



2ej
7
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

“SIMULACION NUMERICA DE SECUENCIAS
DE DEFORMACION GEOLOGICA”

T E S I S

QUE PARA OBTENER EL TITULO DE
INGENIERO GEOLOGO

P R E S E N T A :

Juan Contreras Pérez

**TESIS CON
FALLA DE ORDEN**

MEXICO, D. F.

1989



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

RESUMEN

Se desarrollaron analíticamente funciones de desplazamiento para el modelado cinemático de la deformación bidimensional de una secuencia estratigráfica al deslizarse sobre una configuración especificada de fallas, cuya geometría es construida a base de segmentos de rectas. Las fallas del modelo subdividen al medio en dominios de echados y campos de vectores de desplazamiento constantes que están delimitados por los planos axiales de las inflexiones en las fallas. La magnitud de desplazamiento de cuerpo rígido es constante para todo el medio y su dirección es paralela al echado del segmento de falla subyacente. A esta deformación se agrega una componente de cizallamiento simple que está distribuida de manera homogénea dentro del medio desplazado. Las ecuaciones generales que controlan el desplazamiento de cada nodo de la malla a lo largo de la configuración de la falla, de acuerdo a un sistema de referencia cartesiano externo que es paralelo a un segmento de la falla, son:

$$\begin{aligned}x' &= x + s + h \cdot \tan\psi \\y' &= y\end{aligned}$$

para partículas desplazadas dentro del mismo dominio de echados, y

$$\begin{aligned}x' &= (x+s+h \cdot \tan\psi)\cos\alpha - (y+m \cdot a-b) \cdot (\cos\alpha+1)/m \\y' &= (x+s+h \cdot \tan\psi)\sin\alpha - (y+m \cdot a-b)\sin\alpha/m + y\end{aligned}$$

para partículas desplazadas a través de un plano axial al dominio de echados vecino, donde x', y' es la posición desplazada de x, y ; s es la magnitud del vector de desplazamiento; ψ y h son un ángulo de cizalla simple y la altura sobre el punto de aplicación de cizalla, respectivamente; α es la diferencia de los echados de dos segmentos vecinos de la superficie de la falla y m es la tangente de la bisectriz formada por los segmentos de falla.

El medio litoestratigráfico no deformado se representó por una malla de elementos finitos cuadrilaterales y las funciones de desplazamiento definidas arriba analíticamente se evaluaron después para los nodos de la malla.

El transporte de la malla a través de planos axiales produce una distorsión longitudinal y por cizallamiento angular de la

celula inicial, mientras que su área permanece constante. Estos cambios están en función de α según las ecuaciones:

$$E_y = \sqrt{4 \cdot \tan^2 \alpha / 2 + 1} - 1$$

$$\theta = \pi/2 - \arccos \left(\frac{\tan \alpha / 2 \cos \alpha (\cos \alpha + \sin \alpha - 1) + 1 / 2 \sin \alpha}{E_y + 1} \right)$$

donde E_y es la distorsión longitudinal y θ la distorsión por cizallamiento angular.

El algoritmo se programó en lenguaje Turbo-Pascal con el uso de bibliotecas de análisis numérico y de graficación y se implementaron en una computadora IBM-PC.

El programa permite modelar la geometría del subsuelo de configuraciones litológicas deformadas por movimientos a lo largo de fallas normales o inversas y apreciar la distorsión provocada en la malla por los desplazamientos.

INDICE

pag

RESUMEN	1
1.- INTRODUCCION	5
2.- LA CONSTRUCCION DE SECCIONES ESTRUCTURALES.	
2.1.- Datos de entrada en la construcción de secciones estructurales.	6
2.2.- Métodos de interpolación y extrapolación en la construcción de secciones.	7
2.3.- Técnicas geométricas usadas en la construcción de secciones estructurales.	10
2.4.- Tratamiento analítico de superficies alabeadas.	14
3.- SECCIONES ESTRUCTURALES BALANCEADAS.	
3.1.- Principios del método.	22
3.2.- Aplicación del método para el cálculo de la profundidad al despegue de una serie litoestratigráfica.	24
3.3.- Aplicación del método para el cálculo del echado de segmentos no expuestos de una cabalgadura.	25
3.4.- Aplicación del método para anticlinales formados sobre una raspa tectónica.	28
3.5.- Aplicación del método para fallas normales.	34
4.- CINEMATICA DE LA DEFORMACION.	
4.1.- La deformación como una serie en el tiempo.	36
4.2.- Relaciones entre el estado inicial y el estado final deformado de un medio litoestratigráfico.	38

5.- MODELADO DIRECTO DE SECUENCIAS DE DEFORMACION	
5.1.- Presentación del algoritmo.	43
5.2.- Programación del algoritmo en Turbo-Pascal.	58
5.3.- Implementación del programa en un sistema IBM-PC	71
5.4.- Comparación de los resultados sintéticos obtenidos con el algoritmo con datos reales de cinturones de pliegues y cabalgaduras.	71
CONCLUSIONES	84
AGRADECIMIENTOS	85
LISTA DE REFERENCIAS	86
APENDICE: LISTADO EN TURBO-PASCAL DEL PROGRAMA FUENTE ...	89

1. INTRODUCCION

Esta tesis usa de manera interdisciplinaria métodos de análisis numérico y graficación por computadora para resolver un problema común en la geología estructural. El problema es el siguiente: Dado un juego de datos reales obtenidos de una configuración estructural a escala regional, es decir, datos de subsuelo, de pozos y de cartografía geológica de superficie, como se puede definir (i) la geometría inicial de la configuración, (ii) las transformaciones que lleven de la geometría inicial hasta la geometría final deformada, y (iii) la geometría completa de la estructura deformada final, comprendida entre el basamento tectónico y la superficie, o hasta con la inclusión de partes ya erosionadas.

En el segundo y tercer capítulo se presenta una recopilación de los conceptos y técnicas necesarios en el análisis geométrico de estructuras geológicas y en la construcción de secciones, poniendo énfasis en el "balanceo de secciones". Esta es una prueba para verificar la consistencia geométrica de un modelo propuesto y está basada en el principio de conservación de masa de la mecánica del medio continuo. El capítulo cuarto es una introducción a los conceptos manejados en el análisis de la deformación. En el capítulo quinto, parte principal de la tesis, se desarrolla, parcialmente basado en los conceptos y técnicas de los capítulos precedentes, un algoritmo para la simulación bidimensional de secuencias de deformación. El algoritmo se implementó en una computadora IBM-PC y se comprobó al aplicarlo a juegos de datos reales de cinturones de pliegues y cabalgaduras.

El programa es una herramienta útil en la automatización de la construcción de secciones estructurales.

2. CONSTRUCCION DE SECCIONES ESTRUCTURALES

2.1 DATOS DE ENTRADA EN LA CONSTRUCCION DE SECCIONES.

La información litoestratigráfica y estructural necesaria en la construcción de secciones se puede obtener de muchas fuentes de información. Una de las principales es la recolección de datos de campo. Esta incluye la medición de espesores de unidades litoestratigráficas, descripciones de las mismas, obtención de la orientación de vectores de significado estructural, como son echado, foliación, lineación, planos de falla, polaridad de las capas, etc. Los datos anteriores también se pueden estimar a partir de imágenes de satélite, fotografías aéreas y cartas geológicas aplicando técnicas geométricas. Del análisis de los patrones de afloramiento se obtiene información acerca del estilo de deformación.

La aplicación de métodos geofísicos de exploración proporciona parámetros y una cuantificación de la geometría del subsuelo. De estos métodos, la sísmología es la que proporciona mayor información y capacidad de resolución. Ofrece los mejores resultados en áreas sin complicaciones estructurales mayores y con buenos horizontes reflectores o difractores. El procesamiento de datos es mejor en aquellas secuencias donde los echados son menores de 15 a 20 grados, que son los valores máximos que admite la migración sin introducir distorsión ni ruido de dispersión. Los métodos potenciales, especialmente la magnetometría, son utilizados para configurar la geometría del basamento.

Otra fuente de datos de entrada en la construcción de secciones la constituyen los pozos. En pozos petroleros de exploración es práctica usual obtener núcleos de roca, construir el perfil litológico que atraviesa la barrena y correr una serie de registros geofísicos para determinar y correlacionar las unidades litológicas a partir de sus propiedades electromagnéticas, nucleares y acústicas. El registro de echados proporciona la dirección y magnitud con que buzan las capas de las rocas sedimentarias. A estos se suma el perfil sísmico vertical o PVS. Los datos obtenidos del PVS permite mejorar la interpretación estructural, estratigráfica y litológica, así como determinar impedancias acústicas por debajo de la barrena.

2.2 METODOS DE INTERPOLACION Y EXTRAPOLACION EN

LA CONSTRUCCION DE SECCIONES.

Por naturaleza, los datos de entrada en la construcción de secciones son discretos y su distribución es heterogénea. Estas características implican que la cuantificación geométrica de cada punto de una sección tiene asociado un grado de confiabilidad, prestándose a usar modelos y criterios probabilísticos. Esto se ha hecho en el campo de la inteligencia artificial, donde se usan bancos de información y reglas de razonamiento, a los cuales se les asigna un grado de calidad o certidumbre, respectivamente. También se aplica en los modelos estadísticos, como son los modelos Bayesianos, donde un experto asigna a cada elemento de la información un grado de certeza razonable o estimador bayesiano. La distribución de valores de probabilidad en dos o tres dimensiones a partir de datos discretos de entrada se pueden visualizar por medio de gráficas de contorno.

Las características de una distribución espacial irregular y discreta también invitan al uso de métodos numéricos de interpolación y extrapolación para complementar la información disponible (Davis, 1973).

La interpolación es un método que conduce a soluciones aproximadas para determinar el comportamiento de una serie de puntos conocidos en los intervalos que quedan entre ellos. Al igual que la interpolación, la extrapolación sólo ofrece soluciones aproximadas para determinar el comportamiento en las vicinidades externas de una serie de puntos conocidos.

Existe una gran cantidad de interpoladores como son el interpolador de Newton, Lagrange, "Spline" cúbico, Spline cúbico paramétrico, aproximaciones por polinomios, Serie de Fourier, etc. (Pres et al., 1986).

Estos interpoladores se aplican todavía muy poco en la geología debido a la complejidad matemática de las técnicas. Otra causa es que generalmente se requiere que las variables independientes tengan que incrementarse continuamente, lo que con frecuencia no ocurre con las curvas cerradas y convolutas comunes en las estructuras geológicas. Este problema se ha resuelto mediante el uso del "spline" cúbico paramétrico (Evans et al., 1986).

El interpolador en su forma básica, es un polinomio de tercer grado de la forma:

$$z = a \cdot x^3 + b \cdot x^2 + c \cdot x + d \dots (2.1)$$

Según McCoss (1987) hay tres razones para seleccionar una ecuación de tercer grado: (i) es el polinomio de menor grado con primera y segunda derivada continua; (ii) las ecuaciones de grado superior necesitan de mayor tiempo de cómputo y pueden originar curvas más convolutas; y (iii) una posible razón física es que la ecuación adopta la forma para la mínima energía de deformación de una varilla elástica.

El método del spline paramétrico consiste en lo siguiente: si de cada pareja de puntos sucesivos se conoce su buzamiento real o aparente, se puede trazar un arco de curva de tercer grado rotada un ángulo θ tal que: a) θ es el ángulo bisector de las dos pendientes (Figura 2.1a); b) θ corresponde a una dirección conocida sin punto fijo de localización (eje del pliegue) (Figura 2.1b); c) θ es un ángulo, tal que la curva tiene que pasar por un punto conocido de la superficie, donde se desconoce su buzamiento (contacto en un pozo) (Figura 2.1c); y d) θ es un ángulo, tal que la longitud de arco sea mínima (esto sucede cuando el coeficiente "a" tiende a cero (Figura 2.1c)).

El primer caso corresponde a dos puntos sin control auxiliar; el ángulo bisector no es una restricción formal, solamente se ha observado que conduce a configuraciones más realistas. Los incisos b y c corresponden a puntos con control auxiliar; y el último caso se aplica cuando se tiene una densidad grande de información.

Los coeficientes a, b, c y d se calculan de la siguiente manera (McCoss, 1987):

$$a = \frac{(\tan(\alpha_1 + 90^\circ - \theta) + \tan(\alpha_2 + 90^\circ - \theta))((x_2 - x_1)\text{sen}\theta - (z_2 - z_1)\text{cos}\theta)}{((x_2 - x_1)\text{sen}\theta - (z_2 - z_1)\text{cos}\theta)^3} - \frac{2((x_2 - x_1)\text{cos}\theta - (z_2 - z_1)\text{sen}\theta)}{((x_2 - x_1)\text{sen}\theta - (z_2 - z_1)\text{cos}\theta)^3} \dots (2.2)$$

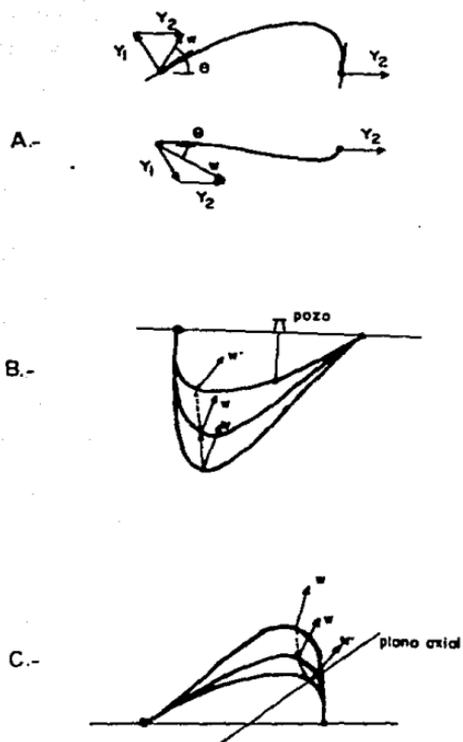


Figura 2.1 Interpolador "Spline" cúbico paramétrico, (a) construcción a partir del ángulo bisector de los ejes conocidos por medio de sus direcciones perpendiculares (y_1 , y_2); (b) construcción a partir de un dato del subsuelo y (c) de acuerdo a una dirección conocida. (Según Mc Coss, 1987, Fig. 3, 5 y 4).

$$b = \frac{3((x_2-x_1)\cos\theta - (z_2-z_1)\sin\theta)}{((x_2-x_1)\cos\theta - (z_2-z_1)\sin\theta)^2} - \frac{(2\tan(\alpha_1+90^\circ-\theta) + \tan(\alpha_2+90^\circ-\theta))((x_2-x_1)\sin\theta - (z_2-z_1)\cos\theta)}{((x_2-x_1)\cos\theta - (z_2-z_1)\sin\theta)^2} \dots (2.3)$$

$$c = \tan(\alpha + 90^\circ - \theta) \dots (2.4)$$

$$d = 0 \dots (2.5)$$

donde α_1 y α_2 son los echados de los puntos conocidos.

Las ecuaciones anteriores se encuentran desplazadas con el origen en (x_1, z_1) por lo que los valores x, z obtenidos en la ecuación 2.1 deben trasladarse nuevamente a $(0, 0)$.

2.3 TECNICAS GEOMETRICAS USADAS EN LA CONSTRUCCION DE SECCIONES ESTRUCTURALES.

Las características estructurales construidas en una sección tienen que corresponder a las observaciones en el terreno, pozos, o secciones sísmicas. Si existen cambios estructurales asociados a los cambios litológicos, estos deben de tomarse en cuenta al construir una sección. Estos cambios en el estilo de la deformación están gobernados por la presión, la temperatura y la velocidad de deformación a la que estuvo y está sujeta la roca y por sus propiedades físicas (Patterson, 1978). Cualitativamente, las rocas incompetentes como la lutita y la argilita se deforman de manera dúctil, mientras que las rocas competentes como los carbonatos y areniscas se deforman rígidamente (Ramsay y Huber, 1987).

La ocurrencia reiterada de los mismos ambientes de deformación, con características estructurales específicas, ha llevado a formular técnicas de construcción de secciones para áreas sujetas a estos ambientes. Uno de los más recurrentes es el plegamiento paralelo. El plegamiento paralelo requiere que cada horizonte tome la forma de las capas adyacentes y que los espesores de las capas, medidos perpendicularmente al echado, sean constantes. Mecánicamente, esto implica que no existe deformación

interna de las capas. Para que esto ocurra se tiene que generar un deslizamiento relativo entre las capas o sea cisallamiento paralelo a la estratificación.

El método de Busk (Suppe, 1985) es una de la técnicas geométricas más conocidas de construcción de secciones con plegamiento paralelo. En esta los pliegues se dibujan por medio de series de arcos de círculos concéntricos para las capas adyacentes y tangentes para las regiones colindantes con radio de curvatura diferente (Figura 2.2a). Esto significa que la evoluta de los pliegues es una función discreta, ya que los horizontes de un mismo intervalo angular tienen el mismo centro de curvatura.

Sin embargo, este tipo de plegamiento acepta modelados geométricos alternos como los pliegues angulares o los pliegues de curvatura no circular (curvatura circular diferencial). Los pliegues angulares ocurren donde la asociación estratigráfica consiste de estratos de rocas competentes (carbonatos y areniscas) que alternan con estratos dúctiles de menor espesor (Ramsay y Huber, 1987). Estos pliegues se modelan mediante series de segmentos de líneas rectas, formando regiones con echados constantes, llamados "dominios de echados" (Figura 2.2b). Estos son limitados por los planos axiales que se generan al flexionarse las capas al forzar los dominios vecinos. Para que los espesores de las capas permanezcan constantes entre regiones adyacentes, es necesario que la orientación de los planos axiales correspondan a la bisectriz del ángulo formado por la inflexión. Este tipo de deformación se observa en varios materiales de la naturaleza, variando su escala desde milímetros hasta un kilómetro. La Figura 2.3a establece las relaciones trigonométricas de los pliegues angulares en sección, donde $\alpha_1, \dots, \alpha_n$ son los echados de las kink bands y β es la pendiente de la bisectriz.

Mecánicamente, los pliegues angulares se forman por una rotación, donde los planos axiales constituyen el lugar geométrico de todos los centros de rotación (Ramsay, 1967; Laubscher, 1976), como lo muestra la Figura 2.3b. A la pérdida de longitud o acortamiento que sufren las rocas por este mecanismo se le llama "acortamiento por rotación de cuerpo rígido", denotado Δx_{rot} en la Figura 2.3a.

El método de pliegues angulares también se aplica al caso de los pliegues sinusoidales, ya que cualquier curva se puede

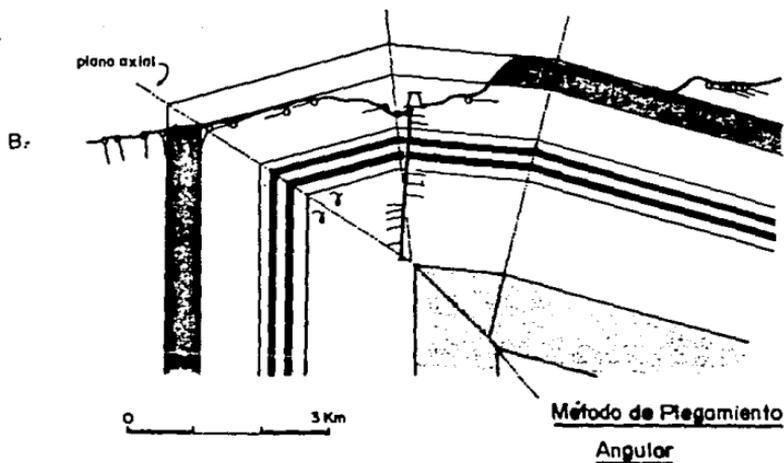
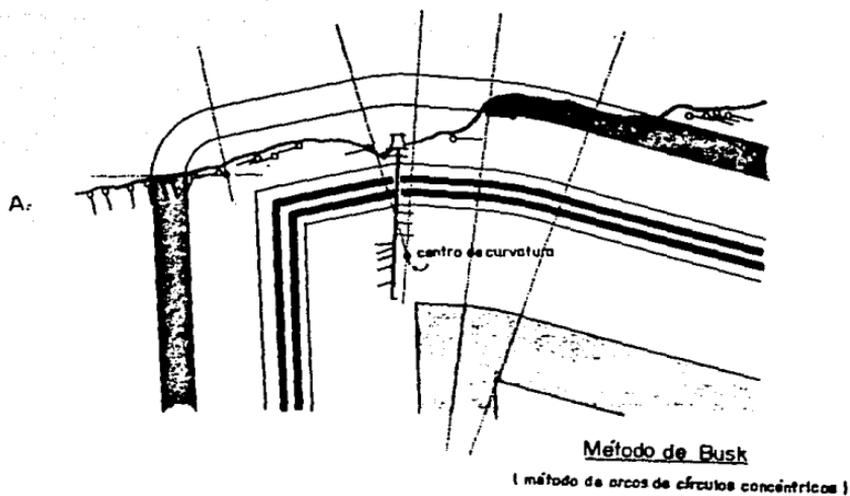


Figura 2.2 Comparación entre los métodos de (a) Busk y (b) plegamiento angular. (Según Suppe, 1985, Fig. 2-24).

