100.887894 5.376720 -88.151894 1.848124
57.142857 80.000000 -102.030724 5.038261 -88.699524 1.300447
64.761902 80.000000 -102.774986 4.552246 -89.201714 0.798259
72.380951 80.000000 -103.199005 3.982277 -89.637886 0.361925
80.000000 80.000000 -103.379738 3.379753 90.000000 -86.693748
19.500000 -19.500000 -50.000484 -0.000002 -44.999996 45.000004
18.500000 -18.500000 -50.000484 -0.000002 -44.999996 45.000004
17.500000 -17.500000 -49.999531 -0.000002 -44.999996 45.000004
16.500000 -16.500000 -49.999531 -0.000002 -44.999996 45.000004
15.500000 -15.500000 -49.999920 -0.000002 -44.999996 45.000004
14.500000 -14.500000 -49.999920 -0.000002 -44.999996 45.000004
13.500000 -13.500000 -49.999973 -0.000002 -44.999996 45.000004
12.500000 -12.500000 -49.999973 -0.000002 -44.999996 45.000004
11.500000 -11.500000 -49.998554 -0.000002 -44.999996 45.000004
10.500000 -10.500000 -49.998554 -0.000002 -44.999996 45.000004
9.500000 -9.500000 -50.002243 -0.000002 -44.999996 45.000004
8.500000 -8.500000 -50.002243 -0.000002 -44.999996 45.000004
7.500000 -7.500000 -49.999512 -0.000002 -44.999996 45.000004
6.500000 -6.500000 -49.999512 -0.000002 -44.999996 45.000004
5.500000 -5.500000 -49.999680 -0.000002 -44.999996 45.000004
4.500000 -4.500000 -49.999680 -0.000002 -44.999996 45.000004
3.500000 -3.500000 -49.999573 -0.000002 -44.999996 45.000004
2.500000 -2.500000 -49.999573 -0.000002 -44.999996 45.000004
1.500000 -1.500000 -50.001171 -0.000002 -44.999996 45.000004
0.500000 -0.500000 -50.001171 -0.000002 -44.999996 45.000004
-0.500000 0.500000 -49.999306 -0.000002 -44.999996 45.000004
-1.500000 1.500000 -49.999306 -0.000002 -44.999996 45.000004
-2.500000 2.500000 -49.999466 -0.000002 -44.999996 45.000004
-3.500000 3.500000 -49.999466 -0.000002 -44.999996 45.000004
-4.500000 4.500000 -50.000488 -0.000002 -44.999996 45.000004
-5.500000 5.500000 -50.000488 -0.000002 -44.999996 45.000004
```

```
-6.500000 6.500000 -49.999836 -0.000002 -44.999996 45.000004
-7.500000 7.500000 -49.999836 -0.000002 -44.999996 45.000004
-8.500000 8.500000 -49.999832 -0.000002 -44.999996 45.000004
-9.500000 9.500000 -49.999832 -0.000002 -44.999996 45.000004
-10.500000 10.500000 -50.000652 -0.000002 -44.999996 45.000004
-11.500000 11.500000 -50.000652 -0.000002 -44.999996 45.000004
-12.500000 12.500000 -49.999908 -0.000002 -44.999996 45.000004
-13.500000 13.500000 -49.999908 -0.000002 -44.999996 45.000004
-14.500000 14.500000 -49.998547 -0.000002 -44.999996 45.000004
-15.500000 15.500000 -49.998547 -0.000002 -44.999996 45.000004
-16.500000 16.500000 -50.001896 -0.000002 -44.999996 45.000004
-17.500000 17.500000 -50.001896 -0.000002 -44.999996 45.000004
-18.500000 18.500000 -50.000092 -0.000002 -44.999996 45.000004
-19.500000 19.500000 -50.000092 -0.000002 -44.999996 45.000004
XPos Ypos Sigma1 Sigma3 Sig1dir Sig3dir
```

Displacements and Stresses at Specified Points in the Body.

Point	X Coord	Y Coord	UX	UY	Sigma-XX	Sigma-YY	Sigma-XY
1	-80.00	-80.00	-0.002583	0.002583	3.38	-103.38	0.00
2	-72.38	-80.00	-0.002199	0.003276	3.98	-103.19	0.68
3	-64.76	-80.00	-0.001775	0.004104	4.53	-102.75	1.50
4	-57.14	-80.00	-0.001324	0.005075	4.98	-101.98	2.43
5	-49.52	-80.00	-0.000867	0.006189	5.27	-100.78	3.48
6	-41.90	-80.00	-0.000432	0.007430	5.31	-99.09	4.39
7	-34.29	-80.00	-0.000054	0.008763	5.06	-96.91	5.18
8	-26.67	-80.00	0.000231	0.010126	4.48	-94.26	5.62
9	-19.05	-80.00	0.000386	0.011428	3.62	-91.33	5.50
10	-11.43	-80.00	0.000388	0.012550	2.58	-88.42	4.63
11	-3.81	-80.00	0.000239	0.013353	1.61	-86.00	2.92
12	3.81	-80.00	-0.000025	0.013707	1.02	-84.65	0.47
13	11.43	-80.00	-0.000326	0.013522	1.09	-84.82	-2.33
14	19.05	-80.00	-0.000566	0.012790	1.92	-86.66	-4.89
15	26.67	-80.00	-0.000649	0.011597	3.28	-89.84	-6.62
16	34.29	-80.00	-0.000525	0.010103	4.76	-93.63	-7.16
17	41.90	-80.00	-0.000196	0.008494	5.91	-97.27	-6.64
18	49.52	-80.00	0.000287	0.006931	6.49	-100.24	-5.40
19	57.14	-80.00	0.000856	0.005520	6.47	-102.33	-3.86
20	64.76	-80.00	0.001445	0.004310	5.99	-103.59	-2.34
21	72.38	-80.00	0.002005	0.003311	5.24	-104.20	-1.03
22	80.00	-80.00	0.002506	0.002506	4.37	-104.37	-0.00
23	-80.00	-72.00	-0.003314	0.002178	3.18	-104.01	-0.72
24	-72.38	-72.00	-0.002922	0.002862	3.95	-104.00	-0.04
25	-64.76	-72.00	-0.002469	0.003704	4.73	-103.73	0.83
26	-57.14	-72.00	-0.001966	0.004721	5.44	-103.09	1.89
27	-49.52	-72.00	-0.001431	0.005925	6.00	-101.96	3.10
28	-41.90	-72.00	-0.000895	0.007310	6.33	-100.22	4.36
29	-34.29	-72.00	-0.000398	0.008846	6.31	-97.79	5.52
30	-26.67	-72.00	0.000012	0.010472	5.88	-94.66	6.36
31	-19.05	-72.00	0.000284	0.012081	5.01	-90.96	6.59
32	-11.43	-72.00	0.000372	0.013524	3.79	-87.05	5.88
33	-3.81	-72.00	0.000262	0.014607	2.49	-83.56	3.97
34	3.81	-72.00	-0.000016	0.015130	1.58	-81.38	0.90
35	11.43	-72.00	-0.000363	0.014935	1.57	-81.40	-2.87
36	19.05	-72.00	-0.000631	0.013989	2.70	-83.98	-6.33
37	26.67	-72.00	-0.000678	0.012424	4.63	-88.57	-8.47
38	34.29	-72.00	-0.000433	0.010497	6.59	-93.90	-8.78
39	41.90	-72.00	0.000075	0.008497	7.87	-98.67	-7.54
40	49.52	-72.00	0.000753	0.006643	8.18	-102.15	-5.49
41	57.14	-72.00	0.001490	0.005056	7.67	-104.24	-3.30
42	64.76	-72.00	0.002199	0.003767	6.65	-105.24	-1.40
43	72.38	-72.00	0.002828	0.002755	5.41	-105.48	0.06
44	80.00	-72.00	0.003356	0.001978	4.18	-105.28	1.09

45	-80.00	-64.00	-0.004194	0.001731	2.69	-104.58	-1.58
46	-72.38	-64.00	-0.003818	0.002384	3.64	-104.80	-0.97
47	-64.76	-64.00	-0.003359	0.003215	4.65	-104.77	-0.10
48	-57.14	-64.00	-0.002822	0.004256	5.66	-104.36	1.03
49	-49.52	-64.00	-0.002220	0.005531	6.59	-103.41	2.42
50	-41.90	-64.00	-0.001583	0.007051	7.32	-101.72	3.98
51	-34.29	-64.00	-0.000952	0.008800	7.70	-99.13	5.57
52	-26.67	-64.00	-0.000385	0.010722	7.60	-95.54	6.93
53	-19.05	-64.00	0.000049	0.012707	6.92	-91.01	7.69
54	-11.43	-64.00	0.000281	0.014572	5.65	-85.81	7.38
55	-3.81	-64.00	0.000260	0.016060	3.99	-80.72	5.47
56	3.81	-64.00	-0.000005	0.016856	2.58	-77.09	1.67
57	11.43	-64.00	-0.000398	0.016679	2.35	-76.63	-3.51
58	19.05	-64.00	-0.000696	0.015432	3.93	-80.34	-8.42
59	26.67	-64.00	-0.000669	0.013317	6.76	-87.30	-11.09
60	34.29	-64.00	-0.000220	0.010777	9.36	-95.01	-10.73
61	41.90	-64.00	0.000565	0.008275	10.54	-101.18	-8.20
62	49.52	-64.00	0.001504	0.006105	10.15	-104.98	-4.94
63	57.14	-64.00	0.002432	0.004373	8.73	-106.71	-2.01
64	64.76	-64.00	0.003248	0.003058	6.91	-107.09	0.18
65	72.38	-64.00	0.003912	0.002089	5.09	-106.73	1.63
66	80.00	-64.00	0.004421	0.001387	3.49	-106.05	2.48
67	-80.00	-56.00	-0.005233	0.001255	1.82	-105.04	-2.58
68	-72.38	-56.00	-0.004909	0.001852	2.90	-105.51	-2.10
69	-64.76	-56.00	-0.004480	0.002644	4.13	-105.80	-1.32
70	-57.14	-56.00	-0.003943	0.003675	5.47	-105.73	-0.20
71	-49.52	-56.00	-0.003303	0.004991	6.82	-105.07	1.29
72	-41.90	-56.00	-0.002583	0.006624	8.05	-103.58	3.10
73	-34.29	-56.00	-0.001821	0.008580	9.01	-100.99	5.10
74	-26.67	-56.00	-0.001078	0.010821	9.53	-97.08	7.04
75	-19.05	-56.00	-0.000433	0.013244	9.41	-91.72	8.58
76	-11.43	-56.00	0.000019	0.015650	8.46	-85.05	8.98
77	-3.81	-56.00	0.000180	0.017715	6.64	-77.69	7.49
78	3.81	-56.00	-0.000002	0.018969	4.53	-71.49	3.09