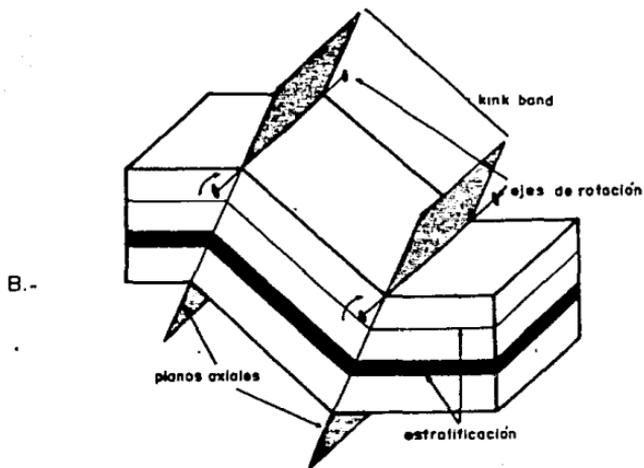
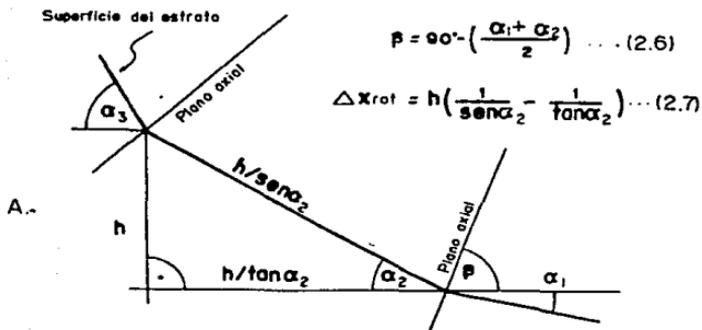


Figura 2.3a Relaciones trigonométricas de los pliegues angulares. (Según Suter, 1981, Fig. 2).

Figura 2.3b Mecánica de la deformación de los pliegues angulares (Según Faill, 1973).

aproximar mediante un número finito de líneas rectas (si se cuenta con una cantidad adecuada de datos). Este método también es aplicable a las secuencias que tienen una razón fija en el cambio de espesor. Esto se calcula mediante la siguiente relación (Suppe, 1985) :

$$\frac{\text{sen} \gamma_1}{\text{sen} \gamma_2} = \frac{T_1}{T_2} \dots (2.8)$$

Donde γ_1 y γ_2 son los ángulos que rigen la posición del eje del pliegue y T_1, T_2 son los espesores reales adyacentes al eje del pliegue (Figura 2.4).

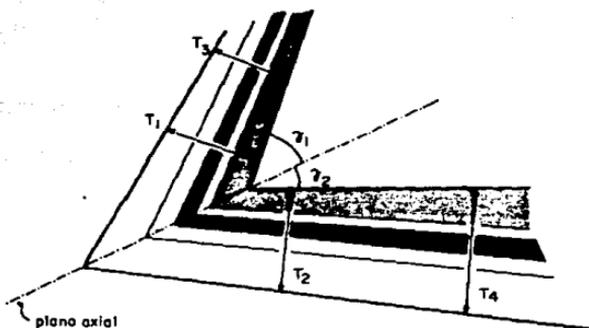
Ramsay y Huber (1987) propusieron un método de construcción más general que los anteriores, que puede ser usado tanto en ambientes paralelos como no paralelos. El método está basado en el análisis de los diagramas de variación de espesores o patrones de isógonas que son líneas que unen puntos con igual echado. Si se conocen los patrones característicos de las isógonas de las rocas en estudio, es posible aplicar las restricciones geométricas impuestas por las líneas de isógonas para construir la geometría de las superficies de las capas adyacentes.

DePaor (1988) describe la técnica de la siguiente manera: "El procedimiento requiere control de superficie y/o control de pozos y algún estimador de la reología de las capas. Primero se establece un factor de convergencia de las isógonas de las rocas estratificadas que afloran en la superficie o algún punto conocido atravesado por un pozo. Las capas del subsuelo son interpoladas manteniendo constantes los echados a lo largo de las isógonas (Figura 2.5a). Después de que se pasa un primer contacto, se busca un nuevo factor de convergencia, tomando en consideración la naturaleza reológica de la capa subyacente" (Figuras 2.5b,c,d).

El método de las isógonas, al igual que el método de los pliegues anulares, también permite aproximar tanto plegamientos curvos como angulares.

2.4 TRATAMIENTO ANALITICO DE SUPERFICIES ALABEADAS

Una superficie alabeada se puede analizar desde dos puntos de vista : como una membrana bidimensional en el espacio, o como la



$$\frac{T_1}{T_2} = \frac{T_3}{T_4} = C$$

Figura 2.4 Pliegues angulares en secuencias con variación constante en el espesor (Según Suppe, 1985, Fig. 2-3).

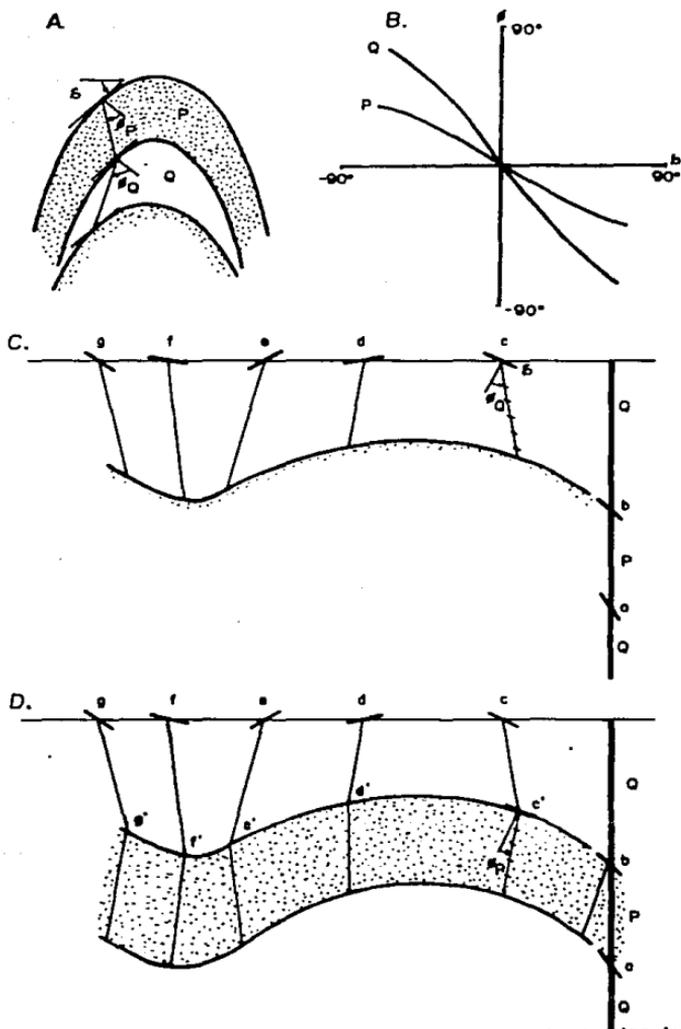


Figura 2.5 Método de construcción de secciones por medio de las isógonas.
 (Según Ramsay y Huber, 1987, Fig. 18.9).

superficie que limita a un sólido del espacio circundante. A partir de ambos puntos de vista se originan propiedades diferentes. Desde el primero se desprenden las propiedades relativas o extrínsecas (aquellas relacionadas con el espacio circundante) y desde la segunda perspectiva, se desprenden las propiedades absolutas o intrínsecas (aquellas relacionadas con su forma). Una de las propiedades absolutas más importantes es la "curvatura".

La curvatura en un punto de una superficie o línea se define como la razón de la variación de la tangente con respecto a la longitud de arco (Ramsay, 1976; Suppe, 1985). Si $f(x)$ es la función que define una curva en el plano y S es la longitud de arco, entonces la curvatura C está dada por:

$$C = \frac{d f'(x)}{ds} \dots (2.9)$$

donde $f'(x)$ es la derivada con respecto a x .

El recíproco de la curvatura es llamado "radio de curvatura" (Ramsay, 1976; Suppe, 1985) y corresponde al radio del círculo tangente cuya curvatura es igual a la del punto en cuestión. En el caso especial de las rectas y de los puntos de inflexión (puntos que separan regiones con diferente concavidad), la curvatura es igual a cero y por tanto, el radio de curvatura es infinito.

Las curvas naturales formadas por las rocas estratificadas plegadas contienen puntos especiales definidos por sus propiedades geométricas (Figura 2.6a). La cresta está definida por tener pendiente nula y además cambia de signo, de (+) a (-). El valle también tiene pendiente nula y cambia de signo de (-) a (+). El punto de charnela está caracterizado por tener curvatura máxima relativa. El punto de inflexión es aquel donde la curva cambia de concavidad. Estas propiedades están matemáticamente definidas en la Figura 2.6a.

Al considerar un punto de una superficie alabeada se tiene que por este pasan un número infinito de curvas (Figura 2.6b). Pero sólo existen dos de curvatura extrema (máxima y mínima), denominadas "curvaturas principales". Estas líneas son ortogonales. Si tomamos todos los puntos de la superficie, estas dos líneas definen un sistema coordenado curvilíneo al que se

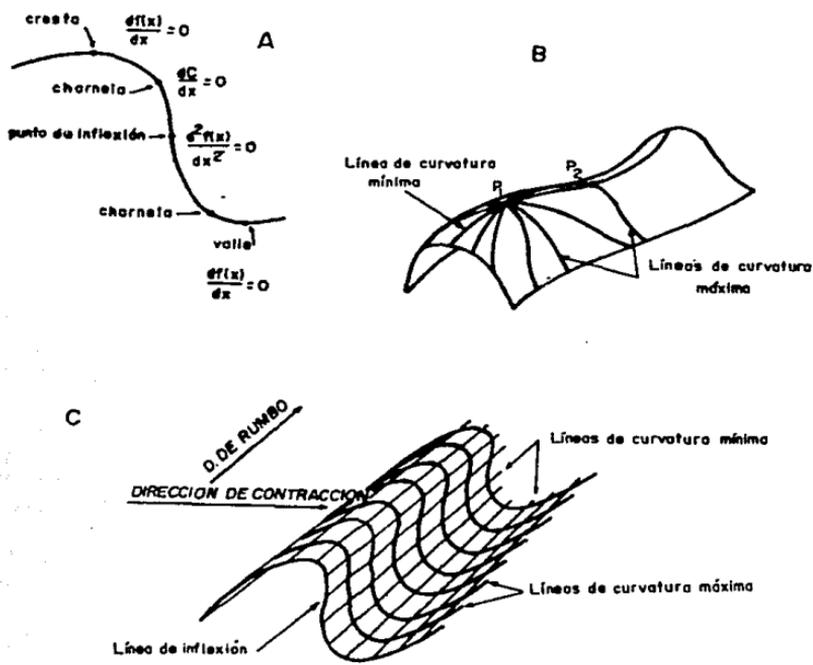


Figura 2.6 (a) Puntos geométricos especiales de un pliegue visto en sección, (b) Curvaturas extremas que pasan por un punto de una superficie no esférica y (c) Sistema coordenado curvilíneo que define las curvaturas extremas. (Según Suppe, 1985, Fig. 9-2 a, b y c).

puede asociar un significado geológico: la dirección promedio de curvatura máxima corresponde a la dirección de la contracción tectónica y la dirección promedio de curvatura mínima al rumbo de los planos axiales de los pliegues (Figura 2.6c).

Una propiedad derivada de la curvatura es la "Curvatura Gaussiana". Esta se define como el producto de las curvaturas principales

$$C_g = (C_{max}) \times (C_{min}) \dots (2.10)$$

y tiene la característica que permanece constante durante cualquier deformación (transformación estrictamente hablando), si esta no introduce un cambio de área. De esta manera, teóricamente los pliegues cilíndricos y cónicos no generan cambios de área, ya que antes y después del plegamiento la curvatura gaussiana es cero.

Si consideramos que las rocas estratificadas generan una familia de superficies (o curvas en sección) que separan cada capa de la adyacente, los ejes de los pliegues corresponden al lugar geométrico de las zonas comunes de la familia de superficies con máximos relativos en la curvatura (Cobbold et al., 1984).

Los planos axiales dependiendo del estilo de la deformación, pueden formar puntos singulares a la primera derivada respecto a la superficie o curva del estrato a que pertenecen (Cobbold et al., 1984). Para pliegues suaves modelables con líneas curvas los planos axiales son superficies continuas y no caen en puntos singulares. Para pliegues angulares, los planos axiales son superficies continuas y caen sobre puntos singulares. Este segundo caso se resuelve introduciendo al análisis matemático bidimensional funciones impulso, salvando el obstáculo de la discontinuidad. Es decir, los límites por la izquierda y derecha para la primera derivada, se ajustan, introduciendo una función impulso (Cobbold et al., 1984).

La manera de expresar analíticamente una superficie, es a través de una función escalar $f(x,y)$. Sin embargo, es mucho más sencillo tratar una superficie a través de los vectores tangentes, normales y de posición, debido a que con estos se caracteriza la posición y orientación de la superficie en un punto (Barr, 1984).

Otra razón es, que cualquier deformación se puede expresar mediante la multiplicación de estos vectores por la matriz jacobiana de la deformación (Kass y Witkin, 1985). Matemáticamente, la transformación se expresa:

$$\begin{array}{l} X = f(x, y, z) \\ Y = g(x, y, z) \\ Z = h(x, y, z) \end{array} \quad \left| \dots \right. \quad (2.11)$$

donde X, Y y Z son las coordenadas transformadas de x, y, z. Esto también se puede expresar de la siguiente manera:

$$\begin{array}{l} \bar{F} = x \cdot i + y \cdot j + z \cdot k \\ \bar{E} = X \cdot i + Y \cdot j + Z \cdot k \quad \dots \quad (2.12) \end{array}$$

donde \bar{F} y \bar{E} son los vectores de posición en el estado inicial y deformado, respectivamente.

La matriz jacobiana esta definida como:

$$J_i = \frac{\partial E(x, y, z)}{\partial x_i} \quad \dots \quad (2.13)$$

$$\begin{array}{l} \text{donde } \frac{\partial x_1}{\partial x} = \frac{\partial x}{\partial x}; \\ \frac{\partial x_2}{\partial y} = \frac{\partial y}{\partial y}; \\ \frac{\partial x_3}{\partial z} = \frac{\partial z}{\partial z}. \end{array}$$

Pero si expresamos la superficie en función de coordenadas curvilíneas u, v

$$F = r(u, v) \quad \dots \quad (2.14)$$

entonces, el vector tangente de la superficie deformada se puede expresar en función de las superficies coordenadas

$$\frac{\partial E}{\partial u} = J \cdot \frac{\partial F}{\partial u} \quad \dots \quad (2.15)$$

en otras palabras el vector tangente deformado $\frac{\partial E}{\partial u}$ es igual a "J" veces el vector tangente inicial.

La transformación para el vector normal \bar{n} (Barr, 1984) está dada por:

$$\bar{N}(X, Y, Z) = \det(J \cdot (J^{-1})^T) \bar{n}(x, y, z) \dots (2.16)$$

donde \bar{N} es el vector transformado.

3 BALANCEO DE SECCIONES ESTRUCTURALES

3.1 PRINCIPIOS DEL METODO

El término "balanceo de secciones" se aplica de dos formas. Geométricamente se refiere a que el área de las rocas que corta una sección ha permanecido constante durante la deformación. Esto se basa en el principio general de la conservación de la masa, establecido en la mecánica del medio continuo, e implica que la deformación a que estuvieron sujetas las rocas no originó cambios en su densidad. Esto significa que el volumen permanezca constante y que el medio sea incompresible. Además, para que el área sea constante, no debe existir un transporte de masa hacia fuera o hacia adentro del sistema, por ejemplo debido a fallas de cizallamiento con componente transversal, a excepción de transporte a lo largo del despegue basal; es decir, se asume un sistema cerrado que únicamente sufre distorsión en el plano ("plane strain") (Suter, 1984a). Para que se cumpla esto, la orientación de la sección deberá coincidir con la dirección del acortamiento tectónico y que no haya distorsión en la dirección axial de la estructura.

En un sentido cinemático, el término se utiliza para indicar que cualquier sección propuesta para un juego de datos, tiene que ser retrodeformable a cualquier estado deformado anterior o completamente retrodeformable. En este caso, la construcción de secciones se basa en el análisis cinemático de la deformación bajo condiciones de área constante.

La construcción manual de este tipo de secciones consume tiempo de trabajo mecánico y por su naturaleza puede contener errores significativos, tornándose necesario el uso de aproximaciones por computadora, basados en principios geométricos de la deformación, donde la rapidez y exactitud de la construcción permite el análisis y la evaluación de soluciones alternas.

En terrenos que no han sufrido una distorsión interna de los estratos, como en el caso del plegamiento paralelo, se puede efectuar una simplificación del método. Debido a que el área de una capa es función de su espesor y longitud y a que el espesor permanece constante, se puede eliminar una dimensión y establecer la conservación del área sólo en función de la longitud. En este caso, la verificación del balance de una sección se reduciría a medir las longitudes de las capas en la

sección palinspástica y en la sección deformada.

Sin embargo, las deformaciones reales introducen cambios netos en el volumen, por lo que se necesitan considerar factores de corrección para la construcción de secciones. La compactación diagenética, por expulsión de agua de los poros y disolución por presión, puede provocar pérdidas en el volumen hasta del 50% (Wood, 1974). En la compactación tectónica, debido a la expulsión de agua de las lutitas y por disolución por presión en las calizas, la pérdida de volumen puede alcanzar 20 a 30%; mientras que el volumen perdido por elongación oblicua al plano de la sección puede ser del orden de 5 al 10% (Hossack, 1979). Para propósitos prácticos se comparan las rocas litificadas no afectadas por la deformación (por ejemplo, en el flanco bajo de una cabalgadura, en el núcleo de un sinclinal acofrado, etc.) con la secuencia deformada actual, y de esta manera la compactación diagenética no entra en los cálculos.

La construcción de secciones balanceadas no se puede efectuar en áreas afalladas con arreglo "en echelón", porque este estilo estructural introduce una distorsión tridimensional de geometría más compleja.