79	11.43	-56.00	-0.000424	0.018892	3.75	-69.71	-4.18
80	19.05	-56.00	-0.000749	0.017206	5.97	-75.17	-11.58
81	26.67	-56.00	-0.000570	0.014235	10.33	-86.38	-14.90
82	34.29	-56.00	0.000240	0.010805	13.65	-97.87	-12.86
83	41.90	-56.00	0.001447	0.007684	14.02	-105.50	-7.95
84	49.52	-56.00	0.002713	0.005224	12.05	-108.92	-3.09
85	57.14	-56.00	0.003818	0.003438	9.14	-109.57	0.43
86	64.76	-56.00	0.004680	0.002194	6.29	-108.89	2.56
87	72.38	-56.00	0.005301	0.001344	3.90	-107.72	3.66
88	80.00	-56.00	0.005719	0.000768	2.07	-106.51	4.09
89	-80.00	-48.00	-0.006427	0.000778	0.48	-105.30	-3.62
90	-72.38	-48.00	-0.006203	0.001292	1.60	-106.05	-3.37
91	-64.76	-48.00	-0.005857	0.002009	2.98	-106.71	-2.80
92	-57.14	-48.00	-0.005376	0.002991	4.59	-107.06	-1.82
93	-49.52	-48.00	-0.004753	0.004304	6.36	-106.86	-0.35

94	-41.90	-48.00	-0.003998	0.006010	8.15	-105.74	1.59
95	-34.29	-48.00	-0.003144	0.008149	9.82	-103.33	3.89
96	-26.67	-48.00	-0.002243	0.010710	11.21	-99.29	6.31
97	-19.05	-48.00	-0.001368	0.013612	12.16	-93.34	8.52
98	-11.43	-48.00	-0.000617	0.016667	12.37	-85.28	10.01
99	-3.81	-48.00	-0.000125	0.019523	11.28	-75.18	9.75
100	3.81	-48.00	-0.000053	0.021551	8.63	-64.55	5.61
101	11.43	-48.00	-0.000420	0.021794	6.53	-59.09	-4.48
102	19.05	-48.00	-0.000754	0.019461	9.60	-67.21	-16.66
103	26.67	-48.00	-0.000253	0.015055	16.83	-86.99	-20.57
104	34.29	-48.00	0.001226	0.010307	20.33	-104.41	-14.20
105	41.90	-48.00	0.003050	0.006516	17.83	-112.42	-5.32
106	49.52	-48.00	0.004643	0.003924	12.73	-113.73	1.00
107	57.14	-48.00	0.005806	0.002266	7.84	-112.26	4.32
108	64.76	-48.00	0.006565	0.001229	4.08	-110.12	5.66
109	72.38	-48.00	0.007011	0.000584	1.46	-108.10	5.93
110	80.00	-48.00	0.007234	0.000182	-0.29	-106.42	5.69
111	-80.00	-40.00	-0.007756	0.000331	-1.41	-105.28	-4.61
112	-72.38	-40.00	-0.007688	0.000740	-0.38	-106.31	-4.67
113	-64.76	-40.00	-0.007495	0.001348	1.00	-107.35	-4.43
114	-57.14	-40.00	-0.007150	0.002232	2.75	-108.23	-3.74
115	-49.52	-40.00	-0.006630	0.003488	4.83	-108.61	-2.47
116	-41.90	-40.00	-0.005931	0.005215	7.10	-108.04	-0.56
117	-34.29	-40.00	-0.005076	0.007495	9.41	-105.97	1.85
118	-26.67	-40.00	-0.004108	0.010356	11.70	-101.94	4.46
119	-19.05	-40.00	-0.003078	0.013742	14.05	-95.66	6.93
120	-11.43	-40.00	-0.002036	0.017496	16.55	-86.83	9.05
121	-3.81	-40.00	-0.001059	0.021321	18.56	-74.58	10.57
122	3.81	-40.00	-0.000372	0.024579	17.77	-57.79	9.17
123	11.43	-40.00	-0.000357	0.025744	13.32	-42.14	-2.87
124	19.05	-40.00	-0.000620	0.022490	16.86	-53.16	-25.77
125	26.67	-40.00	0.000651	0.015383	30.07	-93.40	-28.73
126	34.29	-40.00	0.003414	0.008753	29.56	-118.20	-11.02
127	41.90	-40.00	0.005955	0.004562	19.15	-121.67	2.51
128	49.52	-40.00	0.007609	0.002242	9.57	-117.95	8.04
129	57.14	-40.00	0.008505	0.000984	3.27	-113.59	9.30
130	64.76	-40.00	0.008900	0.000294	-0.43	-110.10	8.87
131	72.38	-40.00	0.008989	-0.000086	-2.50	-107.53	7.91
132	80.00	-40.00	0.008898	-0.000295	-3.59	-105.67	6.86
133	-80.00	-32.00	-0.009172	-0.000043	-3.85	-104.92	-5.36
134	-72.38	-32.00	-0.009323	0.000245	-3.13	-106.16	-5.80
135	-64.76	-32.00	-0.009366	0.000712	-1.98	-107.58	-6.04
136	-57.14	-32.00	-0.009257	0.001451	-0.33	-109.02	-5.79
137	-49.52	-32.00	-0.008960	0.002588	1.82	-110.14	-4.89
138	-41.90	-32.00	-0.008452	0.004271	4.33	-110.30	-3.18
139	-34.29	-32.00	-0.007749	0.006645	6.95	-108.68	-0.78
140	-26.67	-32.00	-0.006902	0.009789	9.57	-104.54	1.74
141	-19.05	-32.00	-0.005965	0.013662	12.64	-97.69	3.57
142	-11.43	-32.00	-0.004916	0.018092	17.44	-88.47	4.41

143	-3.81	-32.00	-0.003612	0.022854	25.42	-76.40	5.31
144	3.81	-32.00	-0.001913	0.027649	35.53	-56.71	8.27
145	11.43	-32.00	-0.000361	0.031155	35.79	-18.41	4.90
146	19.05	-32.00	-0.000045	0.027063	34.89	-20.10	-45.61
147	26.67	-32.00	0.003547	0.013826	60.27	-124.09	-33.33
148	34.29	-32.00	0.008446	0.005394	32.57	-140.74	8.11
149	41.90	-32.00	0.010917	0.001946	9.29	-128.28	17.34
150	49.52	-32.00	0.011752	0.000511	-1.34	-118.35	16.29
151	57.14	-32.00	0.011809	-0.000127	-5.71	-112.19	13.50
152	64.76	-32.00	0.011516	-0.000416	-7.31	-108.38	10.89
153	72.38	-32.00	0.011071	-0.000541	-7.69	-105.95	8.78
154	80.00	-32.00	0.010571	-0.000585	-7.53	-104.34	7.12
155	-80.00	-24.00	-0.010594	-0.000301	-6.75	-104.21	-5.65
156	-72.38	-24.00	-0.011024	-0.000137	-6.60	-105.54	-6.49
157	-64.76	-24.00	-0.011387	0.000172	-6.03	-107.24	-7.22
158	-57.14	-24.00	-0.011633	0.000724	-4.87	-109.26	-7.62
159	-49.52	-24.00	-0.011705	0.001674	-3.02	-111.26	-7.31
160	-41.90	-24.00	-0.011559	0.003236	-0.62	-112.48	-5.90
161	-34.29	-24.00	-0.011203	0.005658	1.81	-111.50	-3.37
162	-26.67	-24.00	-0.010737	0.009117	3.66	-106.83	-0.64
163	-19.05	-24.00	-0.010308	0.013583	5.20	-98.02	0.29
164	-11.43	-24.00	-0.009958	0.018762	8.51	-86.53	-2.39
165	-3.81	-24.00	-0.009429	0.024289	17.73	-74.24	-8.74
166	3.81	-24.00	-0.007979	0.030086	40.89	-59.55	-16.46
167	11.43	-24.00	-0.003914	0.036716	101.01	-21.76	-13.66
168	19.05	-24.00	0.002545	0.037292	123.21	166.56	-107.13
169	26.67	-24.00	0.015945	0.005956	59.62	-217.59	54.51
170	34.29	-24.00	0.018558	0.000967	-14.51	-144.78	44.30
171	41.90	-24.00	0.017757	-0.000221	-21.34	-122.31	28.20
172	49.52	-24.00	0.016470	-0.000602	-19.98	-112.96	19.16
173	57.14	-24.00	0.015189	-0.000715	-17.51	-108.21	13.72
174	64.76	-24.00	0.014020	-0.000724	-15.14	-105.50	10.21
175	72.38	-24.00	0.012979	-0.000691	-13.10	-103.84	7.82
176	80.00	-24.00	0.012058	-0.000642	-11.39	-102.77	6.12
177	-80.00	-16.00	-0.011906	-0.000406	-9.87	-103.21	-5.25
178	-72.38	-16.00	-0.012651	-0.000348	-10.57	-104.47	-6.37
179	-64.76	-16.00	-0.013406	-0.000194	-11.00	-106.26	-7.61
180	-57.14	-16.00	-0.014123	0.000147	-10.89	-108.71	-8.74
181	-49.52	-16.00	-0.014730	0.000842	-9.93	-111.73	-9.28
182	-41.90	-16.00	-0.015145	0.002179	-8.08	-114.57	-8.38
183	-34.29	-16.00	-0.015344	0.004573	-6.09	-115.08	-5.30
184	-26.67	-16.00	-0.015487	0.008435	-5.79	-109.83	-0.91
185	-19.05	-16.00	-0.015953	0.013832	-8.40	-96.87	0.99
186	-11.43	-16.00	-0.017098	0.020242	-12.29	-79.05	-3.86
187	-3.81	-16.00	-0.018983	0.026957	-15.25	-60.71	-16.09
188	3.81	-16.00	-0.021658	0.033974	-21.68	-38.47	-34.29
189	11.43	-16.00	-0.026845	0.043618	-70.35	17.57	-53.69
190	19.05	-16.00	0.047445	-0.016986	-230.38	-117.12	173.79
191	26.67	-16.00	0.036670	0.000580	-158.79	-172.82	8.21

192	34.29	-16.00	0.028250	-0.000342	-81.73	-122.63	20.20
193	41.90	-16.00	0.023397	-0.000629	-50.68	-111.56	15.64
194	49.52	-16.00	0.020144	-0.000680	-35.61	-106.87	11.46
195	57.14	-16.00	0.017749	-0.000652	-26.85	-104.41	8.51
196	64.76	-16.00	0.015886	-0.000599	-21.16	-102.97	6.45
197	72.38	-16.00	0.014385	-0.000540	-17.18	-102.08	4.99
198	80.00	-16.00	0.013146	-0.000484	-14.26	-101.50	3.94
199	-80.00	-8.00	-0.012959	-0.000339	-12.76	-102.12	-3.96
200	-72.38	-8.00	-0.014012	-0.000347	-14.53	-103.13	-5.11
201	-64.76	-8.00	-0.015183	-0.000312	-16.37	-104.72	-6.58
202	-57.14	-8.00	-0.016451	-0.000170	-18.05	-107.25	-8.34
203	-49.52	-8.00	-0.017752	0.000222	-19.00	-111.12	-9.98
204	-41.90	-8.00	-0.018964	0.001185	-18.54	-116.29	-10.27
205	-34.29	-8.00	-0.019952	0.003351	-16.84	-120.46	-6.79
206	-26.67	-8.00	-0.020828	0.007640	-17.25	-116.54	1.65