Existe un parámetro geométrico muy utilizado en el balanceo de secciones: el acortamiento. El acortamiento entre dos puntos de un estrato es la diferencia que existe entre la longitud, siguiendo la traza del estrato, a lo largo de la sección y la distancia horizontal que ocupa actualmente (Figura 3.1). El acortamiento tiene dos componentes, uno por rotación y otro por traslación de cuerpo rígido (Means, 1976). La componente rotacional se origina por el plegamiento de las capas o por cambios en la inclinación de una cabalgadura (Figura 2.3a y Figura 3.2). La segunda componente la origina una traslación de cuerpo rígido. Lo anterior se expresa matemáticamente :

$$S_a = l_1 - l_0 \dots (3.1)$$

$$S_r = \frac{l_1 - l_0}{l_0} (100) \dots (3.2)$$

donde S_a y S_r son el acortamiento absoluto y relativo respectivamente, y l_1 y l_0 las longitudes horizontales original y la deformada.

Ahora bien, si el área de una sección deformada por pliegues paralelos permanece constante, las longitudes de las capas medidas en una sección, a diferentes niveles tienen que permanecer constantes (Dahlstrom, 1969). Esto implica, que dadas dos líneas de referencia (Figura 3.1), el acortamiento que sufren las capas entre dichas líneas es el mismo en todos los niveles estructurales; de no ser así la sección no está "balanceada". Otro criterio para verificar la consistencia geométrica de una sección son los ángulos de corte formados por las fallas y la estratificación; estos deben de ser iguales en los bloques desplazados (Crane, 1987).

En síntesis, se puede decir que la construcción de secciones balanceadas es una prueba para verificar la consistencia geométrica de un modelo cinemático propuesto, y que está basada en el principio de la conservación de la masa. Sin embargo, que si una sección es geoméricamente probable, no significa que esta sea una solución única. Pueden existir interpretaciones alternas que sean compatibles con los datos disponibles. Además, siempre hay incertidumbre en los datos de entrada, como por ejemplo en espesores y echados, o al proyectarse los datos. No obstante, una sección que pase la prueba geométrica puede ser correcta, mientras que una sección no balanceada, sencillamente, no es geoméricamente posible.

3.2 APLICACION DEL METODO PARA EL CALCULO DE LA PROFUNDIDAD AL DESPEGUE DE UNA SECUENCIA LITOESTRATIGRAFICA

En los anticlinales, el acortamiento se efectúa muchas veces cerca de la superficie por medio de plegamiento; mientras que, en su núcleo el acortamiento está regido por una o varias fallas inversas con una raíz común que es un horizonte desgarrado u horizonte de despegue, también llamado "despegue basal" o "décollement". Este horizonte es una falla con desplazamiento paralelo a estratos mecánicamente dúctiles.

Lo que controla el tamaño y forma de los pliegues en un cinturón plegado es la posición del plano de despegue junto con el acortamiento lineal. Cuanto más grueso es el paquete de estratos situado sobre el despegue, tanto mayor es la amplitud y longitud de onda del pliegue individual (Ragan, 1984).

La deformación compresional forma un paquete de rocas más gruesas y cortas que la secuencia estratigráfica inicial. Dado que una sección conserva el área, la superficie levantada tiene que ser exactamente igual a la longitud horizontal perdida por el acortamiento (Figura 3.1). Puesto que, tanto el acortamiento como el engrosamiento se pueden medir en secciones, con ellos es posible determinar la profundidad a la cual se encuentra la zona de despegue (Chamberlain, 1910).

La Figura 3.1 muestra el concepto geométrico y su aplicación en la construcción de una sección de los Montes Jura. Según la definición del acortamiento:

$$s = l_0 - l_1 \dots (3.3)$$

donde l_0 es la longitud original y l_1 la longitud horizontal después de la deformación de un horizonte de referencia. También se tiene que:

$$A_r = A_c = s \cdot h \dots (3.4)$$

donde A_r es el área delimitada por las configuraciones inicial y deformada del horizonte de referencia, A_c es el área del acortamiento y h es la profundidad al despegue. Despejando h :

$$h = A_r / s \dots (3.5)$$

que indica la relación para estimar la profundidad al despegue si A_r y s se conocen. A_r se puede estimar en sección mediante una cuadrícula y s en la sección propuesta mediante un curvímetro (Figura 3.1).

3.3 APLICACION DEL METODO PARA EL CALCULO DEL ECHADO DE SEGMENTOS NO EXPUESTOS DE UNA CABALGADURA

El método de balanceo de masa se puede usar para calcular el echado de segmentos no expuestos de una cabalgadura (Suter, 1981). Como se aprecia en la Figura 3.2, tanto el acortamiento lineal como el área del relieve estructural están en función del echado y del segmento no expuesto de la falla inversa. Además, el área entre el nivel estratigráfico original y el nivel deformado de un

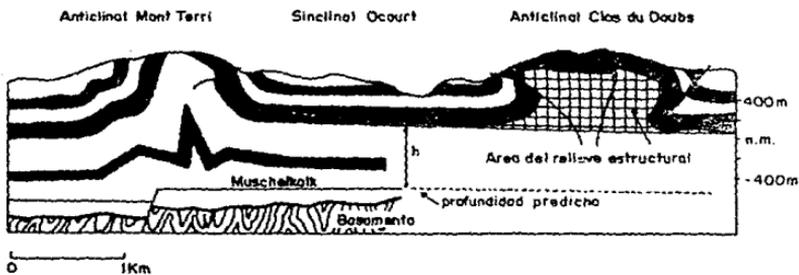
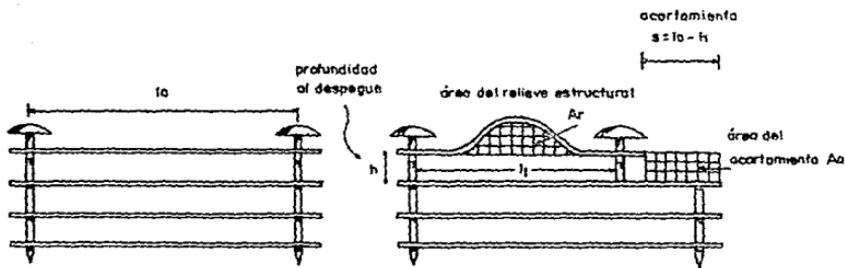


Figura 3.1 Cálculo de la profundidad al despegue (arriba) y un ejemplo de los Montes Jura (abajo). (Según Suppe, 1985, Fig. 2.26 y Laubscher, 1962).

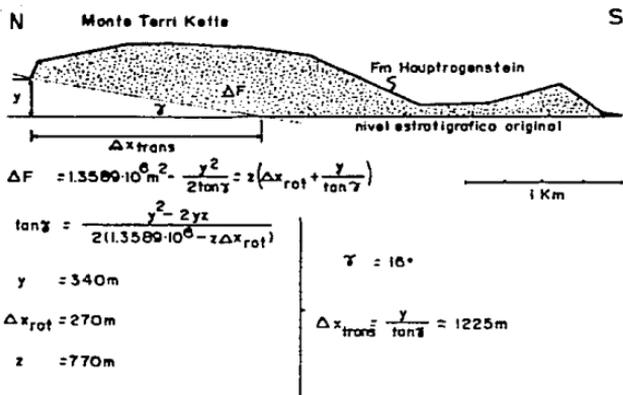


Figura 3.2 Método para calcular el echado (gama) de segmentos no expuestos de una cabalgadura. (Según Suter, 1981, Fig. 3).

horizonte de referencia se puede subdividir en dos elementos de área geoméricamente sencillos en función del echado desconocido:

$$A_r = A_F + y^2/2\tan\gamma \quad \dots (3.6)$$

donde A_r es el área del relieve estructural, A_F el área del bloque imbricado y la segunda expresión del término de la derecha el área entre el nivel estratigráfico original y la traza de la cabalgadura.

$$A_F = A_r - y^2/2\tan\gamma \quad \dots (3.7)$$

Pero A_F también es igual al área levantada por el acortamiento, entonces:

$$A_F = z(\Delta x_{rot} + \Delta x_{trans}) = z(\Delta x_{rot} + y/\tan\gamma) \quad \dots (3.8)$$

donde z es la profundidad al despegue, Δx_{rot} y Δx_{trans} las componentes del acortamiento. Igualando (3.8) con (3.7) y despejando $\tan\gamma$

$$\tan\gamma = \frac{y^2 - 2yz}{2(A_r - z \cdot \Delta x_{rot})} \quad \dots (3.9)$$

que es la ecuación que gobierna el ángulo de la cabalgadura en el subsuelo (Suter, 1981).

Al reconstruir un horizonte de referencia inferior al primero se puede calcular un nuevo buzamiento, repitiendo esta operación hasta el horizonte basal. De los buzamientos calculados se puede obtener, por medio de tratamiento estadístico, una minimización del error por el método de mínimos cuadrados o también se pueden emplear métodos de interpolación.

3.4. APLICACION DEL METODO PARA ANTICLINALES FORMADOS SOBRE UNA RANPA TECTONICA.

Este método, desarrollado por Suppe (1983) es un modelo analítico idealizado en dos dimensiones de pliegues-rampa y sus variantes (Suppe y Marrison, 1979).

Si consideramos que una falla es una superficie alabeada,

entonces debe existir una deformación en uno, o ambos bloques, conforme se deslizan entre sí. Si las rocas están estratificadas, se desarrolla un pliegue en respuesta al cabalgamiento. Este mecanismo de falla-pliegue es conocido como "pliegue-rampa", y se encuentra bien ilustrado en cinturones de pliegues y cabalgaduras, asociado a saltos en el décollement (Suppe, 1983). La cinemática de los pliegues-rampa, causada por salto simple en un décollement, está ilustrada en la Figura 3.3.

Las ilustraciones de la secuencia anterior se construyeron con base en tres suposiciones: 1) conservación de los espesores reales; 2) no exista distorsión dentro de los pliegues angulares; y 3) conservación de la longitud de los estratos. Estas restricciones son las condiciones de partida de su método.

Suppe (op. cit.) establece que la geometría de los pliegues-rampa está gobernada por cuatro ángulos, y determina las relaciones trigonométricas que guardan entre sí. Al manejar ángulos, elimina trabajar con longitudes (espesores, deslizamiento, etc), lo que simplifica el método. Los cuatro ángulos son:

- ϕ : cambio en el ángulo del buzamiento de la falla.
- θ : ángulo de corte formado por la falla y las capas.
- γ : ángulo que gobierna la forma del plegamiento frontal del bloque que cabalga (plano axial).
- β : ángulo formado por las capas del bloque que cabalga y el despegue superior.

La Figura 3.4 muestra el caso de capas del bloque imbricado encontrándose con una inflexión en la rampa. El área de las capas deformadas (triángulo abd) debe ser igual al área de las capas sin deformar (triángulo abc). De manera similar, la longitud de las capas deben permanecer constantes antes y después de la deformación; así las líneas \overline{bd} y \overline{bc} tienen una longitud L . Con estas dos restricciones se puede encontrar la relación que guarda el cambio en el buzamiento de la falla ϕ con respecto a γ . El segmento \overline{bd} de longitud L es dividido por el punto e en dos segmentos: \overline{be} y \overline{ed} ; donde:

$$\overline{be} = L \sin \theta / \sin(2\gamma - \theta) \dots (3.10)$$

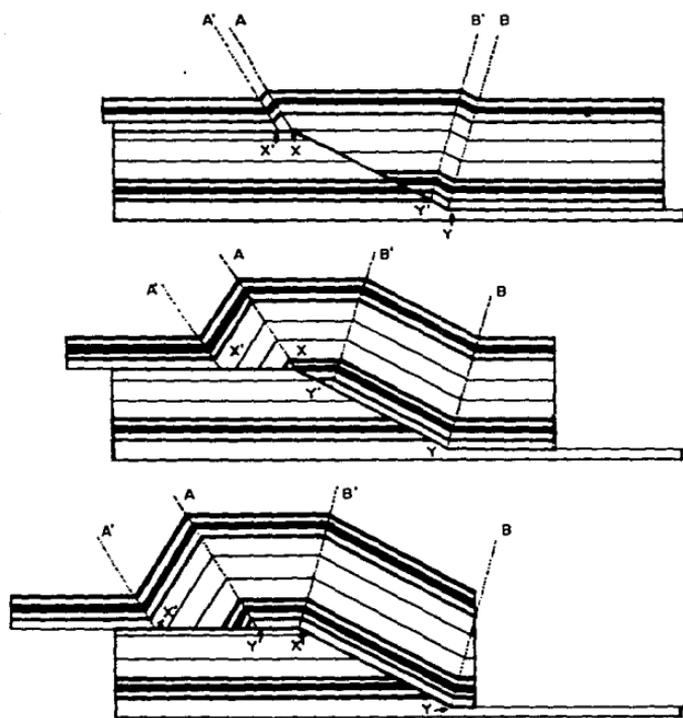


Figura 3.3 Cinemática de la formación de un plegue-rampa. (Según Suppe, 1983, Fig. 3).

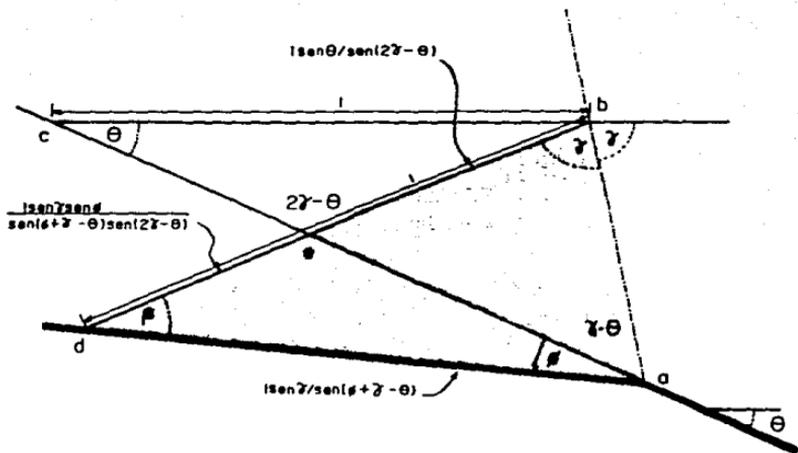


Figura 3.4 Relaciones geométricas del flanco alto de un pliegue-rampa.
(Según Suppe, 1983, Fig. 6).

$$\overline{od} = \frac{\ell \text{sen} \gamma \text{sen} \phi}{\text{sen}(\phi + \gamma - \theta) \text{sen}(2\gamma - \theta)} \dots (3.11)$$

sumando las ecuaciones (3.10) y (3.11)

$$\overline{be} + \overline{od} = \ell = \frac{\ell \text{sen} \theta}{\text{sen}(2\gamma - \theta)} + \frac{\ell \text{sen} \gamma \text{sen} \phi}{\text{sen}(\phi + \gamma - \theta) \text{sen}(2\gamma - \theta)} \dots (3.12)$$

utilizando la siguiente relación trigonométrica:

$$\text{sen}(\phi + \gamma - \theta) = \text{sen} \phi \cos(\gamma - \theta) + \cos \phi \text{sen}(\gamma - \theta) \dots (3.13)$$

y sustituyendo en (3.12) y dividiendo entre ℓ queda:

$$1 = \frac{\text{sen} \theta}{\text{sen}(2\gamma - \theta)} + \frac{\text{sen} \gamma \text{sen} \phi}{(\text{sen} \phi \cos(\gamma - \theta) + \cos \phi \text{sen}(\gamma - \theta)) \text{sen}(2\gamma - \theta)} \dots (3.14)$$

multiplicando por (3.13) y dividiendo entre $\text{sen} \phi$

$$\cos(\gamma - \theta) + \frac{\text{sen}(\gamma - \theta)}{\tan \phi} = \frac{(\cos(\gamma - \theta) + \text{sen}(\gamma - \theta) / \tan \phi) \text{sen} \theta}{\text{sen}(\gamma - \theta)} + \frac{\text{sen} \gamma}{\text{sen}(2\gamma - \theta)} \dots (3.15)$$

despejando $\tan \phi$

$$\tan \phi = - \frac{\text{sen}(\gamma - \theta)(\text{sen}(2\gamma - \theta) - \text{sen} \theta)}{\cos(\gamma - \theta)(\text{sen}(2\gamma - \theta) - \text{sen} \theta) - \text{sen} \gamma} \dots (3.16)$$

De la figura también se observa que:

$$\beta = 180^\circ - (2\gamma + \phi - \theta) \dots (3.17)$$

Las relaciones (3.16) y (3.17) constituyen un sistema de dos ecuaciones lineales con 4 incógnitas. Ahora, si consideramos el caso más sencillo, de un décollement con un sólo escalón, se tiene que $\phi = \theta$. Sustituyendo la igualdad anterior en (3.16) se tiene:

$$1 = \frac{\text{sen}\theta}{\text{sen}(2\gamma-\theta)} + \frac{\text{sen}\gamma\text{sen}\theta}{(\text{sen}\theta\cos(\gamma-\theta)+\cos\theta\text{sen}(\gamma-\theta))\text{sen}(2\gamma-\theta)}$$

... (3.18)

factorizando y despejando $\text{sen}(2\gamma-\theta)$

$$\text{sen}(2\gamma-\theta) = \text{sen}\theta + \frac{\text{sen}\gamma\text{sen}\theta}{\text{sen}(\theta+\gamma-\theta)} \dots (3.19)$$

simplificando:

$$\text{sen}(2\gamma-\theta) = 2\text{sen}\theta \dots (3.20)$$

aplicando una relación similar a (3.13)

$$\text{sen}2\gamma\cos\theta - \cos2\gamma\text{sen}\theta = 2\text{sen}\theta \dots (3.21)$$

dividiendo entre $\text{sen}\theta$

$$\frac{\text{sen}2\gamma}{\tan\theta} - \cos2\gamma = 2 \dots (3.22)$$

despejando $\tan\theta$ y sustituyendo $\cos2\gamma = 2\cos^2\gamma-1$ en (3.22)

$$\tan\theta = \tan\phi = \frac{\text{sen}2\gamma}{2\cos^2\gamma-1} \dots (3.23)$$

y sustituyendo la igualdad $\theta = \phi$ en (3.17) se obtiene:

$$\beta = 180^\circ - 2\gamma \dots (3.24)$$

Las ecuaciones (3.23) y (3.24) son las relaciones que guardan θ , γ y β entre sí, y gobiernan la geometría de un décollement con un salto sencillo. Un ejemplo real que concuerda con esta parte del modelado es la Cabalgadura de Xilitla en la Sierra Madre Oriental del centro-oriente de México (Suter, 1984b).

La eficacia del método radica en un buen control geológico superficial e información del subsuelo. Con estos datos y por medio de interpolación y extrapolación se determinan dos parámetros, ya que se trata de un sistema de dos ecuaciones con 4

incógnitas; generalmente γ , que domina la forma del pliegue, y θ , el ángulo de corte, o bien β , que puede ser más sencillo determinar. Estos se sustituyen en las ecuaciones que correspondan según el caso. El método se puede usar para estimar el echado de segmentos no expuestos de cabalgaduras. En este caso, el objetivo es el mismo que en el capítulo 3.3, sin embargo, los dos métodos son independientes. Si los resultados no concuerdan con los datos reales, entonces el plegamiento no corresponde a un ambiente paralelo o la forma es demasiado compleja para ser precedida por las ecuaciones.

3.5 APLICACION PARA FALLAS NORMALES

La construcción de secciones balanceadas se desarrolló inicialmente para afallamiento y plegamiento relacionado con acortamiento, manteniéndose fuera de consideración las deformaciones relacionadas con extensión. Sin embargo, los terrenos distensivos están sujetos a las mismas consideraciones que los terrenos compresivos y se pueden efectuar simplificaciones idénticas, si se cumplen las restricciones a que están sujetos.

Gibbs (1982) aplica el modelo de Chamberlain (1910) para calcular la profundidad al despegue en zonas con fallamiento normal. La Figura 3.5 muestra la aplicación del principio de conservación de área y la relación entre el acortamiento y la profundidad al despegue basal. El área ganada horizontalmente (área B en la Figura 3.5) es igual al área perdida verticalmente (A). B se puede expresar como la extensión por la profundidad al despegue (d); entonces

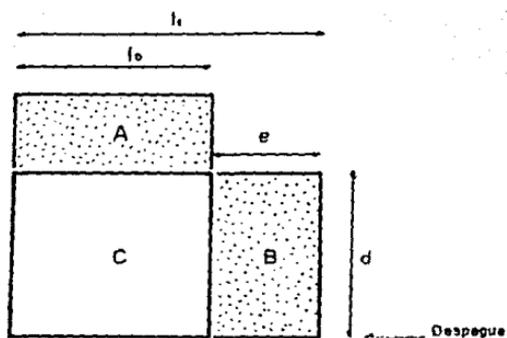
$$B = A = d \cdot (l_0 - l_1) \dots (3.25)$$

despejando d

$$d = A / (l_0 - l_1) \dots (3.26)$$

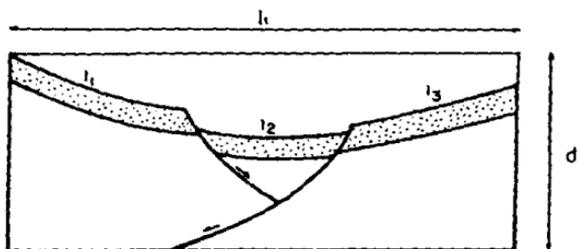
Que es la relación (3.5)

El método se ha usado en "rifts" intracontinentales, como por ejemplo en los campos petroleros del Mar del Norte (Gibbs, 1982).



$$\text{Area A} = \text{Area B} : e \cdot d = A = B$$

$$(l_1 - l_0)d = A = B$$



$$l_0 = \sum l_i$$

$$(l_1 - l_0)d = A$$

$$d = A / (l_1 - l_0)$$

Figura 3.5 Cálculo de la profundidad al despegue en terrenos distensivos. (Según Gibbs, 1983, Fig. 1).

4. CINEMATICA DE LA DEFORMACION

4.1 LA DEFORMACION COMO UNA SERIE EN EL TIEMPO

La condición instantánea de esfuerzo y deformación de un medio rocoso es conocida como el estado mecánico del sistema. El estado mecánico a un instante determinado, está caracterizado por las fuerzas que actúan sobre cada parte del sistema y por los desplazamientos que producen así como por la ley constitutiva, es decir, el mecanismo de deformación que relaciona las fuerzas con las reacciones. Para describir cuantitativamente cualquiera de esas características, se necesita de un marco de referencia, colocado usualmente en alguna parte considerada inmovil. Un sistema se considera descrito si conocemos las fuerzas que actúan (en posición, magnitud y dirección) y la posición que ocupa cada partícula del medio, al ocurrir un cambio en las condiciones mecánicas del medio.

El cambio en el estado mecánico de un sistema se refiere a las diferencias entre dos condiciones instantáneas, transcurrido un intervalo de tiempo entre ambas. Estas diferencias quedan establecidas por reacciones que origina el campo de esfuerzos en la geometría y dinámica del medio. Estas incluyen desplazamientos, distorsiones y cambios en el estado de esfuerzos.

Las características estructurales observadas en una secuencia plegada están genéticamente relacionadas con la disposición de las fuerzas que actúan sobre las rocas en el espacio y en el tiempo; la evolución mecánica. Dentro de la historia mecánica de una secuencia se estudian dos aspectos: la disposición de las fuerzas en el tiempo y la deformación que estas generan.

Se puede considerar que las estructuras se desarrollan a partir de una serie de pequeños ajustes que tienen lugar durante un período finito de tiempo. Algunas estructuras se desarrollan más rápidamente que otras, pero todas son producto de la evolución cinemática. Esta serie de estados es referida como la secuencia que sigue la deformación. La descripción más completa posible no sólo especifica la secuencia, sino también el tiempo en que se verifica cada estado dentro de la evolución, estableciendo la secuencia fechada de deformación.

La Figura 4.1 ilustra cuatro aspectos relacionados al movimiento de un sistema de partículas. El estado instantáneo

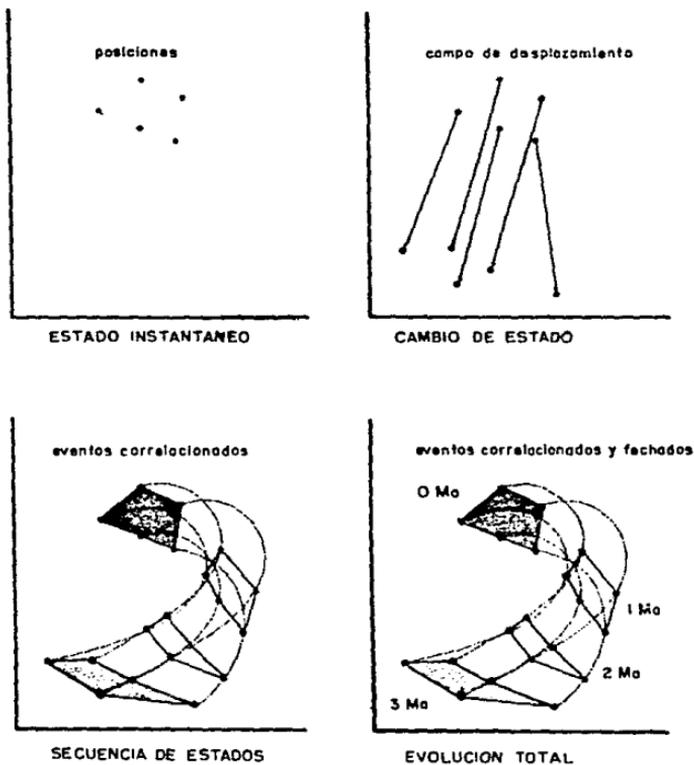


Figura 4.1 Parámetros de correlación entre el estado inicial y el estado final deformado. (Según Means, 1976, Fig. 4.2).

inicial es descrito por la posición de todas las partículas en un momento dado. El cambio en el estado mecánico lo indica el campo de desplazamiento, relacionando la posición de las partículas en los estados instantáneos inicial y final. La secuencia progresiva de estados deformados está indicada por las trayectorias de las partículas. Finalmente, la evolución total está indicada por la familia de trayectorias fechadas de las partículas.

En las estructuras deformadas, por lo general sólo se cuenta con indicadores de los estados extremos. Dentro de estos, hay dos tipos: (i) los indicadores de distorsión, como son los calciesférulos y crinoides elípticos, originalmente circulares, y los estratos plegados que inicialmente fueron de formas planar. (ii) Los indicadores de movimiento, como las fallas y estructuras relacionadas al movimiento a lo largo de estas. Ocasionalmente se cuenta con indicadores de la evolución total, como en algunos casos del registro paleomagnético.

Al interpretar los indicadores, es importante reconocer las limitaciones de los datos que proporcionan. De esta manera, al observar una estructura que señale dos fases, es necesario obtener otras evidencias para reconstruir los estados intermedios de deformación. Por ejemplo, el desplazamiento lateral de una falla de rumbo puede interpretarse debido a una serie de movimientos pequeños continuos o, alternativamente, por una serie de desplazamientos discretos, que involucran grandes movimientos y acompañados de períodos inactivos. Otro ejemplo es un crinoide con forma elíptica con eje menor orientado Este-Oeste, se interpreta que ha sido acortado un campo de esfuerzos compresivos Este-Oeste, o también, por acortamiento Norte-Sur, seguido por una rotación de 90°. Las secuencias de deformación, en la mayoría de las veces, no pueden establecerse determinísticamente. Pueden existir una serie de secuencias alternas posibles, mismas que pueden ser correctas.

4.2 RELACIONES ENTRE EL ESTADO INICIAL Y EL ESTADO FINAL DEFORMADO

Existen dos maneras de describir el movimiento de un continuo: a) Una descripción referencial, cuyas variables independientes son la posición X de la partícula, dentro de un sistema referencial arbitrario, y el tiempo t . b) Una descripción

espacial, cuyas variables independientes son la posición x ocupada por la partícula en el tiempo presente t .

La descripción referencial se relaciona con el movimiento de un sistema de referencia inicial, en el cual una partícula P ocupa la posición X . En la teoría de elasticidad, la configuración de referencia, es el estado inicial, sin compresión (en el tiempo $t=0$), llamándose a este tipo "descripción lagrangiana". En la descripción espacial no importa la posición inicial que ocupó la partícula P , sólo su posición x en el tiempo presente. Este tipo de descripción es muy utilizada en la mecánica de fluidos y es llamada la "descripción euleriana". Ambas descripciones se relacionan de la siguiente manera (Figura 4.2)

$$x = x(X) \dots (4.1)$$

que en componentes rectangulares queda

$$\bar{x} = \bar{x}(X_1, X_2, X_3) \dots (4.2)$$

donde X_i son las coordenadas materiales, mientras que \bar{x} son las coordenadas espaciales en el tiempo t . Es decir, x es la posición ocupada en el tiempo t por la partícula P , que ocupó la posición X en la configuración inicial de referencia.

A partir de estas relaciones, entre marcos de referencia, se pueden obtener las siguientes matrices, al operar el jacobiano

$$J\left(\frac{U_i}{x_j}\right) = \left(\frac{\partial U_i}{\partial x_j}\right) = E_{i,j} \dots (4.3)$$

$$J\left(\frac{U_i}{\bar{x}_j}\right) = \left(\frac{\partial U_i}{\partial \bar{x}_j}\right) = e_{i,j} \dots (4.4)$$

$$J\left(\frac{x_i}{x_j}\right) = \left(\frac{\partial x_i}{\partial x_j}\right) = C_{i,j} \dots (4.5)$$

$$J\left(\frac{X_i}{\bar{x}_j}\right) = \left(\frac{\partial X_i}{\partial \bar{x}_j}\right) = c_{i,j} \dots (4.6)$$

donde $x_1=x$, $x_2=y$, $x_3=z$ y $X_1=X$, $X_2=Y$, $X_3=Z$ y U es el vector de desplazamiento relativo.

La ecuación (4.3) es el gradiente de desplazamiento referido al estado sin deformar y es conocido como el tensor lagrangiano. (4.4) representa el gradiente desplazamiento referido al estado

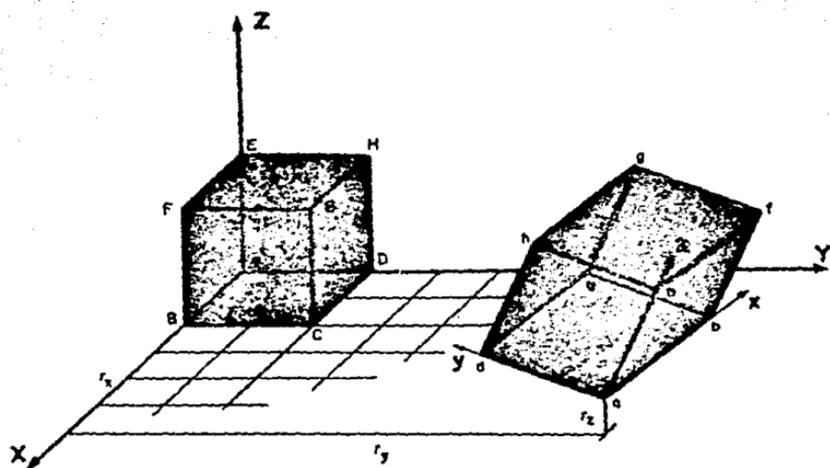


Figura 4.2 Deformación de un cubo en el espacio.

deformado o tensor euleriano. Las relaciones (4.5) y (4.6) son el gradiente de deformación respecto a ambos sistemas y son conocidos como los tensores de Green y Cauchy, respectivamente. Con estas cuatro ecuaciones se tiene totalmente establecida la deformación, como se muestra en el ejemplo siguiente.

Supongase una transformación cuya relación entre las coordenadas materiales y especiales sea la siguiente

$$\begin{aligned}x &= -a \cdot X + b \cdot Y + 0 \cdot Z - r_X \\y &= c \cdot X - d \cdot Y + 0 \cdot Z - r_Y \\z &= 0 \cdot X + e \cdot Y + f \cdot Z - r_Z\end{aligned}$$

los vectores de desplazamiento relativo están determinados por

$$\begin{aligned}U_x &= x - X = -a \cdot X + b \cdot Y - r_X - X \\U_y &= y - Y = c \cdot X + d \cdot Y - r_Y - Y \\U_z &= z - Z = e \cdot Y + f \cdot Z - r_Z - Z\end{aligned}$$

Obteniendo cada uno de los gradientes, se tiene

$$J\left(\frac{U_i}{X_j}\right) = \begin{bmatrix} -a-1 & b & 0 \\ c & -d-1 & 0 \\ 0 & e & f-1 \end{bmatrix}$$

$$J\left(\frac{U_i}{X_j}\right) = \begin{bmatrix} cb/d - cb/ad & cb^2/ad + b & 0 \\ c^2a/(ad+bc) & dac/(ad+bc) & 0 \\ 0 & (e^2/f)ca/(ad+bc) & 1 + 1/f \end{bmatrix}$$

$$J\left(\frac{x_i}{X_j}\right) = \begin{bmatrix} -a & b & 0 \\ c & -d & 0 \\ 0 & e & f \end{bmatrix}$$

$$J\left(\frac{X_i}{X_j}\right) = \begin{bmatrix} cb/ad & cb/ad + 1 & 0 \\ ca/(ad+bc) & ac/(ad+bc) & 0 \\ 0 & (e/f)ca/(ad+bc) & 1/f \end{bmatrix}$$

Del tensor lagrangiano se obtienen la matriz de la distorsión E y la matriz de rotación Ω de cuerpo rígido, notese que $E + \Omega = E_{ij}$

$$E = \begin{bmatrix} -a-1 & 1/2(b+c) & 0 \\ 1/2(c+b) & -d-1 & 1/2e \\ 0 & 1/2e & f \end{bmatrix}$$

$$\Omega = \begin{bmatrix} 0 & 1/2(b-c) & 0 \\ 1/2(c-b) & 0 & -1/2e \\ 0 & 1/2e & 0 \end{bmatrix}$$

la razón local de volumen después de la deformación está dado por el determinante de $J(x_i/X_j)$

$$|J(x_i/X_j)| = a \cdot d \cdot f - c \cdot b \cdot e$$

la elongación unitaria relativa dU/dS a lo largo de la dirección $X = (1,0,0)$ está dado por:

$$dU/dS = E \cdot X^T = \begin{bmatrix} -a-1 \\ 1/2(c+b) \\ 0 \end{bmatrix}$$

y la elongación neta es

$$\sqrt{(dU_x/dS)^2 + (dU_y/dS)^2 + (dU_z/dS)^2} = \sqrt{(-a-1)^2 + 1/4(c+b)^2}$$

de igual manera, para las direcciones $Y = (0,1,0)$ y $Z = (0,0,1)$, la elongación neta es:

$$\sqrt{(dU_x/dS)^2 + (dU_y/dS)^2 + (dU_z/dS)^2} = \sqrt{1/4(c+b)^2 + (1-d)^2 + 1/4e^2}$$

$$\sqrt{(dU_x/dS)^2 + (dU_y/dS)^2 + (dU_z/dS)^2} = \sqrt{1/4e^2 + f^2}$$

Para obtener formas más complejas con distorsión heterogénea, sólo hay que introducir términos no lineales, o utilizar interpoladores para aproximar la forma distorsionada. Sin embargo, la posibilidad de describir geométricamente una deformación no indica que esta sea mecánicamente posible.

5. MODELADO DIRECTO DE SECUENCIAS DE DEFORMACION.

5.1 PRESENTACION DEL ALGORITMO.

El objetivo de este trabajo es presentar un algoritmo para modelar numéricamente la deformación bidimensional de una secuencia estratigráfica representada por una red de nodos regularmente espaciada, partiendo de la geometría inicial y de una configuración específica de fallas. Es decir, el algoritmo simula secuencias de deformación propuestas, en sección, asociadas a fallas normales o cabalgaduras. El algoritmo ejecuta la cinemática de la secuencia propuesta, y el usuario puede completar y cuantificar la secuencia de eventos, eliminando discrepancias entre los resultados sintéticos y los datos reales a través de ensayo y error.

El algoritmo está basado en las siguientes consideraciones:

- i) La geometría de la fallas está compuesta de segmentos de líneas rectas. Segmentos curvados se pueden aproximar con segmentos de rectas.
- ii) La magnitud de la trayectoria del desplazamiento es constante para todos los nodos.
- iii) Todas las partículas se desplazan paralelamente a los tramos rectos que componen a la falla.
- iv) El desplazamiento está contenido en el plano de la sección.

Estas consideraciones repercuten en la geometría generada por el algoritmo durante la simulación de la deformación. Estas repercusiones son las siguientes:

- 1) Dado que las fallas están constituidas por segmentos rectos, la geometría del medio está gobernada por pliegues angulares. Por lo tanto, cada segmento recto forma un dominio de echados constantes que, como se explicó en el capítulo 2.3, está limitado por las bisectrices del ángulo formado por dos tramos vecinos con echados diferentes. Estos dominios también están caracterizados por desplazamientos o campos de desplazamientos constantes.
- 2) Como efecto de la subdivisión del medio en dominios de echados y desplazamientos constantes se genera distorsión, que esta en función del ángulo formado por las rectas adyacentes, como se demostrará más adelante.

Considérese el nodo P de coordenadas (X, Y) , que se encuentra sobre el segmento recto f_1 de la falla que corta al continuo representado por la retícula (Figura 5.1a). La partícula P se

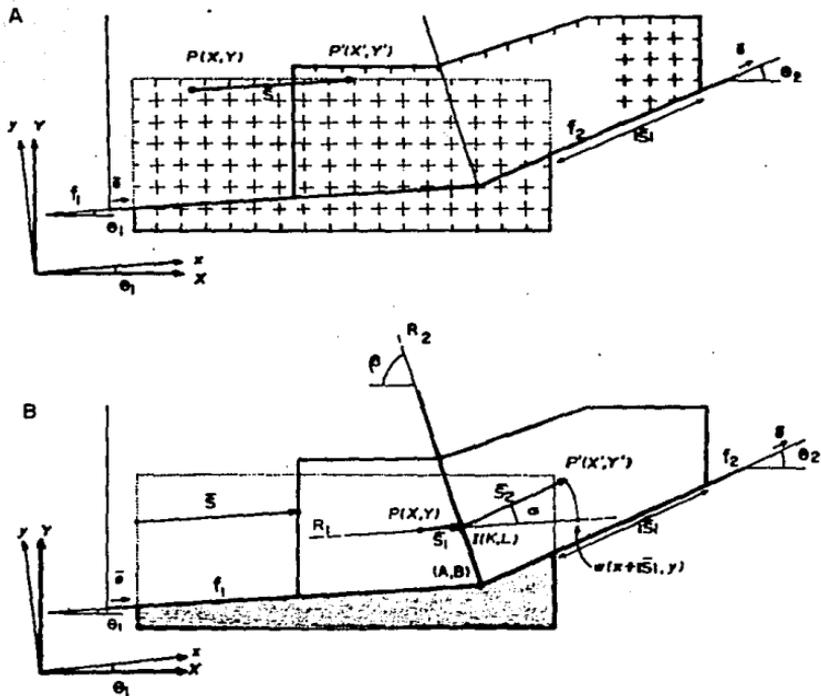


Figura 5.1a Desplazamiento de un nodo P a P' dentro de su propio dominio.

Figura 5.1b Desplazamiento de un nodo P a P' en el dominio vecino.

desliza una distancia $|\bar{s}|$, constante para cualquier partícula, sobre f_1 hasta llegar a P' , de coordenadas (X', Y') . Si la distancia entre el punto P y el eje axial es mayor que la magnitud del desplazamiento, el nodo desplazado sigue dentro del mismo dominio de echados.