207	-19.05	-8.00	-0.022290	0.014540	-24.63	-97.27	8.65
208	-11.43	-8.00	-0.025201	0.023094	-36.39	-69.54	4.71
209	-3.81	-8.00	-0.029696	0.031811	-47.80	-47.90	-5.58
210	3.81	-8.00	-0.035731	0.040947	-66.00	-36.48	-4.80
211	11.43	-8.00	0.043660	-0.034263	-79.49	-19.33	10.60
212	19.05	-8.00	0.035938	-0.016072	-69.27	-116.64	52.49
213	26.67	-8.00	0.032544	-0.003358	-57.38	-163.24	0.95
214	34.29	-8.00	0.028705	-0.000581	-60.32	-129.37	-8.55
215	41.90	-8.00	0.024681	-0.000238	-48.98	-112.63	-3.32
216	49.52	-8.00	0.021372	-0.000217	-37.63	-106.25	-0.38
217	57.14	-8.00	0.018768	-0.000221	-29.18	-103.51	0.76
218	64.76	-8.00	0.016705	-0.000217	-23.13	-102.15	1.11
219	72.38	-8.00	0.015042	-0.000206	-18.73	-101.41	1.14
220	80.00	-8.00	0.013675	-0.000192	-15.47	-100.97	1.06
221	-80.00	0.00	-0.013593	-0.000118	-14.84	-101.25	-1.77
222	-72.38	0.00	-0.014880	-0.000138	-17.60	-101.91	-2.52
223	-64.76	0.00	-0.016398	-0.000153	-21.00	-103.05	-3.66
224	-57.14	0.00	-0.018184	-0.000138	-25.07	-105.11	-5.38
225	-49.52	0.00	-0.020251	-0.000022	-29.47	-108.99	-7.79
226	-41.90	0.00	-0.022517	0.000427	-32.87	-116.21	-10.26
227	-34.29	0.00	-0.024701	0.001922	-32.72	-127.37	-8.95
228	-26.67	0.00	-0.026497	0.006252	-30.75	-132.11	4.93
229	-19.05	0.00	-0.028815	0.015442	-41.52	-102.82	23.83
230	-11.43	0.00	-0.033481	0.027440	-57.32	-56.51	16.47
231	-3.81	0.00	-0.039644	0.038050	-57.82	-40.53	0.74
232	3.81	0.00	0.039644	-0.038050	-57.82	-40.53	0.74
233	11.43	0.00	0.033481	-0.027440	-57.32	-56.51	16.47
234	19.05	0.00	0.028815	-0.015442	-41.52	-102.82	23.83
235	26.67	0.00	0.026497	-0.006252	-30.75	-132.11	4.93
236	34.29	0.00	0.024701	-0.001922	-32.72	-127.37	-8.95
237	41.90	0.00	0.022517	-0.000427	-32.87	-116.21	-10.26
238	49.52	0.00	0.020251	0.000022	-29.47	-108.99	-7.79
239	57.14	0.00	0.018184	0.000138	-25.07	-105.11	-5.38
240	64.76	0.00	0.016398	0.000153	-21.00	-103.05	-3.66

241	72.38	0.00	0.014880	0.000138	-17.60	-101.91	-2.52
242	80.00	0.00	0.013593	0.000118	-14.84	-101.25	-1.77
243	-80.00	8.00	-0.013675	0.000192	-15.47	-100.97	1.06
244	-72.38	8.00	-0.015042	0.000206	-18.73	-101.41	1.14
245	-64.76	8.00	-0.016705	0.000217	-23.13	-102.15	1.12
246	-57.14	8.00	-0.018768	0.000221	-29.18	-103.51	0.76
247	-49.52	8.00	-0.021372	0.000217	-37.63	-106.25	-0.38
248	-41.90	8.00	-0.024681	0.000238	-48.97	-112.63	-3.33
249	-34.29	8.00	-0.028705	0.000581	-60.32	-129.37	-8.55
250	-26.67	8.00	-0.032544	0.003358	-57.38	-163.24	0.95
251	-19.05	8.00	-0.035938	0.016072	-69.27	-116.64	52.49
252	-11.43	8.00	-0.043660	0.034263	-79.49	-19.33	10.60
253	-3.81	8.00	0.035731	-0.040947	-66.00	-36.48	-4.80
254	3.81	8.00	0.029696	-0.031811	-47.80	-47.90	-5.58
255	11.43	8.00	0.025201	-0.023094	-36.39	-69.54	4.72
256	19.05	8.00	0.022290	-0.014540	-24.63	-97.27	8.65
257	26.67	8.00	0.020828	-0.007640	-17.25	-116.54	1.65
258	34.29	8.00	0.019952	-0.003351	-16.84	-120.46	-6.79
259	41.90	8.00	0.018964	-0.001185	-18.54	-116.29	-10.27
260	49.52	8.00	0.017752	-0.000222	-19.00	-111.12	-9.98
261	57.14	8.00	0.016451	0.000170	-18.05	-107.25	-8.34
262	64.76	8.00	0.015183	0.000312	-16.37	-104.72	-6.58
263	72.38	8.00	0.014012	0.000347	-14.53	-103.13	-5.11
264	80.00	8.00	0.012959	0.000339	-12.76	-102.12	-3.96
265	-80.00	16.00	-0.013146	0.000484	-14.26	-101.50	3.94
266	-72.38	16.00	-0.014385	0.000540	-17.18	-102.08	4.99
267	-64.76	16.00	-0.015886	0.000599	-21.16	-102.97	6.45
268	-57.14	16.00	-0.017749	0.000652	-26.85	-104.41	8.51
269	-49.52	16.00	-0.020144	0.000680	-35.61	-106.87	11.46
270	-41.90	16.00	-0.023397	0.000629	-50.68	-111.56	15.64
271	-34.29	16.00	-0.028250	0.000342	-81.73	-122.63	20.20
272	-26.67	16.00	-0.036670	-0.000580	-158.79	-172.82	8.21
273	-19.05	16.00	-0.047445	0.016986	-230.38	-117.12	173.79
274	-11.43	16.00	0.026845	-0.043618	-70.35	17.57	-53.69
275	-3.81	16.00	0.021658	-0.033974	-21.68	-38.47	-34.29
276	3.81	16.00	0.018983	-0.026957	-15.25	-60.71	-16.09
277	11.43	16.00	0.017098	-0.020242	-12.29	-79.05	-3.86
278	19.05	16.00	0.015953	-0.013832	-8.40	-96.87	0.99
279	26.67	16.00	0.015487	-0.008435	-5.79	-109.83	-0.91
280	34.29	16.00	0.015344	-0.004573	-6.09	-115.08	-5.30
281	41.90	16.00	0.015145	-0.002179	-8.08	-114.57	-8.38
282	49.52	16.00	0.014730	-0.000842	-9.93	-111.73	-9.28
283	57.14	16.00	0.014123	-0.000147	-10.89	-108.71	-8.74
284	64.76	16.00	0.013406	0.000194	-11.00	-106.26	-7.61
285	72.38	16.00	0.012651	0.000348	-10.57	-104.47	-6.37
286	80.00	16.00	0.011906	0.000406	-9.87	-103.21	-5.25
287	-80.00	24.00	-0.012058	0.000642	-11.39	-102.77	6.12
288	-72.38	24.00	-0.012979	0.000691	-13.10	-103.84	7.82
289	-64.76	24.00	-0.014020	0.000724	-15.14	-105.50	10.21

290	-57.14	24.00	-0.015189	0.000715	-17.51	-108.21	13.72
291	-49.52	24.00	-0.016470	0.000602	-19.98	-112.96	19.16
292	-41.90	24.00	-0.017757	0.000221	-21.34	-122.31	28.20
293	-34.29	24.00	-0.018558	-0.000967	-14.51	-144.78	44.30
294	-26.67	24.00	-0.015945	-0.005956	59.62	-217.59	54.51
295	-19.05	24.00	-0.002545	-0.037292	123.21	166.56	-107.13
296	-11.43	24.00	0.003914	-0.036716	101.01	-21.76	-13.66
297	-3.81	24.00	0.007979	-0.030086	40.89	-59.55	-16.46
298	3.81	24.00	0.009429	-0.024289	17.73	-74.24	-8.74
299	11.43	24.00	0.009958	-0.018762	8.51	-86.53	-2.39
300	19.05	24.00	0.010308	-0.013583	5.20	-98.02	0.29
301	26.67	24.00	0.010737	-0.009117	3.66	-106.83	-0.64
302	34.29	24.00	0.011203	-0.005658	1.81	-111.50	-3.37
303	41.90	24.00	0.011559	-0.003236	-0.62	-112.48	-5.90
304	49.52	24.00	0.011705	-0.001674	-3.02	-111.26	-7.31
305	57.14	24.00	0.011633	-0.000724	-4.87	-109.26	-7.62
306	64.76	24.00	0.011387	-0.000172	-6.03	-107.24	-7.22
307	72.38	24.00	0.011024	0.000137	-6.60	-105.54	-6.49
308	80.00	24.00	0.010594	0.000301	-6.75	-104.21	-5.65
309	-80.00	32.00	-0.010571	0.000585	-7.53	-104.34	7.12
310	-72.38	32.00	-0.011071	0.000541	-7.69	-105.95	8.78
311	-64.76	32.00	-0.011516	0.000416	-7.31	-108.38	10.89
312	-57.14	32.00	-0.011809	0.000127	-5.71	-112.19	13.50
313	-49.52	32.00	-0.011752	-0.000511	-1.34	-118.35	16.29
314	-41.90	32.00	-0.010917	-0.001946	9.29	-128.28	17.34
315	-34.29	32.00	-0.008446	-0.005394	32.57	-140.74	8.11
316	-26.67	32.00	-0.003547	-0.013826	60.27	-124.09	-33.33
317	-19.05	32.00	0.000045	-0.027063	34.89	-20.10	-45.61
318	-11.43	32.00	0.000361	-0.031156	35.79	-18.41	4.90
319	-3.81	32.00	0.001913	-0.027649	35.53	-56.71	8.27
320	3.81	32.00	0.003612	-0.022854	25.42	-76.40	5.31
321	11.43	32.00	0.004916	-0.018092	17.44	-88.47	4.41
322	19.05	32.00	0.005965	-0.013662	12.64	-97.69	3.57
323	26.67	32.00	0.006902	-0.009789	9.57	-104.54	1.74
324	34.29	32.00	0.007749	-0.006645	6.95	-108.68	-0.78
325	41.90	32.00	0.008452	-0.004271	4.33	-110.30	-3.18
326	49.52	32.00	0.008960	-0.002588	1.82	-110.14	-4.89
327	57.14	32.00	0.009257	-0.001451	-0.33	-109.02	-5.79
328	64.76	32.00	0.009366	-0.000712	-1.98	-107.58	-6.01
329	72.38	32.00	0.009323	-0.000245	-3.13	-106.16	-5.80
330	80.00	32.00	0.009172	0.000043	-3.85	-104.92	-5.36
331	-80.00	40.00	-0.008898	0.000295	-3.59	-105.67	6.86
332	-72.38	40.00	-0.008989	0.000086	-2.50	-107.53	7.91
333	-64.76	40.00	-0.008900	-0.000294	-0.43	-110.10	8.87
334	-57.14	40.00	-0.008505	-0.000984	3.27	-113.59	9.30