La relación entre P y P' es:

$$\left. \begin{aligned} X' &= X + |\bar{s}| \cos \theta_1 \\ Y' &= Y + |\bar{s}| \sin \theta_1 \end{aligned} \right\} \dots (5.1)$$

Si consideramos el nuevo sistema coordenado xy rotado con respecto a XY el ángulo θ_1 , como lo muestra la Figura 5.1, se tiene que los nodos P y P' tendrán en el nuevo sistema las coordenadas $p(x, y)$ y $p'(x', y')$ respectivamente, y tienen la siguiente relación:

$$\left. \begin{aligned} x' &= x + |\bar{s}| \\ y' &= y \end{aligned} \right\} \dots (5.2)$$

Las ecuaciones (5.1) y (5.2) son las expresiones que controlan el deslizamiento para un nodo que se desplaza dentro de su dominio.

Se observa que el vector unitario de desplazamiento es:

$$\bar{s} = \cos \theta_1 \bar{i} + \sin \theta_1 \bar{j} \quad (5.3)$$

donde \bar{i} y \bar{j} son los vectores base unitarios, por lo que las ecuaciones (5.1) se pueden expresar:

$$\left. \begin{aligned} X &= X + |\bar{s}| \cdot \bar{s}_x \\ Y &= Y + |\bar{s}| \cdot \bar{s}_y \end{aligned} \right\} \dots (5.4)$$

lo que otorga a la ecuación (5.1) un sentido cinemático.

Considerese ahora un nodo en la vecindad de un eje axial, como lo muestra la Figura 5.1b. Al desplazarse una distancia $|\bar{s}_1|$ sobre f_1 , la partícula cambia de dominio, por lo que hay que considerar dos recorridos rectos \bar{s}_1 y \bar{s}_2 paralelos a f_1 y f_2 respectivamente, tal que:

$$\vec{s} = \vec{s}_1 + \vec{s}_2$$

Sin embargo, ejecutar esta descomposición en dos vectores para toda la malla es analíticamente complicada, ya que s_1 y s_2 dependen de la posición de la partícula.

Otra forma de apreciar la transformación es considerando a $P'(X', Y')$ como la rotación de P (o en la Figura 5.1b, ya que está expresado en el sistema xy) un ángulo $\alpha = \theta_2 - \theta_1$ alrededor de $I(K, L)$, como se muestra en la Figura 5.1b. Si $I(K, L)$ coincide con el origen del sistema, la rotación se expresa matricialmente:

$$(X', Y') = (X + |\vec{s}| \cos \theta_1, Y + |\vec{s}| \sin \theta_1) \cdot \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} \dots (5.5)$$

Cuando el centro de rotación se encuentra fuera del origen del sistema, hay que efectuar transformaciones adicionales. Hay que trasladar el origen al centro de rotación, efectuar la rotación y finalmente trasladarlo nuevamente a su posición original.

Lo anterior se expresa matricialmente :

$$(X' \ Y' \ 1)^T = \begin{bmatrix} X + |\vec{s}| \cos \theta_1 \\ Y + |\vec{s}| \sin \theta_1 \\ 1 \end{bmatrix}^T \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -K & -L & 1 \end{bmatrix} \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ K & L & 1 \end{bmatrix} \dots (5.6)$$

donde el supraíndice (T) indica la matriz traspuesta.

Se observa que se ha introducido una columna y un renglón adicionales para hacer compatibles las matrices.

Desarrollando la ecuación (5.6) para los respectivos sistemas coordenados:

$$\begin{aligned} X' &= (X + |\vec{s}| \cos \theta_1) \cos \alpha - (Y + |\vec{s}| \sin \theta_1) \sin \alpha - K(\cos \alpha - 1) + L \sin \alpha \\ Y' &= (X + |\vec{s}| \cos \theta_1) \sin \alpha + (Y + |\vec{s}| \sin \theta_1) \cos \alpha - K \sin \alpha - L(\cos \alpha - 1) \end{aligned} \quad (5.7)$$

y

$$\begin{cases} x' = (x + |\bar{r}|) \cos \alpha - y \sin \alpha - K(\cos \alpha - 1) + L \sin \alpha \\ y' = (x + |\bar{r}|) \sin \alpha + y \cos \alpha - L \sin \alpha - K(\cos \alpha - 1) \end{cases} \dots (5.8)$$

Por otra parte, como se explicó, K y L varían de acuerdo a la posición de $P(X, Y)$, entonces:

$$\begin{cases} K = K(X, Y) \\ L = L(X, Y) \end{cases} \dots (5.9)$$

Geométricamente $I(K, L)$ se pueden considerar como la intersección de las rectas R_1 y R_2 , como se muestra en la Figura 5.2., aunque en realidad R_1 es una familia de rectas. Las expresiones para R_1 y R_2 son:

$$\begin{cases} R_1 = m_1 X + C_1 \\ R_2 = m_2 X + C_2 \end{cases} \dots (5.10)$$

donde $m_1 = \tan \theta_1$, $m_2 = \tan \beta$ y C_1, C_2 son los términos constantes de las rectas

$$\begin{cases} C_1 = Y - m_1 X \\ C_2 = B - m_2 A \end{cases} \dots (5.11)$$

Sustituyendo (5.11) en (5.10) y expresándolo en forma de matriz

$$\begin{bmatrix} 1 & -m_1 \\ 1 & -m_2 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} Y - m_1 X \\ B - m_2 A \end{bmatrix} \dots (5.12)$$

Resolviendo el sistema por el método de Cramer se tiene:

$$\begin{cases} K = (B - m_2 A - Y + m_1 X) / (m_1 - m_2) \\ L = ((B - m_2 A)m_1 - (Y - m_1 X)m_2) / (m_1 - m_2) \end{cases} \dots (5.13)$$

que son las ecuaciones del lugar geométrico que forman las intersecciones de la familia de rectas R_1 con la recta R_2 . Expresadas respecto a xy son:

$$k = (y + m_2^1 a - b) / m_2^1 \quad \left. \begin{array}{l} \\ z = y \end{array} \right\} \dots (5.14)$$

donde: $m_2^1 = \tan(\beta - \theta_1) \dots (5.15)$

Sustituyendo (5.13) y (5.14) en (5.7) y (5.8) respectivamente

$$X' = (X + |\bar{u}| \cos \theta_1) \operatorname{sen} \alpha - (Y + |\bar{u}| \operatorname{sen} \theta_1) \operatorname{cose} \alpha - \\ (B - m_2 A - Y + m_1 X) \cdot \frac{\operatorname{cose}(\alpha - 1)}{m_1 - m_2} + ((B - m_2 A) m_1 - (Y - m_1 X) m_2) \cdot \frac{\operatorname{sen} \alpha}{m_1 - m_2} \dots (5.16a)$$

$$Y' = (X + |\bar{u}| \cos \theta_1) \operatorname{cose} \alpha + (Y + |\bar{u}| \operatorname{sen} \theta_1) \operatorname{cose} \alpha - \\ (B - m_2 A - Y + m_1 X) \cdot \frac{\operatorname{sen} \alpha}{m_1 - m_2} - ((B - m_2 A) m_1 - (Y - m_1 X) m_2) \cdot \frac{\operatorname{cose}(\alpha - 1)}{m_1 - m_2} \dots (5.16b)$$

y

$$\left. \begin{array}{l} x' = (x + |\bar{u}|) \operatorname{cose} \alpha - (y' + m_2^1 a - b) \cdot (\operatorname{cose} \alpha - 1) / m_2^1 \\ y' = (x + |\bar{u}|) \operatorname{sen} \alpha - (y' + m_2^1 a - b) \cdot (\operatorname{sen} \alpha) / m_2^1 + y \end{array} \right\} \dots (5.17)$$

(5.16) y (5.17) son las expresiones para calcular la nueva posición del nodo cuando cambia de dominio.

Considerese ahora, un desplazamiento adicional por cizalla simple sobre el medio continuo desplazado, como lo muestra la Figura 5.2. El desplazamiento w_x del punto P está en función del ángulo ψ y de la altura sobre Q. De esta manera se tiene:

$$|\bar{w}_x| = h \cdot \tan \psi \dots (5.18)$$

pero de la Figura 5.2 se tiene:

$$h = |P - Q|$$

donde Q tiene coordenadas (X_Q, Y_Q) , entonces

$$h = \sqrt{(X - X_Q)^2 + (Y - Y_Q)^2} \dots (5.19)$$

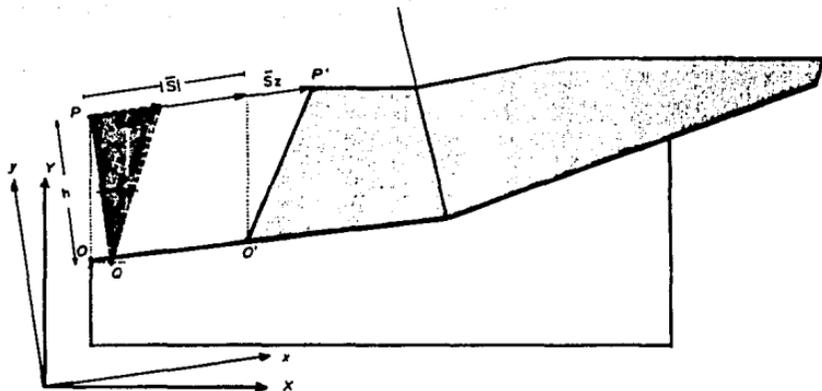


Figura 5.2 Deformación de una cobijadura por cizalla simple.

o de acuerdo a los ejes xy (notese el cambio de los subíndices a minúsculas, indicando la transformación de coordenadas de Q a q):

$$h = y - y_q \dots (5.20)$$

las magnitudes de las componentes de \bar{s}_x son:

$$\left. \begin{aligned} |\bar{s}_{x_x}| &= |\bar{s}_x| \cos\theta_1 \\ |\bar{s}_{x_y}| &= |\bar{s}_x| \sin\theta_1 \end{aligned} \right\} \dots (5.21)$$

y

$$\left. \begin{aligned} |\bar{s}_{y_x}| &= (y-y_q) \tan\psi \\ |\bar{s}_{y_y}| &= 0 \end{aligned} \right\} \dots (5.22)$$

Sustituyendo (5.19) y (5.18) en (5.21) y sumandose a (5.1) se tiene:

$$\left. \begin{aligned} X' &= X + (|\bar{s}| + \sqrt{(X-X_q)^2 + (Y-Y_q)^2} \cdot \tan\psi) \cdot \cos\theta_1 \\ Y' &= Y + (|\bar{s}| + \sqrt{(X-X_q)^2 + (Y-Y_q)^2} \cdot \tan\psi) \cdot \sin\theta_1 \end{aligned} \right\} \dots (5.23)$$

$$\left. \begin{aligned} x' &= x + |\bar{s}| + (y-y_q) \tan\psi \\ y' &= y \end{aligned} \right\} \dots (5.24)$$

de igual manera, al agregarse deslizamiento por cizalla simple a las expresiones (5.17), se obtiene:

$$\left. \begin{aligned} x' &= (x + |\bar{s}| + (y-y_q) \tan\psi) \cos\alpha - (y + m'_2 a - b) \cdot (\cos\alpha - 1) / m'_2 \\ y' &= (x + |\bar{s}| + (y-y_q) \tan\psi) \sin\alpha - (y + m'_2 a - b) \cdot \sin\alpha / m'_2 + y \end{aligned} \right\} (5.25)$$

Las ecuaciones (5.25) son las expresiones generales que controlan el desplazamiento de un nodo sobre la configuración de una falla, de acuerdo a un sistema de referencia xy rotado con un ángulo θ_1 , paralelo a un segmento de la falla.

La componente de cizallamiento simple está distribuida de manera homogénea dentro del medio desplazado, lo que corresponde en la naturaleza a un tipo de plegamiento denominado "flexural-flow folding" en la literatura (Ramsay y Huber, 1987). La componente también se podría introducir para un cizallamiento

simple, limitado a los planos entre las capas, que no afecta al interior de los estratos. Este tipo de deformación se denomina "flexural-slip folding" en la literatura (Ramsay y Huber, 1987). Sin embargo la simulación de este tipo de estilo estructural implica cambios en la formulación del tratamiento matemático introducido. Otra posibilidad sería distribuir el desplazamiento no solo dentro del flanco desplazado a lo largo de la falla, sino dentro de todo el medio, incluyendo el flanco bajo de la falla (Geiser, 1988).

Como se indicó al inicio de la presentación del modelo, se genera una distorsión neta durante el desplazamiento, lo que se puede apreciar en las Figuras 5.3. Esta distorsión de la malla se origina por el transporte de nodos de la malla a través de planos axiales. El campo de los vectores resultantes de desplazamientos es inhomogéneo a través de los planos axiales, puesto que se desarrollan diferencias entre los vectores tanto en sus magnitudes como en sus direcciones (Figura 5.3b).

La distorsión que sufre un elemento lineal dentro del medio, generada por el modelo (Figura 5.4) está determinada por:

$$E = \frac{|K'P' - KP|}{|KP|} \dots (5.26)$$

Ahora bien, considérese dos rectas que unen a los puntos P,Q y P,O (Figura 5.4a), en el estado inicial, y P',Q' y P',O' en el estado final deformado. Entonces el ángulo ϕ , formado por las rectas, lo determina la expresión:

$$\phi_i = \text{angcos} \left(\frac{QP \cdot OP}{|QP| |OP|} \right) \dots (5.27)$$

$$\phi_f = \text{angcos} \left(\frac{Q'P' \cdot O'P'}{|Q'P'| |O'P'|} \right) \dots (5.28)$$

Donde ϕ_i y ϕ_f son los ángulos en el estado inicial y final deformado, respectivamente.

De manera similar, el cambio por unidad de área que introduce el modelo está dado por:

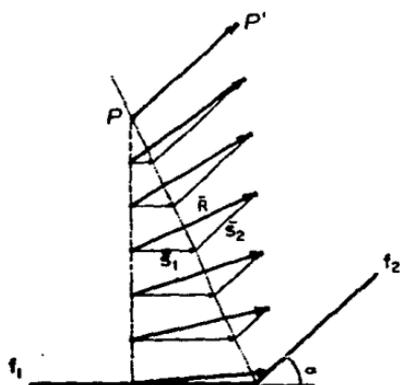
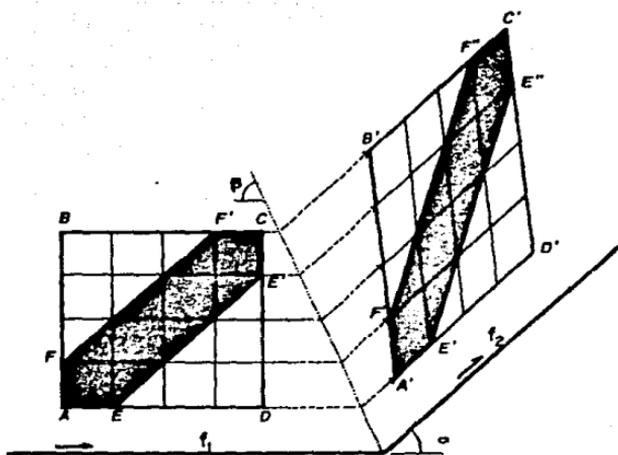


Figura 5.3a Distorsión originada sobre un elemento de área cuadrangular provocada por el transporte a través de un plano axial.

Figura 5.3b Campo de vectores de desplazamiento heterogéneo a través de un plano axial. Se aprecian diferencias tanto en la magnitud como en la dirección de los vectores.

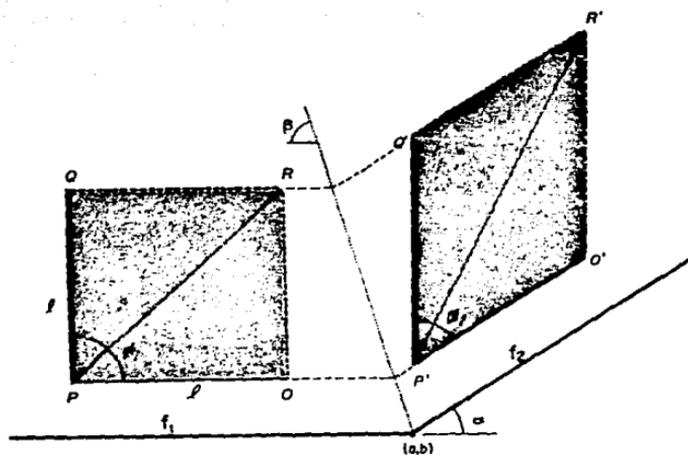


Figura 5.4 Relaciones entre los estados inicial y deformado de un elemento de área del modelo transportado a través de un plano axial, donde se aprecia la distorsión longitudinal y por cizallamiento angular.

$$AA = \frac{\begin{vmatrix} O'P'_X & O'P'_Y \\ O'P''_X & O'P''_Y \end{vmatrix} - \begin{vmatrix} OP_X & OP_Y \\ OP'_X & OP'_Y \end{vmatrix}}{\begin{vmatrix} OP_X & OP_Y \\ OP'_X & OP'_Y \end{vmatrix}} \dots (5.29)$$

donde los subíndices indican las componentes de los vectores.

De la Figura 5.4 se deduce que las relaciones entre los puntos Q, P y O, P son:

$$X_Q = X_P ; \quad Y_Q = Y_P + \ell \dots (5.30)$$

$$X'_O = X_P + \ell ; \quad Y'_O = Y_P \dots (5.31)$$

$$m = \tan((\alpha-180^\circ)/2) \dots (5.32)$$

Obteniendo la transformación de (5.30) y (5.31) de acuerdo a las expresiones (5.17), se obtiene:

$$X'_Q = (X_P + |\bar{S}|) \cos \alpha - (Y_P + \ell + b - ma) \cdot (\cos \alpha - 1) / m$$

$$Y'_Q = (X_P + |\bar{S}|) \operatorname{sen} \alpha - (Y_P + \ell + b - ma) \cdot \operatorname{sen} \alpha / m + Y_P + \ell$$

y

$$X'_O = (X_P + \ell + |\bar{S}|) \cos \alpha - (Y_P + b - ma) \cdot (\cos \alpha - 1) / m$$

$$Y'_O = (X_P + \ell + |\bar{S}|) \operatorname{sen} \alpha - (Y_P + b - ma) \cdot \operatorname{sen} \alpha + Y_P$$

Entonces, la elongación a lo largo de los ejes X y Y está dado por:

$$E_X = \frac{|O'P'_X| - |OP_X|}{|OP_X|} = \frac{\sqrt{\ell^2 \cos^2 \alpha + \ell^2 \operatorname{sen}^2 \alpha} - \ell}{\ell} = 0 \dots (5.33)$$

$$E_Y = \frac{|O'P'_Y| - |OP_Y|}{|OP_Y|} = \frac{\sqrt{(-\ell(\cos \alpha - 1)/m)^2 + (-\ell \operatorname{sen} \alpha / m + \ell)^2} - \ell}{\ell}$$

Desarrollando y simplificando la ecuación anterior

$$E_Y = \sqrt{(1 - \cos \alpha - m \cdot \operatorname{sen} \alpha)^2 / m^2} + 1 - 1$$

Sustituyendo $\cos \alpha = \cos^2 \alpha / 2 - \operatorname{sen}^2 \alpha / 2$

$$\operatorname{sen} \alpha = 2 \operatorname{sen} \alpha / 2 \cdot \cos \alpha / 2$$

$$m = \tan((\alpha-180^\circ)/2) = -\cos \alpha / 2 / \operatorname{sen} \alpha / 2$$

en la expresión anterior y simplificando:

$$E_y = \sqrt{(2\text{sen}^2\alpha/2 + 2\text{cos}^2\alpha/2) \cdot 2/m^2 + 1} - 1$$

$$E_y = \sqrt{4/m^2 + 1} - 1 \dots (5.34)$$

o expresandola en función de α

$$E_y = \sqrt{4\tan^2\alpha/2 + 1} - 1 \dots (5.35)$$

Las ecuaciones (5.33) y (5.35) determinan la elongación que introduce el modelo en el medio.

Si se consideran las rectas ortogonales \overline{OP} y $\overline{OP'}$ la distorsión que sufre el ángulo ϕ durante el proceso es (de acuerdo a (5.28))

$$\phi_f = \text{angcos} \left(\frac{(-\ell(\text{cosa}-1)/m_i + (\ell-\ell\text{sen}\alpha/m)) \cdot (\ell\text{cosa} + \ell\text{sen}\alpha)}{\ell^2 \cdot (E_y + 1)} \right)$$

Desarrollando y simplificando la expresión anterior

$$\phi_f = \text{angcos} \left(\frac{-\text{cos}^2\alpha/m + \text{cosa}/m + \text{sen}\alpha - \text{cos}\alpha\text{sen}\alpha/m}{E_y + 1} \right)$$

Factorizando cosa

$$\phi_f = \text{angcos} \left(\frac{\text{cosa}(1-\text{sen}\alpha-\text{cosa}) + m\text{sen}\alpha}{m \cdot (E_y + 1)} \right) \dots (5.36)$$

sustituyendo m por $-\text{cosa}/2/\text{sen}\alpha/2$ y simplificando

$$\phi_f = \text{angcos} \left(\frac{\text{tana}/2\text{cosa}(\text{sen}\alpha+\text{cosa}-1) + 1/2\text{sen}\alpha}{E_y + 1} \right) \dots (5.37)$$

A partir de este ángulo se puede calcular la distorsión por cizallamiento angular ψ , el cambio de un ángulo originalmente de 90° .

$$\psi = 90^\circ - \phi_f \dots (5.38)$$

sustituyendo ϕ_f en (5.38)

$$\phi = \pi/2 - \arccos\left(\frac{\ell \tan \alpha / 2 \cos \alpha (\text{sen} \alpha + \text{cosen} \alpha - 1) + 1 / 2 \text{sen} \alpha}{E_y + 1}\right) \dots (5.39)$$

y la cantidad de cizallamiento está definido como:

$$\begin{aligned} \gamma &= \tan \phi \\ \gamma &= \tan(\pi/2 - \phi_f) \dots (5.40) \end{aligned}$$

Para encontrar el cambio por unidad de área se aplica la ecuación (5.29), sustituyendo queda:

$$\Delta A = \frac{\left| \begin{array}{cc} \ell \cos \alpha & \ell \text{sen} \alpha \\ -\ell(\cos \alpha - 1)/m & -\ell \text{sen} \alpha / m + \ell \end{array} \right|}{\ell^2} - \ell^2$$

desarrollando y simplificando:

$$\Delta A = \text{sen} \alpha (\cos \alpha - 1) / m - \cos \alpha \text{sen} \alpha / m + \cos \alpha - 1$$

simplificando la expresión anterior

$$\Delta A = \cos \alpha - \text{sen} \alpha / m - 1$$

efectuando una sustitución similar a la hecha en (5.34)

$$\Delta A = \cos^2 \alpha / 2 - \text{sen}^2 \alpha / 2 + 2 \text{sen}^2 \alpha / 2 - 1$$

simplificando

$$\Delta A = \cos^2 \alpha / 2 + \text{sen}^2 \alpha / 2 - 1 = 0 \dots (5.41)$$

por lo que, el cambio de área introducido por el modelo es nulo.

La Figura 5.3 muestra el comportamiento de las ecuaciones (5.35), (5.37) y (5.40). El eje de las abscisas corresponde al incremento en radianes del ángulo alfa. La elongación por unidad de longitud en dirección Y (E_y) exhibe un comportamiento siempre ascendente, sin alcanzar 1.25 para un rotación de 90°. La distorsión en el ángulo que forman dos líneas perpendiculares (ϕ_f) tiene un comportamiento menos uniforme, siendo este ángulo más agudo ($\approx 50^\circ$) cuando alfa se encuentra entre los 50 y 55 grados,

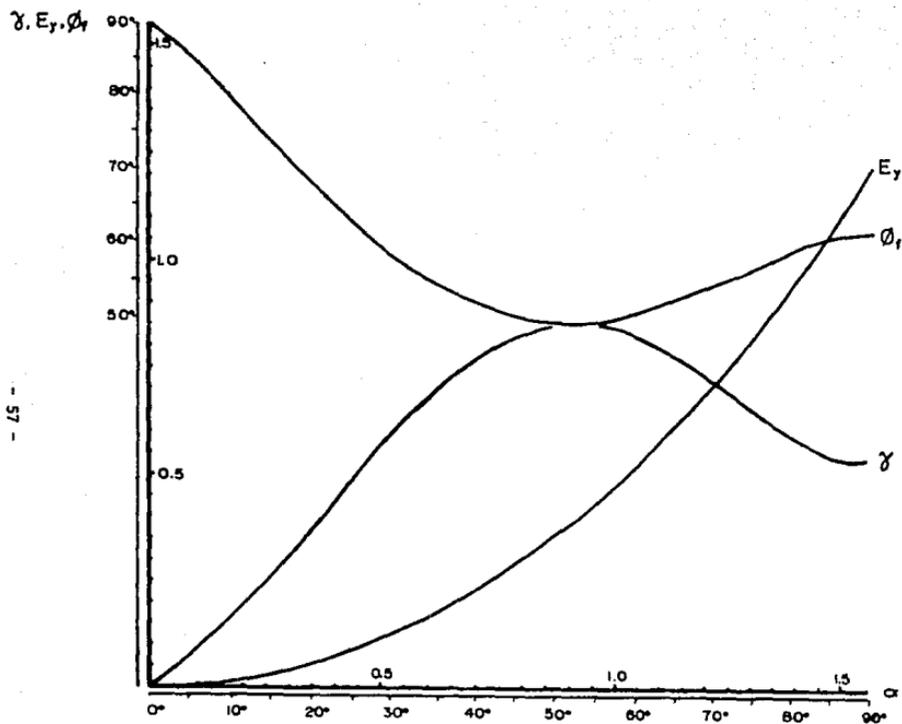


Figura 5.5 Comportamiento de la elongación por unidad de longitud a lo largo del eje Y (distorsión longitudinal E_y) y de la distorsión angular por cizallamiento ϕ_f y de la cantidad de cizalla γ .

mientras que la cantidad de cizallamiento (γ) es una imagen del comportamiento de ϕ_r .

Si se analizan de manera similar las ecuaciones (5.26), se llega a conclusiones similares. El transporte a través de planos axiales causa en general una distorsión de líneas y de ángulos, pero que el área permanece constante.

Se puede concluir de acuerdo a lo anterior, que el modelo produce una distorsión neta de elementos lineales y de ángulos pero no de área. Entonces el resultado es una sección con conservación de área, y con conservación de longitud en las direcciones que sean paralelas al segmento de falla subyacente y cuando el vector de desplazamiento no cambia de dominio de echado.

5.2 PROGRAMACION DEL ALGORITMO EN TURBO-PASCAL.

El algoritmo presentado anteriormente se programó en Turbo-Pascal, versión 3.07A de la compañía Borland International, aprovechando de las subrutinas comerciales de graficado Turbo Pascal Graphix Toolbox versión 1,07A.

Existen algunas ventajas del lenguaje Pascal y del compilador sobre otros como BASIC o FORTRAN, que se discuten a continuación:

i) Pascal es un lenguaje estructurado. Esto otorga mayor claridad y legibilidad en los programas e incrementa la eficiencia de los mismos. En la Figura 5.6 se muestra la morfología de una declaración estructurada, en donde la declaración general consta de un anidamiento de instrucciones sencillas o estructuradas y cada bloque de instrucciones está claramente delimitado. La Figura 5.7 muestra las partes de que consta un programa en Pascal.

ii) En Pascal no se requiere de formatos de entrada para los parámetros proporcionados por el usuario pero sí en los de salida, lo que facilita la utilización del programa.

iii) Pascal es un lenguaje formal que reduce la posibilidad de error al programar, debido a que el inicio del programa, funciones y subrutinas se definen las variables a utilizarse en los mismos; no es permitido usar variables, funciones y subrutinas no especificadas.

iv) Pascal cuenta con un número mayor de instrucciones de control de proceso, como son: *goto*, *if then else*, *case of*, *for do*, *while do* y *repeat until*.

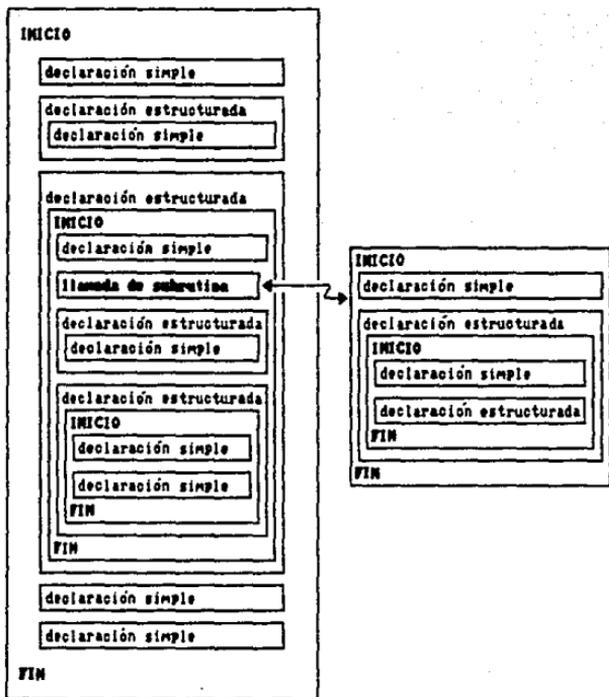


Figura 5.6 Relación entre las instrucciones de bloque en programación estructurada.

- | | |
|--|---|
| <p>1.- DECLARACION DE PROGRAMAS</p> <p>2.- CONSTANTES PREDEFINIDAS POR PROGRAMADOR</p> <p>3.- DECLARACION DE VARIABLES
 -ENTERAS
 -REALES
 -CARACTERES
 -CADENAS DE CARACTERES
 -BOLEANAS</p> <p>4.- SUBROUTINAS
 -DECLARACION DE CONSTANTES PREDEFINIDAS
 -DECLARACION DE VARIABLES LOCALES
 -SUBROUTINAS
 -FUNCIONES
 -INICIO
 INSTRUCCIONES
 -FIN</p> <p>5.- FUNCIONES
 -DECLARACION DE CONSTANTES PREDEFINIDAS
 -DECLARACION DE VARIABLES LOCALES
 -SUBROUTINAS
 -FUNCIONES
 -INICIO
 INSTRUCCIONES
 -FIN</p> | <p>6.- DECLARACION DE ETIQUETAS</p> <p>7.- INICIO DEL PROGRAMA
 -INSTRUCCIONES</p> <p>8.- FIN DEL PROGRAMA</p> |
|--|---|

Figura 5.7 Elementos de que consta un programa en Pascal.

v) El compilador Turbo-Pascal cuenta con su propio editor que despliega mensajes de errores y omisiones, con especificación de tipo, cometidos durante la edición. También permite realizar pruebas preliminares con los programas ejecutables almacenados en memoria principal, lo que aumenta la velocidad de acceso a los mismos, o bien, se pueden almacenar en memoria auxiliar (disco flexible).

vi) El compilador permite editar módulos independientes en archivos separados para posteriormente compilarse e integrarse en uno solo.

vii) Este compilador dispone de paquetes comerciales de graficación y análisis numérico.

El programa genera una red de nodos regularmente separados que representan un medio rocoso estratificado. Este es sujeto a afallamiento y deslizamiento considerando las propiedades geométricas del deslizamiento expuestas en el inciso anterior. La red ayuda a visualizar la distorsión que sufre el medio, ya que utilizar únicamente líneas para referenciar la estratificación no permite apreciar totalmente la distorsión interna del medio. Esto equivale a proponer un sistema Lagrangiano de referencia para apreciar las diferencias entre el estado inicial y la geometría final deformada.

La ejecución del algoritmo está controlada por un menú principal y varios menús secundarios que controlan la entrada y salida de datos y que se relacionan en forma arbórea con el menú principal (Figura 5.8). Sin embargo, las ecuaciones desarrolladas en la presentación del algoritmo se han reemplazado por rutinas numéricas más sencillas, dada la capacidad del ordenador de usar (i) comparaciones lógicas, (ii) recursividad en el manejo de expresiones y (iii) articulación de alto nivel en instrucciones, que no pueden ser manejadas dentro del lenguaje matemático. Esto permite optimizar y reducir el tiempo de proceso. El diagrama de flujo es el siguiente:

ESQUEMA SIMPLIFICADO DE LA SECUENCIA DE EJECUCION DEL PROGRAMA

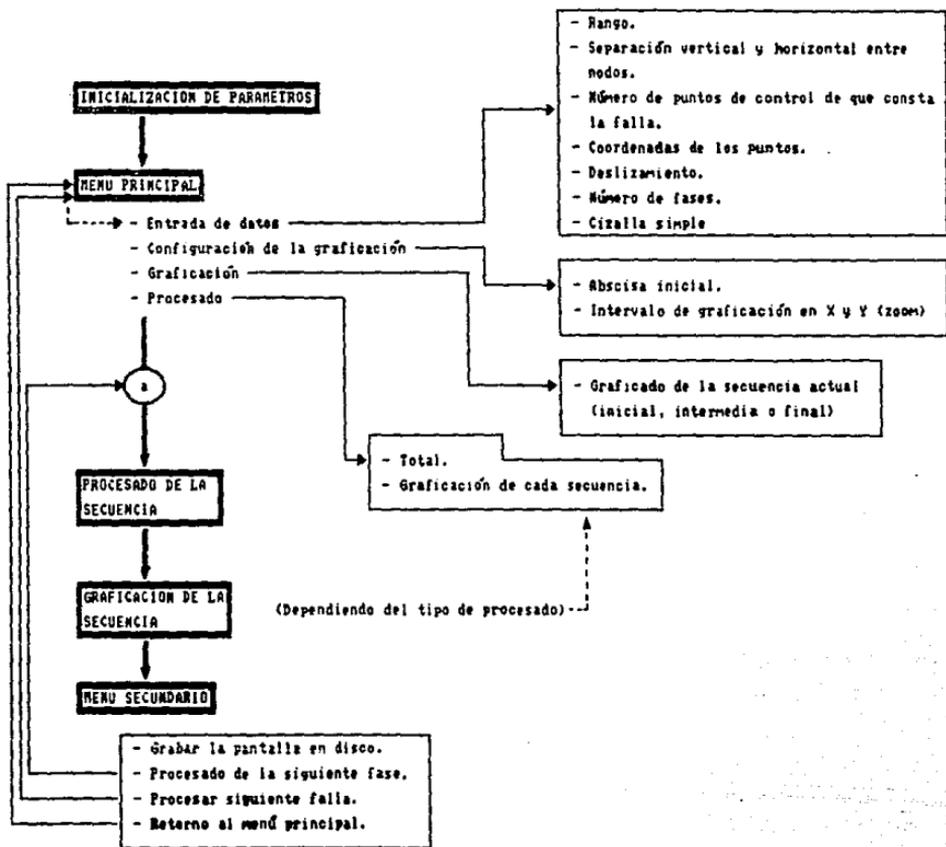


Figura 5.8

VARIABLES USADAS EN EL PROGRAMA DE FLEJO :

CUERPO PRINCIPAL :

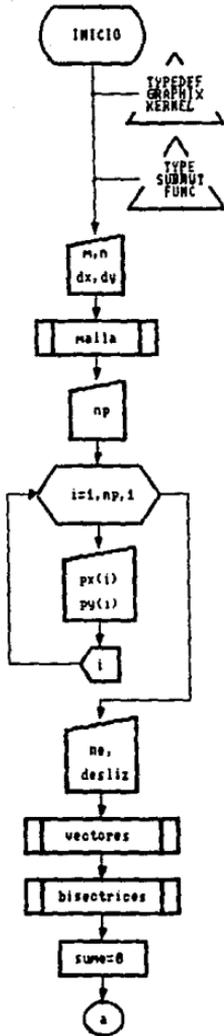
m,n : Rango de la malla de referencia.
dx,dy : Dimensiones de las celdas de la malla.
np : Número de puntos para configurar la falla activa.
px,py : Puntos de control para definir la falla activa.
ne : Número de etapas a simular.
desliz : Desplazamiento total sobre la falla activa.
suma : Sumatoria de etapas.
iso : Desplazamiento parcial que sufre el bloque alófono en cada etapa.
test : Variable booleana para verificar si un nodo es desplazable.
test2 : Variable booleana par verificar si un nodo se encuentra en la proximidad de un eje bisector.
x,y : Coordenadas de los nodos de la malla de referencia.
dif : Distancia entre un nodo a probar, y el eje axial.
bis : Pendiente de los planos axiales (bisectrices de los ángulos que forman los tramos rectos de la falla activa).
pos : Variable para designar el dominio al que pertenece un nodo.
vecx,vecy : Vectores unitarios paralelos a los segmentos que componen a la falla.

SUBROUTINAS :

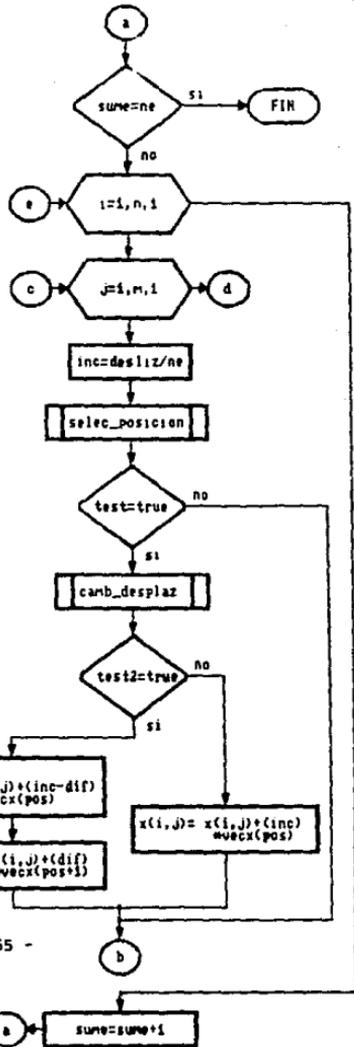
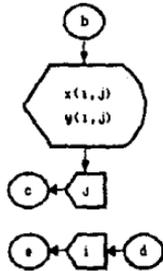
malla : Subrutina que genera de manera automática la red de referencia.
Datos de entrada : m,n,dx,dy.
Salida : x,y.
vectores : Subrutina para calcular los vectores unitarios paralelos a los tramos rectos de la falla activa.
Datos de entrada : np,px,py.
Salida : vecx,vecy.
bisectrices : Subrutina para calcular la pendiente de los ejes bisectores.
Datos de entrada : np,px,py.
Salida : bis.
nodes_posicion : Subrutina para verificar si un nodo es elemento de aquella parte del medio que se va a desplazar.
Datos de entrada : np,px,py,x,y.
Salida : test,pos.

comp_desplaz : Subrutina para comparar la distancia entre un nodo y el plano axial y la magnitud del vector de desplazamiento.
Datos de entrada : $x(i,j)$, $y(i,j)$, $p_x(pos+1)$, $p_y(pos+1)$,
 $his(pos)$, $vecx(pos)$, $vecy(pos)$, inc .
Salida : $test2$, dif .