335	-49.52	40.00	-0.007609	-0.002242	9.57	-117.95	8.04
336	-41.90	40.00	-0.005955	-0.004562	19.15	-121.67	2.51
337	-34.29	40.00	-0.003414	-0.008753	29.56	-118.20	-11.02
338	-26.67	40.00	-0.000651	-0.015383	30.07	-93.40	-28.71

339	-19.05	40.00	0.000620	-0.022490	16.86	-53.16	-25.77
340	-11.43	40.00	0.000357	-0.025744	13.32	-42.14	-2.87
341	-3.81	40.00	0.000372	-0.024579	17.77	-57.79	9.17
342	3.81	40.00	0.001059	-0.021321	18.56	-74.58	10.57
343	11.43	40.00	0.002036	-0.017496	16.55	-86.83	9.05
344	19.05	40.00	0.003078	-0.013742	14.05	-95.66	6.93
345	26.67	40.00	0.004108	-0.010356	11.70	-101.94	4.46
346	34.29	40.00	0.005076	-0.007495	9.41	-105.97	1.85
347	41.90	40.00	0.005931	-0.005215	7.10	-108.04	-0.56
348	49.52	40.00	0.006630	-0.003488	4.83	-108.61	-2.47
349	57.14	40.00	0.007150	-0.002232	2.75	-108.23	-3.74
350	64.76	40.00	0.007495	-0.001348	1.00	-107.35	-4.43
351	72.38	40.00	0.007688	-0.000740	-0.38	-106.31	-4.67
352	80.00	40.00	0.007756	-0.000331	-1.41	-105.28	-4.61
353	-80.00	48.00	-0.007234	-0.000182	-0.29	-106.42	5.69
354	-72.38	48.00	-0.007011	-0.000584	1.46	-108.10	5.93
355	-64.76	48.00	-0.006565	-0.001229	4.08	-110.12	5.66
356	-57.14	48.00	-0.005806	-0.002266	7.84	-112.26	4.32
357	-49.52	48.00	-0.004643	-0.003924	12.73	-113.73	1.00
358	-41.90	48.00	-0.003050	-0.006516	17.83	-112.42	-5.32
359	-34.29	48.00	-0.001226	-0.010307	20.33	-104.41	-14.20
360	-26.67	48.00	0.000253	-0.015055	16.83	-86.99	-20.57
361	-19.05	48.00	0.000754	-0.019461	9.60	-67.21	-16.66
362	-11.43	48.00	0.000420	-0.021794	6.53	-59.09	-4.48
363	-3.81	48.00	0.000053	-0.021550	8.63	-64.55	5.61
364	3.81	48.00	0.000125	-0.019523	11.28	-75.18	9.75
365	11.43	48.00	0.000617	-0.016667	12.37	-85.28	10.01
366	19.05	48.00	0.001368	-0.013612	12.16	-93.34	8.52
367	26.67	48.00	0.002243	-0.010710	11.21	-99.29	6.31
368	34.29	48.00	0.003144	-0.008149	9.82	-103.33	3.89
369	41.90	48.00	0.003998	-0.006010	8.15	-105.74	1.59
370	49.52	48.00	0.004753	-0.004304	6.36	-106.86	-0.35
371	57.14	48.00	0.005376	-0.002991	4.59	-107.06	-1.82
372	64.76	48.00	0.005857	-0.002009	2.98	-106.71	-2.80
373	72.38	48.00	0.006203	-0.001292	1.60	-106.05	-3.37
374	80.00	48.00	0.006427	-0.000778	0.48	-105.30	-3.62
375	-80.00	56.00	-0.005719	-0.000768	2.07	-106.51	4.09
376	-72.38	56.00	-0.005301	-0.001344	3.90	-107.72	3.66
377	-64.76	56.00	-0.004680	-0.002194	6.29	-108.89	2.56
378	-57.14	56.00	-0.003818	-0.003438	9.14	-109.57	0.43
379	-49.52	56.00	-0.002713	-0.005224	12.05	-108.92	-3.09
380	-41.90	56.00	-0.001447	-0.007684	14.02	-105.50	-7.95
381	-34.29	56.00	-0.000240	-0.010805	13.65	-97.87	-12.86
382	-26.67	56.00	0.000570	-0.014235	10.33	-86.38	-14.90
383	-19.05	56.00	0.000749	-0.017206	5.97	-75.17	-11.58
384	-11.43	56.00	0.000424	-0.018892	3.75	-69.71	-4.18
385	-3.81	56.00	0.000002	-0.018969	4.53	-71.49	3.09
386	3.81	56.00	-0.000180	-0.017715	6.64	-77.69	7.49
387	11.43	56.00	-0.000019	-0.015650	8.46	-85.05	8.98

388	19.05	56.00	0.000433	-0.013244	9.41	-91.72	8.53
389	26.67	56.00	0.001078	-0.010821	9.53	-97.08	7.04
390	34.29	56.00	0.001821	-0.008580	9.01	-100.99	5.10
391	41.90	56.00	0.002583	-0.006624	8.05	-103.58	3.10
392	49.52	56.00	0.003303	-0.004991	6.82	-105.07	1.29
393	57.14	56.00	0.003943	-0.003675	5.47	-105.73	-0.20
394	64.76	56.00	0.004480	-0.002644	4.13	-105.80	-1.32
395	72.38	56.00	0.004909	-0.001852	2.90	-105.51	-2.10
396	80.00	56.00	0.005233	-0.001255	1.82	-105.04	-2.58
397	-80.00	64.00	-0.004421	-0.001387	3.49	-106.05	2.48
398	-72.38	64.00	-0.003912	-0.002089	5.09	-106.73	1.63
399	-64.76	64.00	-0.003248	-0.003058	6.91	-107.09	0.18
400	-57.14	64.00	-0.002432	-0.004373	8.73	-106.71	-2.01
401	-49.52	64.00	-0.001504	-0.006105	10.15	-104.98	-4.94
402	-41.90	64.00	-0.000565	-0.008275	10.54	-101.18	-8.20
403	-34.29	64.00	0.000220	-0.010777	9.36	-95.01	-10.73
404	-26.67	64.00	0.000669	-0.013317	6.76	-87.30	-11.09
405	-19.05	64.00	0.000696	-0.015432	3.93	-80.34	-8.42
406	-11.43	64.00	0.000398	-0.016679	2.35	-76.63	-3.51
407	-3.81	64.00	0.000005	-0.016856	2.58	-77.09	1.67
408	3.81	64.00	-0.000260	-0.016060	3.99	-80.72	5.47
409	11.43	64.00	-0.000281	-0.014572	5.65	-85.81	7.38
410	19.05	64.00	-0.000049	-0.012707	6.92	-91.01	7.69
411	26.67	64.00	0.000385	-0.010722	7.60	-95.54	6.93
412	34.29	64.00	0.000952	-0.008800	7.70	-99.13	5.57
413	41.90	64.00	0.001583	-0.007051	7.32	-101.72	3.98
414	49.52	64.00	0.002220	-0.005531	6.59	-103.41	2.42
415	57.14	64.00	0.002822	-0.004256	5.66	-104.36	1.03
416	64.76	64.00	0.003359	-0.003215	4.65	-104.77	-0.10
417	72.38	64.00	0.003818	-0.002384	3.64	-104.80	-0.97
418	80.00	64.00	0.004194	-0.001731	2.69	-104.58	-1.58
419	-80.00	72.00	-0.003356	-0.001978	4.18	-105.28	1.09
420	-72.38	72.00	-0.002828	-0.002755	5.41	-105.48	0.06
421	-64.76	72.00	-0.002199	-0.003767	6.65	-105.24	-1.40
422	-57.14	72.00	-0.001490	-0.005056	7.67	-104.24	-3.30
423	-49.52	72.00	-0.000753	-0.006643	8.18	-102.15	-5.49
424	-41.90	72.00	-0.000075	-0.008497	7.87	-98.67	-7.54
425	-34.29	72.00	0.000433	-0.010497	6.59	-93.90	-8.78
426	-26.67	72.00	0.000678	-0.012424	4.63	-88.57	-8.47
427	-19.05	72.00	0.000631	-0.013989	2.70	-83.98	-6.33
428	-11.43	72.00	0.000363	-0.014935	1.57	-81.40	-2.87
429	-3.81	72.00	0.000016	-0.015130	1.58	-81.38	0.90
430	3.81	72.00	-0.000262	-0.014607	2.49	-83.56	3.97
431	11.43	72.00	-0.000372	-0.013524	3.79	-87.05	5.88
432	19.05	72.00	-0.000284	-0.012081	5.01	-90.96	6.59
433	26.67	72.00	-0.000012	-0.010472	5.88	-94.66	6.36
434	34.29	72.00	0.000398	-0.008846	6.31	-97.79	5.52
435	41.90	72.00	0.000895	-0.007310	6.33	-100.22	4.36
436	49.52	72.00	0.001431	-0.005925	6.00	-101.96	3.10

437	57.14	72.00	0.001966	-0.004721	5.44	-103.09	1.89
438	64.76	72.00	0.002469	-0.003704	4.73	-103.73	0.83
439	72.38	72.00	0.002922	-0.002862	3.95	-104.00	-0.04
440	80.00	72.00	0.003314	-0.002178	3.18	-104.01	-0.72
441	-80.00	80.00	-0.002506	-0.002506	4.37	-104.37	-0.00
442	-72.38	80.00	-0.002005	-0.003311	5.24	-104.20	-1.03
443	-64.76	80.00	-0.001445	-0.004310	5.99	-103.59	-2.34
444	-57.14	80.00	-0.000856	-0.005520	6.47	-102.33	-3.86
445	-49.52	80.00	-0.000287	-0.006931	6.49	-100.24	-5.40
446	-41.90	80.00	0.000196	-0.008494	5.91	-97.27	-6.64
447	-34.29	80.00	0.000525	-0.010103	4.76	-93.63	-7.16
448	-26.67	80.00	0.000649	-0.011597	3.28	-89.84	-6.60
449	-19.05	80.00	0.000566	-0.012790	1.92	-86.66	-4.89
450	-11.43	80.00	0.000326	-0.013522	1.09	-84.82	-2.33
451	-3.81	80.00	0.000025	-0.013707	1.02	-84.65	0.47
452	3.81	80.00	-0.000239	-0.013353	1.61	-86.00	2.92
453	11.43	80.00	-0.000388	-0.012550	2.58	-88.42	4.63
454	19.05	80.00	-0.000386	-0.011428	3.62	-91.32	5.50
455	26.67	80.00	-0.000231	-0.010126	4.48	-94.26	5.62
456	34.29	80.00	0.000054	-0.008763	5.06	-96.91	5.18
457	41.90	80.00	0.000432	-0.007430	5.31	-99.09	4.39
458	49.52	80.00	0.000867	-0.006189	5.27	-100.78	3.43
459	57.14	80.00	0.001324	-0.005075	4.98	-101.98	2.43
460	64.76	80.00	0.001775	-0.004104	4.53	-102.75	1.50
461	72.38	80.00	0.002199	-0.003276	3.98	-103.19	0.68
462	80.00	80.00	0.002583	-0.002583	3.38	-103.38	-0.00

(Zero_Test/logfile)

```
Mon Mar  6 17:47:05 MST 1995
-45 degree fault with strength 00 under 100 MPa N-S compression
      Boundary Element input file is ZeroStrength.