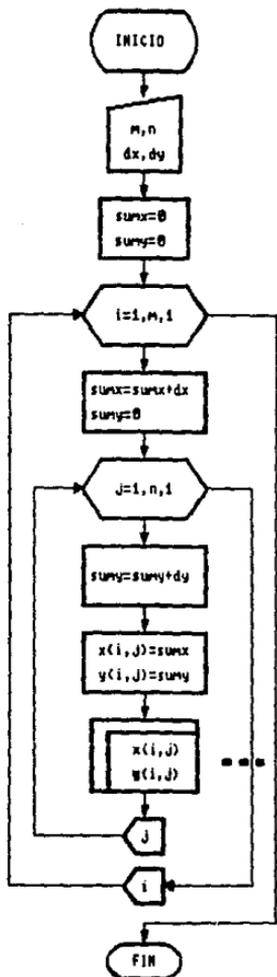
CUERPO PRINCIPAL



Archivos incluidos con subrutinas de graficación y subrutinas de análisis numérico.



SUBRUTINA MALLA

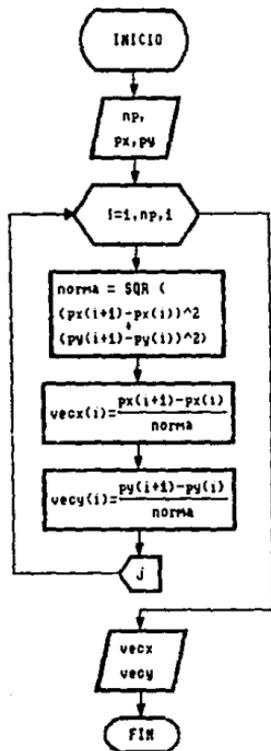


VARIABLES LOCALES

- sumx** : Sumatoria de la distancia horizontal que separa a nodos vecinos para formar una malla de $m \times dx$ de largo.
- sumy** : Sumatoria de la distancia vertical que separa a nodos vecinos para formar una malla de $m \times dy$ de ancho.

┌ - Colocación en
 | memoria de los
 | vértices
 | calculados
 └ -

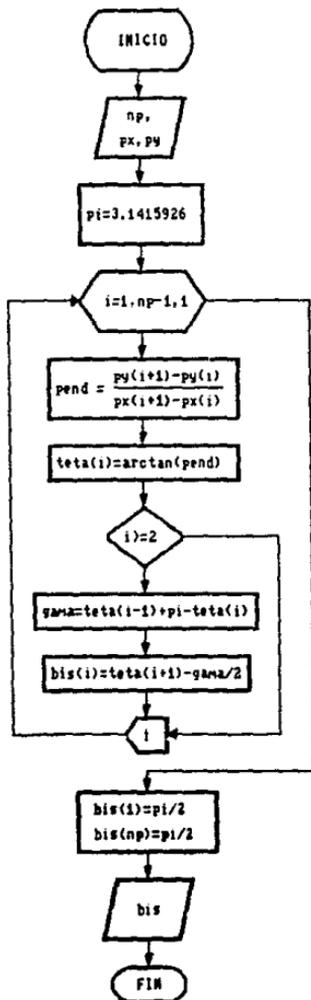
SUBROUTINA VECTORES



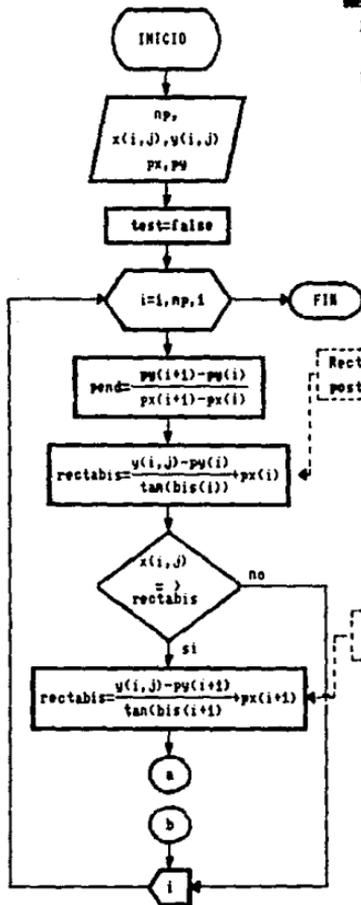
VARIABLES LOCALES

- NORMA** : Distancia entre dos puntos o magnitud del vector que une dos puntos.
- PEND** : Pendiente de la recta que pasa por dos puntos conocidos.
- TETA** : Echado de cada segmento de falla.
- GAMA** : Angulo formado por dos segmentos de falla.

SUBROUTINA BISSECTRICES



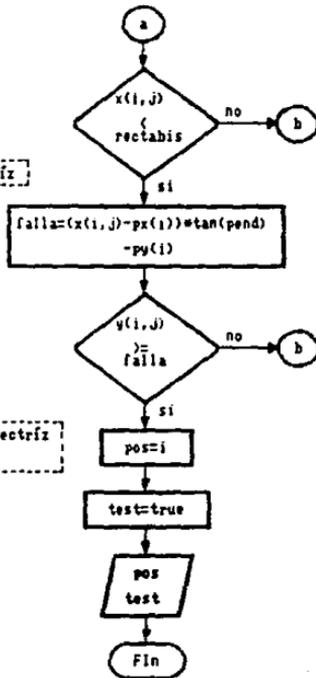
SELECCION DE POSICION



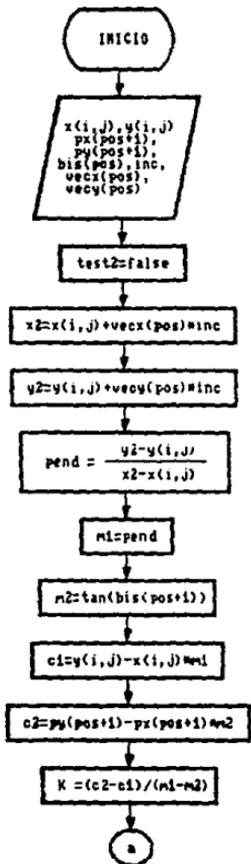
VARIABLES LOCALES

RECTABIS : Ecuación en función de "Y" de la recta bisectriz.

FALLA : Ecuación en términos de "X" de los segmentos de recta.

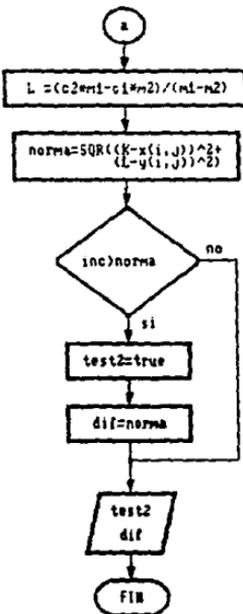


SUBROUTINA COMB_MESPLAS



VARIABLES LOCALES

- X2,Y2** : Posición desplazada de $x(i,j)$, $y(i,j)$.
- m1** : Pendiente de la recta que pasa por $(x(i,j),y(i,j))$ y $(x2,y2)$.
- m2** : Término constante de la ecuación de la recta anterior.
- m3** : Pendiente de la recta bisectriz.
- c2** : Término constante de la ecuación de la recta anterior.
- K,L** : Coordenadas del punto de intersección de las rectas anteriores.



Para correr el programa, se necesitan de los siguientes datos:

- Rango: número de renglones y columnas de los cuales consta la malla. El rango máximo es de 40 renglones por 200 columnas; si alguno de estos límites es excedido la ejecución del programa es abortada.
- Separación horizontal y vertical entre nodos adyacentes.
- Número de puntos de control que configuran la superficie de la falla, cuyo deslizamiento se va a ejecutar. El máximo número de puntos permitidos es 20.
- Coordenadas de los puntos: aquí, las abscisas deben de introducirse en orden progresivo y en el sentido del movimiento. Es decir, si el movimiento es positivo (de izquierda a derecha) entonces:

$$X_1 < X_2 < \dots < X_{n-1} < X_n$$

o por el contrario, si es negativo:

$$X_1 > X_2 > \dots > X_{n-1} > X_n$$

- Deslizamiento: magnitud del vector de desplazamiento.
- Número de etapas: fases a simularse en la deformación. Para un funcionamiento correcto del programa, el número de etapas debe de ser mayor o igual al resultado de la división entera entre deslizamiento y el segmento más corto de la configuración de la falla.
- Cizallamiento simple (opcional, no incorporado en el diagrama de flujo): éste se introduce para obtener un mejor ajuste entre los resultados sintéticos y los datos reales. Si se elige esta opción, se deben proporcionar los siguientes parámetros:
 - Número de puntos de control. Para simular un cizallamiento complejo, se da el número de puntos que aproxima la curva que se requiere para el mejor ajuste. Esta curva define el cizallamiento y suma al movimiento de cuerpo rígido.
 - Coordenadas de los puntos. Se especifican de acuerdo a coordenadas (X,Y). Después, un algoritmo numérico dentro del programa se encarga de descomponer el cizallamiento en las direcciones paralelas a los segmentos que componen a la falla.

En el capítulo 5.4 se muestran dos ejemplos, basados en datos reales con los parámetros de entrada usados.

El objetivo de introducir etapas de deformación es de visualizar estados de deformación intermedios, como las de la Figura 5.9. Incluso es posible obtener animaciones computarizadas de la secuencia cinemática, si las imágenes se almacenan en archivos y posteriormente se extraen de manera secuencial. La Figura 5.10 muestra la capacidad del programa de modelar estructuras de geometría más compleja que aquellas de los ejemplos del capítulo siguiente, sin que esté basado en un juego de datos reales.

5.3 IMPLEMENTACION DEL ALGORITMO EN UN SISTEMA IBM-PC.

El algoritmo se desarrolló para su uso con una computadora IBM-PC con capacidad de memoria de 640 Kilobytes (Kb), un coprocesador matemático 8087 y con un monitor CGA (IBM Color Graphics Adapter), este monitor cuenta con un campo de 640 pixeles en sentido horizontal por 200 pixeles en sentido vertical.

De los 640 Kb de memoria, el código consume 51 Kb. El compilador requiere 80 Kb y reserva 64 Kb para el código de los programas y 64 Kb de pila o "stack". De los restantes 430 Kb, las variables y gráficas del programa pueden ocupar hasta 230 Kb de memoria. El tiempo de procesamiento de los datos varía de acuerdo al tamaño de la malla, el número de segmentos y fallas, así como de su disposición. Por ejemplo, para procesar una red de 20 renglones por 100 columnas, un computador de 8 Mhz de frecuencia requiere de un minuto; mientras que sin coprocesador, el procesamiento de la misma configuración, tarda 15 minutos o más.

5.4 COMPARACION DE LOS RESULTADOS SINTETICOS CON DATOS REALES DE CINTURONES DE PLIEGUES Y CABALGADURAS.

Campo Cantarell. -

El Campo Cantarell se localiza en el Golfo de México dentro de la provincia petrolera de la Sonda de Campeche entre los 5° 95' y 6° 10' longitud E y los 21° 35' y 21° 45' latitud N, en el borde occidental de la Plataforma de Yucatán como lo muestra la Figura 5.11a (Evaluación de Formaciones de México, Schlumberger Offshore Services, 1984).

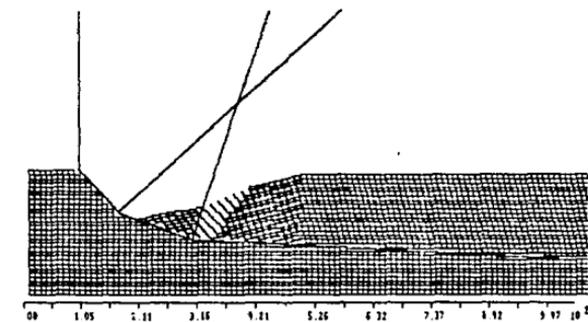
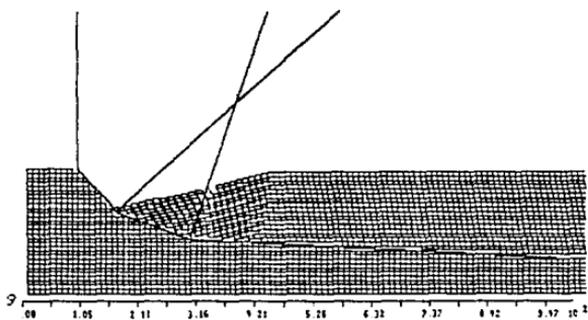
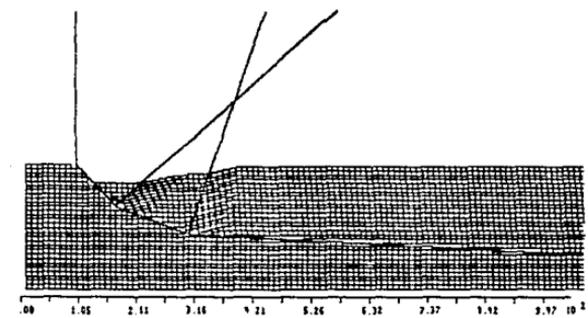


Figure 5.9 Simulación de una falla normal por medio del programa.



Figura 5.10 Modelado directo de una serie de lentes de cizallamiento. El modelo no corresponde a un ejemplo específico. (Retocado para su mejor visualización).

Las rocas almacenadoras de los campos marinos de la Sonda de Caspeche son las calizas dolomitizadas y dolomías del Kimeridgiano, del Cretácico y de la base del Paleoceno.

Según la sección estructural de la Figura 5.11b (Suter, 1984c) el Campo Cantarell está estructuralmente caracterizado por un sistema de cabalgaduras de orientación NW-SE con raíz común. El campo está cortado por dos fallas inversas de gran magnitud, atravesadas por el pozo Cantarell-2239, que generan una estructura antiformal de poco más de 2000 m de relieve estructural. Además el campo es intersectado por un patrón de fallas normales (líneas gruesas punteadas de la Figura 5.11a).

Para simular la deformación se digitizaron directamente los puntos de cambio de pendiente en la sección estructural de la Figura 5.11b, desplazando el punto donde se unen las cabalgaduras 15 cm hacia la izquierda. Los valores se introdujeron en un sistema de referencia con longitud unitaria arbitraria.

Los datos de entrada del procesamiento fueron los siguientes (notese que en Pascal no es necesario separarlos con comas):

- Rango de la malla (renglones x columnas): 10 100
- Espaciamiento horizontal y vertical entre puntos vecinos:
30 15 (unidades de longitud)

FALLA No 1 (inferior)

- Número de puntos que definen la falla: 3
- Px(1) = 0 ; Py(1) = 0
- Px(2) = 1500 ; Py(2) = 0
- Px(3) = 3500 ; Py(3) = 220

- Deslizamiento sobre la falla: 500
- Número de etapas: 1
- Cisalla simple ? N

FALLA No 2 (superior)

- Número de puntos que definen la falla: 3
- Px(1) = 0 ; Py(1) = 0
- Px(2) = 1500 ; Py(2) = 0
- Px(3) = 2260 ; Py(3) = 120

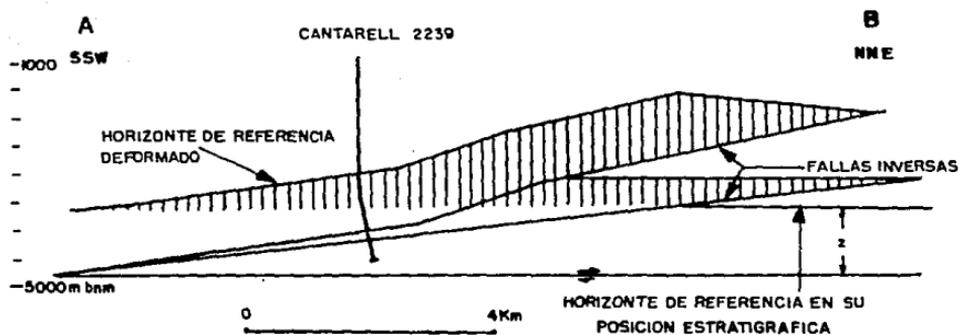
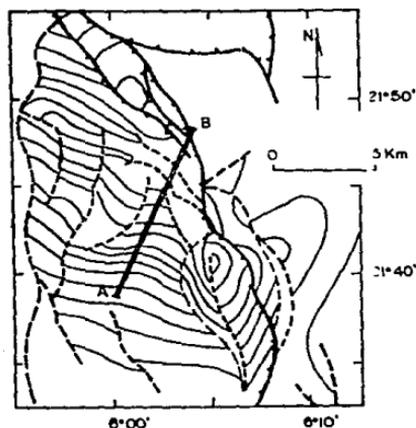


Figura 5. 11a Mapa de contornos estructurales de la cima del Paleoceno del Campo Cantarell. (Evaluación de Formaciones en México, 1984, Fig. 1-31).

Figura 5. 11b Sección estructural a través del Campo Cantarell. El área sombreada indica el levantamiento sobre el nivel estratigráfico de las Brechas del Paleoceno. (Según Suter, 1984c).

Px(4) = 2450 ; Py(4) = 200
Py(5) = 3200 ; Py(5) = 350

- Deslizamiento sobre la falla: 650
- Número de etapas: 4
- Cizalla simple ? N

La malla generada y deformada por el programa se muestra en la Figura 5.12 y una sección estructural basada en esta en la Figura 5.13.

Como se aprecia, existen discrepancias entre la sección estructural de la Figura 5.11b y la sección sintética. Así, la estructura modelada tiene una cresta más plana y es un poco más simétrica. Una forma de ajustar la geometría sintética, sería variar el deslizamiento sobre la falla inferior y la configuración de la falla superior. El ajuste en la proporción entre los frentes de las cabalgaduras controla el relieve estructural y el ángulo de corte en el flanco alto de la rampa superior, debiendo compensar ambos.

Cabalgadura Agua Fria.-

Esta cabalgadura se localiza en el Estado de Hidalgo en la porción oriental del cinturón orogénico Cordillerano, en el centro-este de México a los 21° de latitud norte (Figura 5.14, Suter, 1987).

La columna estratigráfica de esta área comprende tres formaciones. La más antigua, la Formación Las Trancas (Kimeridgiano-Barremiano en las partes expuestas), está constituida por calizas pelágicas y filitas con areniscas, grauvacas y conglomerados subordinados. Le sobryace el miembro de interior de plataforma de la Formación El Abra, consistente de calizas bien estratificadas de edad Barremiano-Coniaciano. Encima, se encuentra la Formación Soyatal, constituida por intercalaciones de lutita y lutita calcárea de edad Campaniano-Maastrichtiano.

En la Figura 5.14, también se muestra una sección estructural simplificada a través de la plataforma carbonatada de Valles-San Luis Potosí, que corta a la Cabalgadura Agua Fria. La estructura del subsuelo se ha construido bajo dos suposiciones: 1) que la

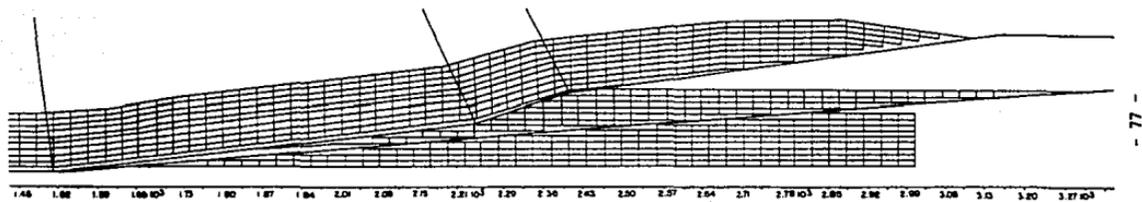


Figura 5.12 Sección sintética del Campo Cantarell generada por el algoritmo.