      Other Element input file is AUTOGRID.
                Origin: -80.000000 -80.000000
                Length: 160.000000
                Spacing: 20
                Number of elements: 21
      The target directory is Zero_Test.
      The number of boundary line segments = 1.
      The number of other line segments = 21.
      P-wave Velocity = 6100.000000 m/s
      S-wave Velocity = 3600.000000 m/s
      Density = 2750.000000 kg/m^3
      Poisson's Ratio = 0.232784
      Young's Modulus = 87872.804688 MPa
      XX Component of Field Stress = 0.000000 MPa
      XY Component of Field Stress = 0.000000 MPa
      YY Component of Field Stress = -100.000000 MPa
      Ignored 0 points!
Mon Mar  6 17:47:17 MST 1995
```

The actual file contents have been omitted as they are used strictly as a plotxy script. To use this file, type:

plotxy.post < plotxy.dir1

This will produce a cluttered plot of the most compressive stress orientations.

The actual file contents have been omitted as they are used strictly as a plotxy script. To use this file, type:

plotxy.post < plotxy.dir3

This will produce a cluttered plot of the least compressive stress orientations.

The actual file contents have been omitted as they are used strictly as a plotxy script. To use this file, type:

plotxy.post < plotxy.model

This will produce a plot of the faults defined by the input files.

Appendix C: SMUDGE Manual Pages

- SMUDGE
- ELMTPLT
- FIELD_PLT
- FIELD_PRIMER
- PLOTXY.DYN
- PXY2EPS
- STRESSROT
- UPLOT

```
%!PS-Adobe-1.0
%%Creator: griffy:purbut (Roderick Flores)
%%Title: ManualPages
%%CreationDate: Tue Mar  7 14:56:19 1995
%%DocumentFonts: Courier-Bold Courier
% Start of enscript.pro -- prolog for text file translator
% Copyright (c) 1984,1985,1987 Adobe Systems Incorporated. All Rights Reserved.
% GOVERNMENT END USERS: See Notice file in TranScript library directory
% -- probably /usr/lib/ps/Notice
% RCSID: $Header: enscript.pro,v 2.2 87/11/17 16:39:56 byron Rel $
letter
save/EnscriptJob exch def
/StartEnscriptDoc{$enscript begin}def
/$enscript 50 dict def $enscript begin
/EndEnscriptDoc{end}def
/S/show load def
/X{exch 0 rmoveto S}def
/Y{exch 0 exch rmoveto S}def
/B{3 1 roll moveto S}def
/F{$fd exch get setfont}def
/StartPage{/svpg save def .05 dup scale}def
/EndPage{svpg restore showpage}def
/DoPreFeed{/statusdict where{pop
 statusdict/prefeed known{statusdict exch/prefeed exch put 0}if}if pop}def
/Landscape{90 rotate 0 -15840 translate}def
/SetStTime{statusdict /manualfeedtimeout 120 put} def
/SetStatus{statusdict /manualfeed true put
   statusdict /product get (LaserWriter) eq
   {version (23.0) eq  % Don't redefine EndPage if printer is not "Classic LW"
     {/EndPage {svpg restore
         {statusdict /printerstatus get exec 16#22000000 and 0 eq{exit}if}loop
         showpage}def}if }if}def

/ISO-Latin-1 [
/.notdef /.notdef /.notdef /.notdef /.notdef  /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef  /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef  /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef  /.notdef /.notdef /.notdef
/space /exclam  /quotedbl  /numbersign  /dollar /percent /ampersand
/quoteright /parenleft  /parenright /asterisk  /plus /comma /hyphen
/period /slash /zero /one /two /three /four /five /six /seven /eight
/nine /colon /semicolon /less /equal /greater /question /at /A /B /C /D
/E /F /G /H /I /J /K /L /M /N /O /P /Q /R /S /T /U /V /W /X /Y /Z
/bracketleft /backslash  /bracketright  /asciicircum  /underscore
/quoteleft /a /b /c /d /e /f /g /h /i /j /k /l /m /n /o /p /q /r /s /t
/u /v /w /x /y /z /braceleft /bar /braceright /asciitilde /.notdef
/dotlessi /dieresis/ring  /grave  /acute  /circumflex/tilde /macron
/brev /dotaccent /cedilla /.notdef /.notdef /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef  /.notdef /.notdef /.notdef
/.notdef /.notdef /.notdef /.notdef /.notdef  /.notdef /.notdef /.notdef
/space /exclamdown /cent /sterling /currency /yen /brokenbar /section
/dieresis /copyright /ordfeminine /guillemotleft /logicalnot /endash /registered
/macron /degree /plusminus /twosuperior /threesuperior /acute /mu
/paragraph /bullet /cedilla /onesuperior /ordmasculine /guillemotright
/onequarter /onehalf /threequarters /questiondown /Agrave /Aacute /Acircumflex
/Atilde /Adieresis /Aring /AE /Ccedilla /Egrave /Eacute /Ecircumflex
/Edieresis /Igrave /Iacute /Icircumflex /Idieresis /Eth /Ntilde
/Ograve /Oacute /Ocircumflex /Otilde /Odieresis /multiply /Oslash /Ugrave
/Uacute /Ucircumflex /Udieresis /Yacute /Thorn /germandbls /agrave
/aacute /acircumflex /atilde /adieresis /aring /ae /ccedilla /egrave
/eacute /ecircumflex /edieresis /igrave /iacute /icircumflex /idieresis
/eth /ntilde /ograve /oacute /ocircumflex /otilde /odieresis
/divide /oslash /ugrave /uacute /ucircumflex /udieresis /yacute
/thorn /ydieresis ] def

/SetUpFonts
 {dup/$fd exch array def{
    findfont dup maxlength dict begin
       { 1 index /FID ne { def } { pop pop } ifelse } forall
       /Encoding ISO-Latin-1 def
       /FontName /Null def
       FontName currentdict definefont
    end
    exch scalefont $fd 3 1 roll put}repeat}def

/InitGaudy{/TwoColumn exch def /BarLength exch def
 /ftD /Times-Bold findfont 12 UP scalefont def
 /ftF /Times-Roman findfont 14 UP scalefont def
 /ftP /Helvetica-Bold findfont 30 UP scalefont def}def
/U{1440 mul}def
/UP{U 72 div}def
/LB{/pts exch UP def /charcolor exch def /boxcolor exch def /font exch def
 /label exch def /dy exch def /dx exch def /lly exch def /llx exch def
 gsave boxcolor setgray
 llx lly moveto dx 0 rlineto 0 dy rlineto dx neg 0 rlineto closepath fill
 /lines label length def
 /yp lly dy add dy lines pts mul sub 2 div sub pts .85 mul sub def
 font setfont charcolor setgray
 label {dup stringwidth pop 2 div llx dx 2 div add exch sub yp moveto show
   /yp yp pts sub def}forall grestore}def
/Gaudy{/Page exch def /Date exch def /File exch def /Comment exch def
 .25 U 10.2 U BarLength .1 sub U .25 U [File] ftF .97 0 14 LB
 .25 U 10.45 U BarLength .1 sub U .25 U [Comment] ftF 1 0 14 LB
 .25 U 10.2 U 1 U .5 U Date ftD .7 0 12 LB
 BarLength .75 sub U 10.2 U 1 U .5 U [Page] ftP .7 1 30 LB
 TwoColumn{BarLength 2 div .19 add U 10.2 U moveto 0 -10 U rlineto stroke}if
}def
end
StartEnscriptDoc % end fixed prolog
1 200 /Courier-Bold
0 200 /Courier
2 SetUpFonts
%%EndProlog
%%Page: ? 1
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        1)B
0 F
900 13800(SMUDGE\(N\)        MISC. REFERENCE MANUAL PAGES        SMUDGE\(N\))B
900 12920(NAME)B
1500 12700(SMUDGE - Stress Modeling Utilizing a DBI-algorithm  in  Geo-)B
1500 12480(physical Elastostatics \(2-Dimensional\))B
900 12040(SYNOPSIS)B
1500 11820(smudge -h -in [control filename])B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
-120(G_)X
-120(E Calculates the stresses and displacements for a plate)X
1500 10940(of  infinite extent for a given set of initial boundary con-)B
1500 10720(ditions.  _)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
```

```
-120(G_)X
-120(E uses either the  default  instruction  file)X
1500 10500(_)B
-120(c_)X
-120(o_)X
-120(n_)X
-120(t_)X
-120(r_)X
-120(o_)X
-120(l or a file specified on the command line.)X
900 10060(OPTIONS)B
1500 9840(The default is -in "control" \(use the input file title  _)B
-120(c_)X
-120(o_)X
-120(n_)X
-120(-)X
1500 9620(_)B
-120(t_)X
-120(r_)X
-120(o_)X
-120(l \).)X
1500 9180(-h                Help \(bring up this manual page\).)B
1500 8960(-in _)B
-120(f_)X
-120(i_)X
-120(l_)X
-120(e_)X
-120(n_)X
-120(a_)X
-120(m_)X
-120(e      Instruction file name \(See Below\).)X
900 8520(INSTALLING)B
1500 8300(Extract the archive by typing:)B
2100 8080(tar xvf smudge.final.tar)B
1500 7860(Change directory to smudge.  type:)B
2100 7640(make)B
1500 7420(and the executables will be compiled.  Make  sure  that  the)B
1500 7200(directory  smudge  is  in your directory.  To clean up after)B
1500 6980(smudge, type:)B
2100 6760(make clean)B
900 6100(INSTRUCTION FILE:)B
1500 5880(The instruction file contains a number of lines defining the)B
1500 5660(properties  of  an  the operations that _)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
-120(G_)X
-120(E will perform.)X
1500 5440(An example of this file is:)B
2100 5000(Vertical Slit in Infinite Plate)B
2100 4780(number of boundary segments, number of other segments)B
2100 4560(8 21)B
2220 4340(Vp Vs Density)B
2220 4120(6100 3600 2750)B
2100 3900(Field stress values \(XX, XY, YY\))B
2100 3680(0.0 0.0 -100.0)B
2100 3460(Boundary Element Input File)B
2100 3240(instructions)B
2100 3020(Other Element Input File)B
2100 2800(Studyarea_grid)B
2100 2580(Output Destination Directory)B
2100 2360(Gap)B
1500 1920(The first line of this file contains a descriptive title for)B
1500 1700(the  model  of  up to 80 characters.  This feature should be)B
900 820(Sun Release 4.1    Last change: March 7, 1995                1)B
EndPage
%%Page: ? 2
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        2)B
0 F
900 13800(SMUDGE\(N\)         MISC. REFERENCE MANUAL PAGES          SMUDGE\(N\))B
1500 12920(used in order to maintain an accurate record of exactly what)B
1500 12700(the model is approximating. The second line as well as every)B
1500 12480(other line onward is strictly used to give information  per-)B
1500 12260(taining  to  the  next item that is to be input.  This makes)B
1500 12040(making changes to the file quick because no field  memoriza-)B
1500 11820(tion is required.  Remember, these lines are ignored!!!  The)B
1500 11600(third line contains two values: the number of boundary  seg-)B
1500 11380(ments  and  the  number  of  non-boundary segments for which)B
1500 11160(stress and displacement values are desired.   Boundary  seg-)B
1500 10940(ments  are  straight lines made up of any number of boundary)B
1500 10720(elements that describe the initial conditions of the  infin-)B
1500 10500(ite  plate  \(See the _)B
-120(B_)X
-120(o_)X
-120(u_)X
-120(n_)X
-120(d_)X
-120(a_)X
-120(r_)X
-120(y _)X
-120(E_)X
-120(l_)X
-120(e_)X
-120(m_)X
-120(e_)X
-120(n_)X
-120(t _)X
-120(I_)X
-120(n_)X
-120(p_)X
-120(u_)X
-120(t _)X
-120(F_)X
-120(i_)X
-120(l_)X
-120(e section for)X
1500 10280(more information\).  The non-boundary  segments  are  usually)B
1500 10060(grids that are used for visualization of the state of stress)B
1500 9840(\(See the _)B
-120(S_)X
-120(p_)X
-120(e_)X
-120(c_)X
-120(i_)X
-120(f_)X
-120(i_)X
-120(e_)X
-120(d  _)X
-120(E_)X
-120(l_)X
-120(e_)X
-120(m_)X
-120(e_)X
-120(n_)X
-120(t  _)X
```

```
-120(I_)X
-120(n_)X
-120(p_)X
-120(u_)X
-120(t  _)X
-120(F_)X
-120(i_)X
-120(l_)X
-120(e  section  for  more)X
1500 9620(information\).  The  fifth line specifies the constants of P-)B
1500 9400(wave velocity, S-wave velocity, and density.  These  values)B
1500 9180(characterize  the  elastic  properties of the medium and are)B
1500 8960(typically more available than the constants which  they  are)B
1500 8740(converted  into  \(Poisson's ratio and Young's modulus\).  The)B
1500 8520(seventh line is the symmetric tensor describing the  initial)B
1500 8300(state  of  stress  of the body, where the first value is the)B
1500 8080(XX, the second is the XY, and the third is the YY  component)B
1500 7860(of the regional stress tensor:)B
2100 7640(Sigma =  [ XX XY ])B
3180 7420([ YX XX ].)B
1500 7200(It is important to note that the values  specified  on  this)B
1500 6980(and  the  fifth  line  are the only scaled dependent factors)B
1500 6760(given to the program.  It is wise to use consistent units of)B
1500 6540(measure  for  these values because all results will have the)B
1500 6320(same units.  For example a  Young's  modulus  given  in  MPa)B
1500 6100(\(Newtons  per  square  meter  or  kilogram  per meter-square)B
1500 5880(second\) implies that the initial stress values are  also  in)B
1500 5660(MPa  and  that coordinate system used to specify fault loca-)B
1500 5440(tion is based upon meters.  The  ninth  and  eleventh  lines)B
1500 5220(contain  the  names  of  the input files used to specify the)B
1500 5000(locations and initial conditions for the  boundary  elements)B
1500 4780(and  specified elements respectively.  See the _)B
-120(B_)X
-120(o_)X
-120(u_)X
-120(n_)X
-120(d_)X
-120(a_)X
-120(r_)X
-120(y _)X
-120(E_)X
-120(l_)X
-120(e_)X
-120(-)X
1500 4560(_)B
-120(m_)X
-120(e_)X
-120(n_)X
-120(t _)X
-120(I_)X
-120(n_)X
-120(p_)X
-120(u_)X
-120(t _)X
-120(F_)X
-120(i_)X
-120(l_)X
-120(e and the _)X
-120(S_)X
-120(p_)X
-120(e_)X
-120(c_)X
-120(i_)X
-120(f_)X
-120(i_)X
-120(e_)X
-120(d _)X
-120(E_)X
-120(l_)X
-120(e_)X
-120(m_)X
-120(e_)X
-120(n_)X
-120(t  _)X
-120(I_)X
-120(n_)X
-120(p_)X
-120(u_)X
-120(t  _)X
-120(F_)X
-120(i_)X
-120(l_)X
-120(e  sec-)X
1500 4340(tions  for  details on these files.  The thirteenth line, or)B
1500 4120(should I say the fourteenth in order to avoid any bad omens,)B
1500 3900(is  the name of the directory in which the output files will)B
1500 3680(be placed.)B
900 3020(BOUNDARY ELEMENT INPUT FILE)B
1500 2800(The boundary element file can consist of any number of lines)B
1500 2580(which  define  the  locations and initial conditions for all)B
1500 2360(the straight boundary segments in the  body.   Each  segment)B
1500 2140(can  in  turn be broken into any number of even spaced boun-)B
1500 1920(dary elements having  the  same  initial  conditions.   When)B
1500 1700(choosing the locations of these segments, it is necessary to)B
900 820(Sun Release 4.1    Last change: March 7, 1995              2)B
EndPage
%%Page: ? 3
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        3)B
0 F
900 13800(SMUDGE\(N\)          MISC. REFERENCE MANUAL PAGES          SMUDGE\(N\))B
1500 12920(consider the best  approximation  for  the  actual  boundary)B
1500 12700(curve.  It is also necessary to choose an adequate number of)B
1500 12480(elements to accurately solve the linear system of  equations)B
1500 12260(without  overwhelming  the capabilities of the platform \(SUN)B
1500 12040(Sparcs are spooky, but that is what we've got\).  Each  boun-)B
1500 11820(dary  _)B
-120(e_)X
-120(l_)X
-120(e_)X
-120(m_)X
-120(e_)X
-120(n_)X
-120(t  requires  around  .1  Kilobytes of memory, so)X
1500 11600(choose carefully.)B
1500 11160(The format of this file is: number of boundary elements,  X-)B
1500 10940(start, Y-start, X-end, Y-end, Strength on positive side, and)B
1500 10720(Strength on negative side.  For example:)B
2100 10280(40 20 -20 -20 20  00. 00.)B
1500 9840(defines a boundary  segment  consisting  of  forty  elements)B
1500 9620(which extends from the X-Y coordinates \(20,-20\) to \(-20,20\).)B
1500 9400(The strength on both sides of the fault is zero.  The  pro-)B
1500 9180(gram  expects  to  read the number of lines specified in the)B
1500 8960(first field on the the instruction file \(See the _)B
-120(I_)X
-120(n_)X
-120(s_)X
```

```
-120(t_)X
-120(r_)X
-120(u_)X
-120(c_)X
-120(t_)X
-120(i_)X
-120(o_)X
-120(n)X
1500 8740(_)B
-120(F_)X
-120(i_)X
-120(l_)X
-120(e  section  for  details on this file\).  Any lines beyond)X
1500 8520(the number specified will be ignored.  This  can  be  useful)B
1500 8300(for annotating the specifics about the model.)B
900 7640(SPECIFIED ELEMENT INPUT FILE)B
1500 7420(This specified element input file can consist of any  number)B
1500 7200(of  lines  which define the locations of points where stress)B
1500 6980(and displacement information are desired within the infinite)B
1500 6760(plate.   Each  segment can in turn be broken into any number)B
1500 6540(of even spaced elements.  When choosing these  points,  keep)B
1500 6320(in mind that each point requires the as many calculations as)B
1500 6100(there are boundary elements.  The format of this file is:)B
2100 5660(xbeg ybeg xend yend num)B
1500 5220(where xbeg and ybeg are the beginning X  and  Y  coordinates)B
1500 5000(respectively,  and xend and yend are the terminating X and Y)B
1500 4780(coordinates respectively.  This line segment will be  broken)B
1500 4560(into num equal length segments.  This file must also contain)B
1500 4340(four informational lines at the its end.   These  must  take)B
1500 4120(the form:)B
2100 3680(Origin: X-origin Y-origin)B
2100 3460(Length: L)B
2100 3240(Spacing: S)B
2100 3020(Number of elements: S+1)B
1500 2580(where X-origin and Y-origin are the X-Y coordinates  of  the)B
1500 2360(lowermost left side of the points; L is the length on a side)B
1500 2140(of a square encompassing all specified points; and S is  the)B
1500 1920(number  of  steps  taken  from  the Origin along the side of)B
1500 1700(length, L.  These values are mostly for the  edification  of)B
900 820(Sun Release 4.1    Last change: March 7, 1995              3)B
EndPage
%%Page: ? 4
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        4)B
0 F
900 13800(SMUDGE\(N\)          MISC. REFERENCE MANUAL PAGES          SMUDGE\(N\))B
1500 13140(Statistics   kept   by  _)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
-120(G_)X
-120(E  and  their  values  are  not)X
1500 12920(paramount.)B
1500 12480(Rather than creating a specific grid file, it is also possi-)B
1500 12260(ble  to have the program create its own grid \(In my opinion,)B
1500 12040(this is the only option\). This is accomplished by  replacing)B
1500 11820(the original grid file on the eleventh line with the keyword)B
1500 11600(_)B
-120(A_)X
-120(U_)X
-120(T_)X
-120(O_)X
-120(G_)X
-120(R_)X
-120(I_)X
-120(D followed by the  origin  of  the  grid  \(lower-left)X
1500 11380(corner\)  in  \(x,y\)  coordinates,  the length of the grid box)B
1500 11160(\(x,y\), and the number of grid points that are  to  occur  in)B
1500 10940(the area.  For example,)B
1500 10500(Other Element Input File \(line 10\))B
1500 10280(AUTOGRID -40 -40 80 160)B
1500 9840(creates a grid beginning from  \(-40,-40\)  which  extends  to)B
1500 9620(\(40,40\) with a spacing of 1/2 a coordinate.)B
900 8960(OUTPUT FILE: Bdry_input)B
1500 8740(This file contains all the initial values for  the  boundary)B
1500 8520(elements.  It exists solely for confirmation purposes.  The)B
1500 8300(format of this file is five informational lines followed  by)B
1500 8080(two  lines  for  each element \(one for the positive side and)B
1500 7860(one for the negative side\) as such:)B
2100 7420(Element X-cen Y-cen L Theta Sigma_S)B
1500 6980(where Element is the element number, X-cen and Y-cen  center)B
1500 6760(are the coordinates of the Center of the boundary element, L)B
1500 6540(is the length of the boundary element, Theta  is  the  angle)B
1500 6320(the  boundary  element makes with respect to the X-axis, and)B
1500 6100(Sigma_S is the given strength of that element.)B
900 5440(OUTPUT FILE: Bdry_Elements)B
1500 5220(This file contains all the values for the specified boundary)B
1500 5000(elements:  calculated and given.  Specifically, the file has)B
1500 4780(the format of four information lines followed by  two  lines)B
1500 4560(for  each  element containing \(one for the positive side and)B
1500 4340(one for the negative side\) as such:)B
2100 3900(Us Un Ux Uy Sigma_s Sigma_N)B
1500 3460(where Us and Un  are  the  shear  and  normal  displacements)B
1500 3240(respectively,  Ux  and Uy are the displacements rotated into)B
1500 3020(absolute X-Y coordinates, and Sigma_s and  Sigma_n  are  the)B
1500 2800(shear  and  normal stresses respectively.  This file is used)B
1500 2580(by the program ELMTPLT.)B
900 820(Sun Release 4.1    Last change: March 7, 1995              4)B
EndPage
%%Page: ? 5
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        5)B
0 F
900 13800(SMUDGE\(N\)          MISC. REFERENCE MANUAL PAGES          SMUDGE\(N\))B
900 12700(OUTPUT FILE: Disp.gph)B
1500 12480(This file contains all of the x and  y  displacement  values)B
1500 12260(for  the  boundary and specified elements.  This file format)B
1500 12040(has the format of two informational lines  followed  by  two)B
1500 11820(lines  for  each boundary element \(one for the positive side)B
1500 11600(and one for the negative side\) and one line for each  speci-)B
1500 11380(fied element as such:)B
2100 10940(X-pos Y-pos Ux Uy,)B
1500 10500(where X-pos and Y-pos are the X and  Y  coordinates  of  the)B
1500 10280(center  of  the  boundary element or specified point, and Ux)B
1500 10060(and Uy are the displacements at these points.  This file  is)B
1500 9840(used by the program UPLOT and the CSH script FIELD_PRIMER.)B
900 9180(OUTPUT FILE: Sigma_Spy)B
1500 8960(This file contains all of the shear and normal stress values)B
1500 8740(for  the  boundary and specified elements.  This file format)B
1500 8520(has the format of two informational lines  followed  by  one)B
1500 8300(line for each specified element and two lines for each boun-)B
1500 8080(dary element \(one for the positive  side  and  one  for  the)B
1500 7860(negative side\) as such:)B
2100 7420(X-pos Y-pos Sigma_1 Sigma_3 Sigma_1dir Sigma_3dir)B
1500 6980(where X-pos and Y-pos are the X and  Y  coordinates  of  the)B
```

```
1500 6760(center  of  the boundary element or specified point, Sigma_1)B
1500 6540(and Sigma_3 are the  most  and  least  compressive  stresses)B
1500 6320(respectively, and Sigma_1dir and Sigma_3dir are the orienta-)B
1500 6100(tions of these stresses.  This  file  is  used  by  Spyglass)B
1500 5880(Transform to plot the principle stress.)B
900 5220(OUTPUT FILE: Spec_Elements)B
1500 5000(This file contains all of the displacements and stress  ten-)B
1500 4780(sor  for all all specified points.  This file has the format)B
1500 4560(of five informational lines followed by one  line  for  each)B
1500 4340(point assembled as:)B
2100 3900(Point X Y Ux Uy Sigma_xx Sigma_yy Sigma_xy)B
1500 3460(where Point is the sequential numbering of specified points,)B
1500 3240(X  and Y are the coordinates of the point, Ux and Uy are the)B
1500 3020(displacements, and Sigma_xx, Sigma_yy, and SIgma_xy are  the)B
1500 2800(stress  tenors  in the absolute X-Y coordinate system.  This)B
1500 2580(file is used by STRESSROT  to  plot  the  shear  and  normal)B
1500 2360(stresses throughout the infinite plate.)B
900 1700(OUTPUT FILE: logfile)B
1500 1480(A file containing all pertinent  information  pertaining  to)B
900 820(Sun Release 4.1    Last change: March 7, 1995                5)B
EndPage
%%Page: ? 6
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        6)B
0 F
900 13800(SMUDGE\(N\)        MISC. REFERENCE MANUAL PAGES          SMUDGE\(N\))B
1500 12700(the execution of _)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
-120(G_)X
-120(E. It has the format of:)X
2100 12260(time of the inception of the _)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
-120(G_)X
-120(E execution)X
2100 12040(title of model)B
2100 11820(boundary element input file name)B
2100 11600(other input element file name)B
2700 11380(Origin \(See SPECIFIED FILE INPUT\))B
2700 11160(Length \(See SPECIFIED FILE INPUT\))B
2700 10940(Spacing \(\(See SPECIFIED FILE INPUT\))B
2700 10720(Number of elements \(See SPECIFIED FILE INPUT\))B
2100 10500(output directory)B
2100 10280(number of boundary line elements)B
2100 10060(number of specified line elements)B
2100 9840(Poisson's Ratio)B
2100 9620(Young Modulus)B
2100 9400(XX Component of Field Stress)B
2100 9180(XY Component of Field Stress)B
2100 8960(YY Component of Field Stress)B
2100 8740(number of points ignored \(NOIGNORE compilation option\))B
2100 8520(time of the inception of the _)B
-120(S_)X
-120(M_)X
-120(U_)X
-120(D_)X
-120(G_)X
-120(E execution)X
900 7860(OUTPUT FILE: plotxy.dir1)B
1500 7640(A plotxy script that plots  the  orientations  of  the  most)B
1500 7420(compressive  principle stress.  The output of this script is)B
1500 7200(very rough, and should only be used as a "quick  look."   To)B
1500 6980(use this script, type:)B
1500 6760(plotxy.post < plotxy.dir1)B
1500 6540(This  creates  the  file  Sigma1dir.ps.   Use  the  standard)B
1500 6320(postscript viewer to see this file.)B
900 5660(OUTPUT FILE: plotxy.dir3)B
1500 5440(A plotxy script that plots the  orientations  of  the  least)B
1500 5220(compressive  principle stress.  The output of this script is)B
1500 5000(very rough, and should only be used as a "quick  look."   To)B
1500 4780(use this script, type:)B
2100 4340(plotxy.post < plotxy.dir3)B
1500 3900(This  creates  the  file  Sigma3dir.ps.   Use  the  standard)B
1500 3680(postscript viewer to see this file.)B
900 3020(OUTPUT FILE: plotxy.model)B
1500 2800(A plotxy script that plots the boundary segments.   This  is)B
1500 2580(useful  for  visually  checking  your  models.   To use this)B
1500 2360(script, type:)B
2100 1920(plotxy.post < plotxy.model.)B
1500 1480(This creates the file model.ps.  Use the standard postscript)B
900 820(Sun Release 4.1    Last change: March 7, 1995                6)B
EndPage
%%Page: ? 7
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        7)B
0 F
900 13800(SMUDGE\(N\)        MISC. REFERENCE MANUAL PAGES          SMUDGE\(N\))B
1500 12920(viewer to see this file.)B
900 12260(SEE ALSO)B
1500 12040(field_plt\(N\),field_primer,      uplot\(N\),      ghostview\(1\),)B
1500 11820(pageview\(1\), plotxy.dyn\(N\))B
900 11160(REFERENCES)B
1500 10940(Crouch, S.L. and A.M. Starfield)B
1500 10720(Boundary Element Methods is Solid Mechanics)B
1500 10500(Cambridge University Press, 1990.)B
1500 10060(Press, William H., Saul A. Teukolsky, William T. Vetterling,)B
1500 9840(and Brian P. Flannery)B
1500 9620(Numberical Recipes in C: The Art of Scientific Computing)B
1500 9400(Cambridge University Press, 1992.)B
1500 8960(Spyglass Transform Quick Tour and Reference)B
1500 8740(Spyglass Inc., 1993)B
900 8300(AUTHOR)B
1500 8080(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995                7)B
EndPage
%%Page: ? 8
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        8)B
0 F
900 13800(ELMTPLT\(N\)        MISC. REFERENCE MANUAL PAGES          ELMTPLT\(N\))B
900 12920(NAME)B
1500 12700(elmtplt - plot one of the six values output by SMUDGE versus)B
1500 12480(boundary element number.)B
900 12040(SYNOPSIS)B
1500 11820(elmtplt)B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(E_)X
-120(L_)X
-120(M_)X
```

```
-120(T_)X
-120(P_)X
-120(L_)X
-120(T uses the file Bdry_Elements in order to create plots)X
1500 10940(the  boundary  elements \(x-axis\) versus one of the following)B
1500 10720(\(y-axis\):)B
2100 10500(shear displacement              \(Us\))B
2100 10280(normal displacement             \(Un\))B
2100 10060(x-axis parallel displacement    \(Ux\))B
2100 9840(y-axis parallel displacement    \(Uy\))B
2100 9620(shear stress                   \(Sigma_S\))B
2100 9400(normal stress                  \(Sigma_N\).)B
1500 9180(_)B
-120(E_)X
-120(L_)X
-120(M_)X
-120(T_)X
-120(P_)X
-120(L_)X
-120(T begins by asking for the output file name.  It  then)X
1500 8960(asks  for  the bounding box, in X and Y inches, in which the)B
1500 8740(field will be plotted.  _)B
-120(E_)X
-120(L_)X
-120(M_)X
-120(T_)X
-120(P_)X
-120(L_)X
-120(T then queries for  x  and  y-)X
1500 8520(axis  labels.   If  labels  are  chosen, they are input as a)B
1500 8300(string followed by a space and >< as a terminator \(Don't ask)B
1500 8080(why!\).   _)B
-120(E_)X
-120(L_)X
-120(M_)X
-120(T_)X
-120(P_)X
-120(L_)X
-120(T  then  asks  if  numbering  of  the axes is)X
1500 7860(desired.  These are simple yes/no questions.  Next,  _)B
-120(E_)X
-120(L_)X
-120(M_)X
-120(T_)X
-120(P_)X
-120(L_)X
-120(T)X
1500 7640(asks  which  type  of plot to make.  Choose 1-6 as described)B
1500 7420(above.  Next _)B
-120(E_)X
-120(L_)X
-120(M_)X
-120(T_)X
-120(P_)X
-120(L_)X
-120(T outputs automatically chosen bounds  of)X
1500 7200(the  y-axis.   If these are not acceptable, then the user is)B
1500 6980(requested  to  give  a  minimum  and  maximum  respectively.)B
1500 6760(_)B
-120(E_)X
-120(L_)X
-120(M_)X
-120(T_)X
-120(P_)X
-120(L_)X
-120(T  outputs  a  postscript  file   that can be examined)X
1500 6540(using a standard PS-viewer.)B
900 6100(SEE ALSO)B
1500 5880(field_plt\(N\) uplot\(N\), ghostview\(1\), pageview\(1\), smudge\(N\),)B
1500 5660(plotxy.dyn\(N\), pxy2eps\(N\))B
900 5220(AUTHOR)B
1500 5000(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995                1)B
EndPage
%%Page: ? 9
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        9)B
0 F
900 13800(FIELD_PLT\(N\)       MISC. REFERENCE MANUAL PAGES       FIELD_PLT\(N\))B
900 12920(NAME)B
1500 12700(field_plt - plot the displacement  field  output  by  SMUDGE)B
1500 12480(into a postscript document containing a deformation mesh)B
900 12040(SYNOPSIS)B
1500 11820(field_plt)B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
-120(L_)X
-120(T uses the file primer.dat  in  order  to  create  a)X
1500 10940(deformed  mesh representing the displacement field output by)B
1500 10720(smudge.  The CSH script FIELD_PRIMER must be  used  to  pre-)B
1500 10500(process  the  data  from  SMUDGE  into  the file primer.dat.)B
1500 10280(_)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
-120(L_)X
-120(T begins by asking for the  output  file  name.   It)X
1500 10060(then asks for the bounding box, in X-Y coordinates, in which)B
1500 9840(the field will  be  plotted.   These  values  are  input  as)B
1500 9620(lower-left  X-coordinate,  upper-right  X-coordinate, lower-)B
1500 9400(left Y-coordinate, and upper-right Y-coordinate as such:)B
2100 9180(X0 X1 Y0 Y1.)B
1500 8960(_)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
-120(L_)X
-120(T then queries for magnification factors for  the  x)X
1500 8740(and  y  values.   The final input that _)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
```

```
-120(L_)X
-120(T requires is)X
1500 8520(the maximum allowable displacement.  This is useful for)B
1500 8300(plots with high magnification because displacements greater)B
1500 8080(than this value are not plotted.  _)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
-120(L_)X
-120(T outputs a)X
1500 7860(postscript file that can be examined using a standard PS-)B
1500 7640(viewer.)B
900 7200(SEE ALSO)B
1500 6980(field_primer\(N\), elmtplt\(N\), uplot\(N\), ghostview\(1\),)B
1500 6760(pageview\(1\), smudge\(N\), plotxy.dyn\(N\))B
900 6320(AUTHOR)B
1500 6100(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995           1)B
EndPage
%%Page: ? 10
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        10)B
0 F
900 13800(FIELD_PRIMER\(N\)   MISC. REFERENCE MANUAL PAGES   FIELD_PRIMER\(N\))B
900 12920(NAME)B
1500 12700(field_primer - arrange the displacements output from SMUDGE)B
1500 12480(so that field_plt can use them)B
900 12040(SYNOPSIS)B
1500 11820(field_primer)B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
-120(R_)X
-120(I_)X
-120(M_)X
-120(E_)X
-120(R arranges the displacement field output by)X
1500 10940(SMUDGE into a format that _)B
-120(F_)X
-120(I_)X
-120(E_)X
-120(L_)X
-120(D__)X
-120(P_)X
-120(L_)X
-120(T can read.  The file)X
1500 10720(Disp.gph must be present in order to use this utility.)B
900 10280(SEE ALSO)B
1500 10060(field_plt\(N\), smudge\(N\), uplot\(N\), plotxy.dyn\(N\), ghost-)B
1500 9840(view\(1\), pageview\(1\))B
900 9400(AUTHOR)B
1500 9180(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995           1)B
EndPage
%%Page: ? 11
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        11)B
0 F
900 13800(PLOTXY.DYN\(N\)     MISC. REFERENCE MANUAL PAGES     PLOTXY.DYN\(N\))B
900 12920(NAME)B
1500 12700(plotxy.dyn - a version of plotxy.post which dynamically)B
1500 12480(allocates memory.)B
900 12040(SYNOPSIS)B
1500 11820(plotxy.dyn -Kb [memory])B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(P_)X
-120(L_)X
-120(O_)X
-120(T_)X
-120(X_)X
-120(Y._)X
-120(D_)X
-120(Y_)X
-120(N is a version of plotxy.post with dynamic memory)X
1500 10940(allocation.)B
900 10500(OPTIONS)B
1500 10280(The default is -Kb 32.76.)B
1500 9840(-Kb _)B
-120(m_)X
-120(e_)X
-120(m_)X
-120(o_)X
-120(r_)X
-120(y - allocate kilowords of floating point memory)X
900 9400(SEE ALSO)B
1500 9180(plotxy\(1L\))B
900 8740(AUTHOR)B
1500 8520(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995           1)B
EndPage
%%Page: ? 12
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        12)B
0 F
900 13800(PXY2EPS\(N\)        MISC. REFERENCE MANUAL PAGES        PXY2EPS\(N\))B
900 12920(NAME)B
1500 12700(pxy2eps - convert plotxy postscript files into encapsulated)B
1500 12480(postscript.)B
900 12040(SYNOPSIS)B
1500 11820(pxy2eps)B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(P_)X
-120(X_)X
-120(Y_)X
-120(2_)X
-120(E_)X
-120(P_)X
-120(S asks for an input file \(the PLOTXY postscript file\))X
1500 10940(and attempts to match a bounding box to the graph in the)B
1500 10720(document.  _)B
-120(P_)X
-120(X_)X
-120(Y_)X
-120(2_)X
-120(E_)X
```

```
-120(P_)X
-120(S then asks for you to enter the values  of)X
1500 10500(the  bounding box \(in the same order as it estimated\).  Gen-)B
1500 10280(erously overestimate the first two and slightly  the  other.)B
1500 10060(For example if the program estimates:)B
2460 9840(Lower left X: 1)B
2460 9620(Lower left Y: 1)B
2460 9400(Upper right X: 432)B
2460 9180(Upper right Y: 432,)B
1500 8960(give it something like -30 30 440 440.)B
900 8520(SEE ALSO)B
1500 8300(elmtplt\(N\), plotxy.dyn\(N\),)B
900 7860(AUTHOR)B
1500 7640(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995                    1)B
EndPage
%%Page: ? 13
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        13)B
0 F
900 13800(STRESSROT\(N\)       MISC. REFERENCE MANUAL PAGES       STRESSROT\(N\))B
900 12920(NAME)B
1500 12700(stressrot - rotate the stress tensors output by SMUDGE  from)B
1500 12480(absolute  X-Y  coordinates  into  an  X'-Y'  system which is)B
1500 12260(oriented at some specified angle with respect to the  origi-)B
1500 12040(nal system)B
900 11600(SYNOPSIS)B
1500 11380(stressrot)B
900 10940(DESCRIPTION)B
1500 10720(_)B
-120(S_)X
-120(T_)X
-120(R_)X
-120(E_)X
-120(S_)X
-120(S_)X
-120(R_)X
-120(O_)X
-120(T rotates the specified point stress tensors  output)X
1500 10500(in  the  file Spec_Elements by SMUDGE into a coordinate sys-)B
1500 10280(tem, X'-Y', oriented at some angle to the absolute X-Y  sys-)B
1500 10060(tem.   The program prompts for an output file name, the com-)B
1500 9840(ponents of a stress tensor describing  the  regional  trend,)B
1500 9620(and  for  the  angle at between the X'-axis and the original)B
1500 9400(X-axis.  The program first removes  the  regional  state  of)B
1500 9180(stress, if it is specified, and then it rotates each indivi-)B
1500 8960(dual tensor. _)B
-120(S_)X
-120(T_)X
-120(R_)X
-120(E_)X
-120(S_)X
-120(S_)X
-120(R_)X
-120(O_)X
-120(T then outputs the  original  X  and  Y)X
1500 8740(coordinates   and  the  rotated  stress  tensor:  Sigma_xx',)B
1500 8520(Sigma_xy', Sigma_yx', and Sigma_yy'.  These values are  use-)B
1500 8300(ful  for  examining  shear stress or stress drop parallel to)B
1500 8080(the X' axis as well as the normal stress  in  X'-Y'  coordi-)B
1500 7860(nates  [Note: the shear stress is defined to be Sigma_xy' or)B
1500 7640(Sigma_yx'  and  the  normal  stress  to  be   \(Sigma_xx'   +)B
1500 7420(Sigma_yy'\)].  Typically  these  values  are  displayed using)B
1500 7200(Transform.)B
900 6760(SEE ALSO)B
1500 6540(field_plt\(N\), smudge\(N\), uplot\(N\),elmtplt\(N\), plotxy.dyn\(N\),)B
1500 6320(ghostview\(1\), pageview\(1\))B
900 5880(REFERENCES)B
1500 5660(Spyglass Transform Quick Tour and Reference)B
1500 5440(Spyglass Inc., 1993)B
900 5000(AUTHOR)B
1500 4780(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995                    1)B
EndPage
%%Page: ? 14
StartPage
1 F
900 14900 (ManualPages        Tue Mar  7 14:56:19 1995        14)B
0 F
900 13800(UPLOT\(N\)          MISC. REFERENCE MANUAL PAGES          UPLOT\(N\))B
900 12920(NAME)B
1500 12700(uplot - plot the displacement field output by SMUDGE into  a)B
1500 12480(postscript document using vectors)B
900 12040(SYNOPSIS)B
1500 11820(uplot)B
900 11380(DESCRIPTION)B
1500 11160(_)B
-120(U_)X
-120(P_)X
-120(L_)X
-120(O_)X
-120(T uses the file Disp.gph in order to create displacement)X
1500 10940(vectors  from  the  output of smudge.  The vectors are lines)B
1500 10720(emanating from the particle's starting position \(a  dot\)  to)B
1500 10500(its  final  position.  _)B
-120(U_)X
-120(P_)X
-120(L_)X
-120(O_)X
-120(T begins by asking for the output)X
1500 10280(file name.  It then asks for the bounding box, in X-Y  coor-)B
1500 10060(dinates,  in  which the field will be plotted.  These values)B
1500 9840(are  input  as  lower-left  X-coordinate,   upper-right   X-)B
1500 9620(coordinate,  lower-left  Y-coordinate,  and  upper-right  Y-)B
1500 9400(coordinate as such:)B
2100 9180(X0 X1 Y0 Y1.)B
1500 8960(_)B
-120(U_)X
-120(P_)X
-120(L_)X
-120(O_)X
-120(T then queries for magnification factors for the x and y)X
1500 8740(values.   _)B
-120(U_)X
-120(P_)X
-120(L_)X
-120(O_)X
-120(T outputs a postscript file  that can be exam-)X
1500 8520(ined using a standard PS-viewer.)B
900 8080(SEE ALSO)B
1500 7860(field_plt\(N\)    elmtplt\(N\),    ghostview\(1\),    pageview\(1\),)B
1500 7640(smudge\(N\), plotxy.dyn\(N\),)B
900 7200(BUGS)B
1500 6980(The output can be quite troublesome to read for dense grids.)B
1500 6760(Field_plt   gives   a  much  more  useful  output  in  these)B
1500 6540(instances.)B
900 6100(AUTHOR)B
```

```
1500 5880(Roderick Flores)B
900 820(Sun Release 4.1    Last change: March 7, 1995                1)B
EndPage
%%Trailer
EndEnscriptDoc
EnscriptJob restore
```