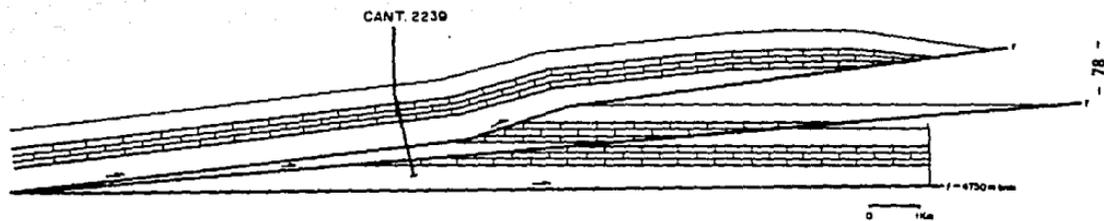


Figura 5.13 Sección estructural basada en la figura 5.10. La ornamentación de caliza es únicamente como referencia.

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

rotación esté restringida al bloque superior, y ii) que el plegamiento esté relacionado con la acumulación de material mecánicamente incompetente de la Formación Las Trancas en los núcleos arriba de la rampa tectónica (Suter, 1987).

Sin embargo, si la rotación se extiende hasta el despegue basal, la sección tendría una apariencia diferente, se plegarían la cabalgadura y el bloque inferior. Otra posibilidad es que la rampa tenga una geometría escalonada, con segmentos horizontales bajo los núcleos de los dos anticlinales y rampas en sus flancos occidentales (Suter, 1987).

Para modelar la sección con el algoritmo se acepto esta última interpretación de la geometría del subsuelo, buscando el óptimo ajuste por ensayo y error. Los datos que ofrecen el mejor ajuste son los siguientes (para la Cabalgadura Lobo-Cienega los datos se midieron directamente del perfil y se introdujeron de manera similar a los del Campo Cantarell):

- Rango de la malla (rengiones x columnas): 10 100
- Espaciamiento horizontal y vertical entre puntos vecinos:
30 20

FALLA No 1 (Cabalgadura Lobo-Cienega)

- Número de puntos que definen a la falla: 3
- $P_x(1) = 0$; $P_y(1) = 0$
- $P_x(2) = 2250$; $P_y(2) = 0$
- $P_x(3) = 3200$; $P_y(3) = 100$

- Deslizamiento sobre la falla: 1000
- Número de etapas: 2
- Cizalla simple ? N

FALLA No 2 (Cabalgadura Agua Fría)

- Número de puntos que definen a la falla: 7
- $P_x(1) = 0$; $P_y(1) = 0$
- $P_x(2) = 1950$; $P_y(2) = 0$
- $P_x(3) = 2070$; $P_y(3) = 60$
- $P_x(4) = 2350$; $P_y(4) = 80$
- $P_x(5) = 2450$; $P_y(5) = 150$
- $P_x(6) = 2650$; $P_y(6) = 180$
- $P_x(7) = 3200$; $P_y(7) = 450$

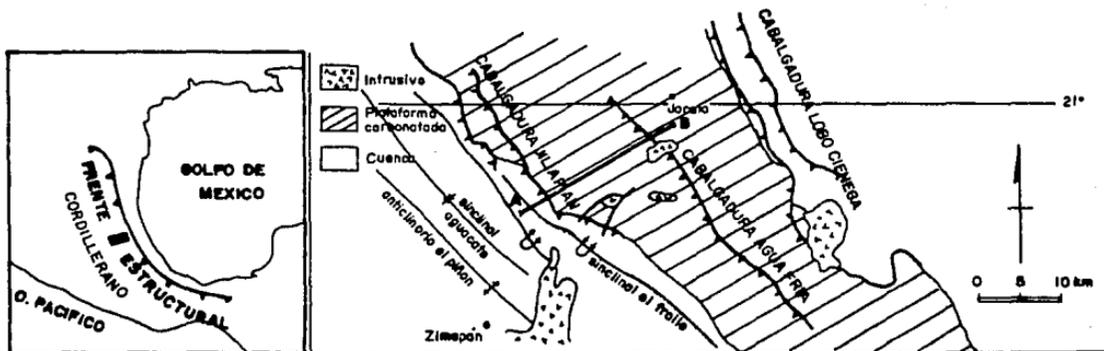
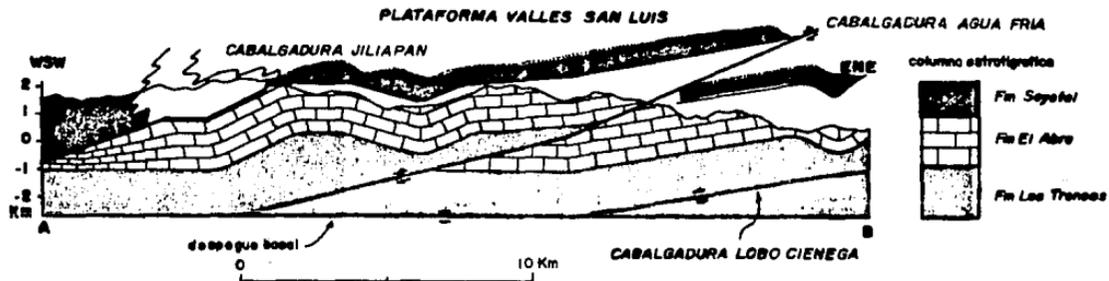


Figura 5.14 Localización y geología simplificada de la cabalgadura Agua Fria.

Abaajo, sección estructural AB donde se muestra la geometría de la rampa. (Según Suter, 1987, Fig. 1 y 5).



- Deslizamiento sobre la falla: 240
- Número de etapas: 3
- Cizalla simple ? S
- Número de puntos de control de deslizamiento: 2
Px(1) = 0 ; Py(1) = 0
Px(2) = 40 ; Py(2) = 200

En la Figura 5.15 se muestra la malla generada y deformada por el programa y en la Figura 5.16, una sección estructural basada en esta. Como se observa, se ha obtenido una similitud aceptable con los datos proporcionados, tomando en cuenta a una geometría de rampa escalonada, sin que sea la única solución posible.

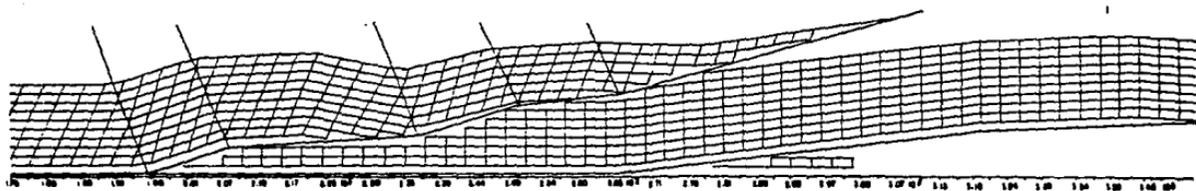


Figura 5.15 Sección sintética de la Cabalgadura Agua Fria generada por el algoritmo.

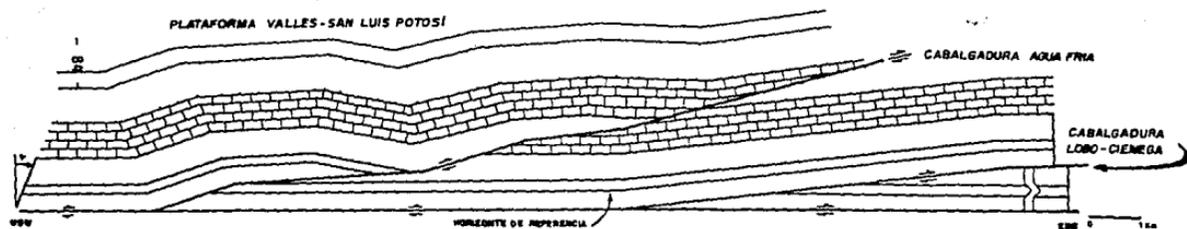


Figura 5.16 Sección estructural basada en la figura 5.13.

CONCLUSIONES

Esta tesis investigó el comportamiento cinemático de secuencias de deformación geológica en cortes transversales por medio de una simulación por computadora. Se programaron con este objetivo algoritmos en el lenguaje Pascal que se implementaron en una IBM-PC.

1.- Se definieron analíticamente por medio de una formulación lagrangiana funciones para el desplazamiento a lo largo de fallas normales y cabalgaduras en un medio litoestratigráfico. Estas traslaciones se definieron a partir de la configuración inicial no deformada (modelado directo).

2.- El medio litoestratigráfico se representó por medio de elementos finitos cuadrilaterales convirtiendo así el problema de un espacio analítico a un espacio discreto, lo que permite almacenar y manejar la presentación en forma de arreglos por computadora.

3.- Después, las transformaciones analíticas se evaluaron numéricamente para los nodos de la malla de elementos finitos y la malla deformada se graficó por computadora.

4.- Finalmente, el algoritmo se comprobó comparando los resultados sintéticos para varias secciones estructurales con datos reales y buscando el mejor ajuste por ensayo y error.

AGRADECIMIENTOS.

Quisiera agradecer principalmente al Dr. Max Suter, por sus aportaciones y el tiempo dedicado a dirigir este trabajo.

También agradezco al Instituto de Geología y a la División de Ciencias de la Tierra de la Facultad de Ingeniería por las facilidades brindadas durante el desarrollo de esta tesis, en especial al Dr. Ricardo J. Padilla y Sanchez.

A los sinodales: Ing. Jorge Nieto Obregón, Ing. Gilberto Silva Romo e Ing. Miguel González Cárdenas que revisaron el texto preliminar.

LISTA DE REFERENCIAS

- Barr, A.H., 1984, Global and local deformations of solid primitives: Computer Graphics, v. 18, p. 21-30.
- Chamberlain, R.T., 1910, The Appalachian folds of Central Pennsylvania: J. Geol., v.18, p. 228-251.
- Cobbold, P.R., Means, V.D. y Bayly, M.B., 1984, Jusep deformations gradients gradients and particle velocities across propagating coherent boundaries: Tectonophisics, v. 108, p. 283-298.
- Crane, R.C., 1987, Use of cut-offs and bed travel distance in balanced cross sections: J. Struct. Geol., v. 9, p. 243-246.
- Dahistrom, C.D.A., 1969, Balanced cross-sections: Canadian Journal of Earth Sciences, v. 6, p. 57-71.
- Davies, J.C., 1973, Statistics and data Analisis in Geology: Jhon Wiley and Sons, New York, 550 p.
- De Paor, D.G., 1988, Balanced sections in thrust belts part 1: constructions: American Association of Petroleum Geologist Bulletin, v. 72, p 73-90.
- Evans, D.G, Schweitzer, P.N., y Hanna, M.S, 1985, Parametric cubic splines and geology shapes descriptions: J. int. Ass. Nhat. Geol. v. 17, p. 611-624.
- Fail, R.T., 1973, Kink band folding, Valley and Ridge Province, Pennsylvania: Geological Society of America Bulletin, v. 84, p. 1289-1314.
- Gibbs, A.D., 1983, Balanced cross sections construction from seismic sections in areas of extensional tectonics: Journal of Structural Geology, v. 5, p. 153-160.
- Hossack, J.R., 1979, The use of the balanced cross-sections in the calculation of orogenic contraction: a review: Journal of the Geological Society of London, v. 136, p. 705-711.

- Kass, H. y Witkin, A., Analysing oriented patterns, paper presented at International Joint Conference for Artificial Intelligence, Univ. of Calif. at Los Angeles, Los Angeles Calif., August 1985.
- Kligfield, R., Geiser, P. y Geiser, J., 1986, Construction of geological cross sections using microcomputer: Geobyte, spring, p. 60-66.
- Laubecher, H.P., 1962, Die Zweiphasenhypothese der Jurafaltung: Eclog. Geol. Helv., v. 55, p. 1-22.
- , 1976, Geometrical adjustments during rotation of a Jura fold limb: Tectonophysics, v. 36, p.347-365.
- McCoss, A.M., 1987, Practical section drawing through folded layers using sequentially rotated cubic interpolators: J. Struct. Geol., v. 9 p. 363-370.
- Means, W.D., 1976, Stress and strain, basic concepts of continuum mechanics for geologist: Springer - Verlag, 339 p.
- Paterson, M.S., 1978, Experimental rock deformation—the brittle field: Springer, New York, 234 p.
- Press, W.H., Flannery, B.P., Teukolsky, S.A., and V.T. Vetterling, 1986, Numerical recipes: Cambridge University Press, New Rochelle, New York, 818 p.
- Ragan, D.M., 1984, Structural geology: An introduction to geometrical techniques: Jhon Wiley and Sons, New York, 393 p.
- Ramsay, J.G., 1967, Folding and fracturing of rocks: New York. McGraw-Hill, 568 p.
- , y Ruber, M.I., 1987, The techniques of modern structural geology, vol 2: folds and fractures: New York, Academic Press, 700p.
- Schlumberger Offshore Services, 1984, Evaluación de formaciones en México, Galile, Paris Francia, p. I-35.
- Suppe, J. y Nesson, J., 1979, Fault-bend origin of frontal folds

of the western Taiwan fold-and-thrust belt: *Petroleum Geology of Taiwan*, No 16, p. 1-18.

- , 1983, Geometry and kinematics of fault-bent folding: *American Journal of Science*, v. 283 p. 684-721.
- , 1985, Principles of structural geology: Englewood Cliffs, New Jersey, Prentice-Hall, 537p.
- Suter, M., 1981, Strukturelles Querprofil durch den nordwestlichen Faltenjura, Mt-Terri-Ranüber Schiebung-Freiberg: *Eclogae geol. Helvetiae*, v. 74, p. 278 y 261.
- , 1984a, Secciones estructurales con balanceo de masa: Resumen de la VII Convención Nacional de la Sociedad Geológica Mexicana, p. 125.
- , 1984b, Cordilleran deformation along the eastern edge of the Valles-San Luis Potosí carbonate platform, Sierra Madre Oriental fold thrust-belt, east-central México: *Geological Society of America Bulletin*, v. 95 p. 1384-1397.
- , 1984c, Interpretación geológica de los datos adquiridos en el pozo Cantarell-2239: Schulumberger Offshore Services, México D.F. (informe inédito).
- , 1987, Structural traverse across the Sierra Madre Oriental fold-thrust belt in east-central México: *Geological Society of America Bull.*, v. 98 p. 249-264.
- Wood, D.S., 1974, Current views of the development of slaty cleavage: *Ann. Rev. Earth Sci.*, v.2, p.369-401.

APENDICE: Listado en turbo Pascal del programa fuente.

A continuación se presenta el listado del programa fuente del que se obtuvieron las secciones sintéticas presentadas en este trabajo. Para este trabajo se utilizó la versión 3.07A del compilador de Turbo Pascal de la compañía Borland. Se recomienda tener precaución si se utiliza alguna versión diferente. Los archivos incluidos deben editarse cada uno por separado con el objetivo de evitar al mínimo pérdidas de información por errores y como una medida para ahorrar código del archivo principal (ya que está restringida a 64 Kb); encargándose el compilador de unirlos al momento de la ejecución. El archivo TIT.PAS se debe de editar y de correr para almacenar en disco la imagen del menú que aparece durante el despliegue de las gráficas. Los archivos TYPEDEF.SYS, GRAPHIX.SYS, KERNEL.SYS y WINDOW.SYS están incluidos en el paquete de graficación TURBO GRAPHICS TOOLBOX de la Borland International versión 1.07A, si se cuenta con una versión diferente primero revícase la sintaxis de los parámetros y forma de operar de cada subrutina para verificar la compatibilidad de las versiones. Las subrutinas de graficado que se encuentran en su mayoría en el archivo SUBGRAF.INC, como DRAWLINE, DRAWPOINT, DEFINEWINDOW, DEFINEWORLD, SELECTWINDOW, STOREWINDOW, etc son propias de los archivos antes mencionados, y se puede obtener su forma de operación en el manual "TURBO GRAPHICS TOOLBOX" editado por la Borland. Un error común, es no tener propiamente instalado el paquete de turbo graphics, por lo que hay que asegurarse de tener la configuración adecuada para el monitor ha usarse (monitor IBM CGA, Hércules, Zenith, etc).

(PROGRAMA PRINCIPAL)

program Falla;

```
(
=====
!!
!! El siguiente programa modela secuencias de deformacion apartir de la !!
!! configuracion de una serie de fallas, desplazamiento y su direccion !!
!! sobre ellas. Genera una malla regularmente espaciada y la deforma por !!
!! regiones limitadas por la bisectrices asociadas a las fallas !!
!!
=====
)

( Archivos incluidos que contienen )
( Paquete de subrutinas de graficacion Turbo Graphix )
( Toolbox. )
( " " )
( " " )
( " " (modificado). )
( Constantes predefinidas y variables del programa. )
( Mensajes del menu y mensajes de error. )
( Funciones (tangente,pendiente,etc). )
( Subrutinas numericas. )
( Subrutinas de graficaco (utilizan las de Turbo )
( Graphix). )
( Archivo incluido con las subrutinas principales. )

(==I TYPEDEF.SYS )
(==I GRAPHIX.SYS )
(==I KERNEL.SYS )
(==I WINDOWS.SYS )
(==I AXIS-MOD.HGH)
(==I VARTYPE.INC )
(==I MENU-ERR.MSG)
(==I FUNC.INC )
(==I SUBNUME.INC )
(==I SUBGRAF.INC )
(==I SUBPRINC.INC)
```

(===== CUERPO PRINCIPAL DEL PROGRAMA =====)

begin

Inicializaci0n;

IO:opciones(1);

if not datos then EntradaDatos;

datos:=true;

vectores_unit(np,ptox,ptoy,rx,ry); (Llamado a las subrutinas)

fallas(numfil,np,ptox,ptoy,numptf,hor,falla); (para generar los param=)

bisectrices(np,ptox,ptoy,penbis); (tros que controlan el)

repeat (programa)

if procesado then

begin

suminci:=suminci+1;

menu(4,0);

tiempo(hi,mi,si);

```

($C-,U-)
ProcesadoDatos;
($C+,U+)

tiempo(h2,m2,s2);
TiempoProc;

endi;

if graficado or (suminc = ne) then
begin
  ( Graficado de la secuencia deformada )
  entergraphic;
  graficado_malla;
  windpg:=1;indi:=0;
  Opciones(4);
  leavegraphic;
  if ch = 'R' then goto 10;
endi;

  ( Repite las instrucciones hasta que la suma de los incrementos sea igual )
  ( al numero de etapas intermedias )

until suminc = ne;

goto 10;

leavegraphic;

end.

```

{ARCHIVO AXIS-MOD.HGH}

```
(*****  
(# TURBO PASCAL GRAPHIX TOOLBOX VERSION 1.07A #)  
(# Copyright (C) 1985, B&B by #)  
(# BORLAND International #)  
(# MODIFICADO PARA ESTE TRABAJO #)  
(*****  
  
procedure DrawAxis(XDens, YDens, XLeft, YTop, XRight, YBottom,  
 XAxis, YAxis : integer; Arrows : boolean);  
var  
 LineStyleLoc, Xk0, Yk0, Xk1, Yk1, Xk2, Yk2, NDiff, X2, Y2,  
  MaxExponentX, MaxExponentY, I, Ys, Xs, Delta, NPoints : integer;  
  Difference, Number, S, Fract : real;  
  X1RefLoc, X2RefLoc, Y1RefLoc, Y2RefLoc,  
  X1RefLoc2, X2RefLoc2, Y1RefLoc2, Y2RefLoc2 : integer;  
  ClippingLoc, DirectModeLoc, HeaderLoc : boolean;  
  
function StringNumber(X1 : real; MaxExponent : integer) : WrkString;  
var  
  Y : WrkString;  
begin  
  Str(X1 : Exp(-MaxExponent * Ln(10.0));5:2, Y);  
  StringNumber := Y;  
end; { StringNumber }  
  
function GetExponent(X1 : real) : integer;  
begin  
  GetExponent := 0;  
  if X1 <> 0.0 then  
    if abs(X1) >= 1.0 then  
      GetExponent := trunc(Ln(abs(X1)) / Ln(10.0))  
    else  
      GetExponent := -trunc(abs(Ln(abs(X1))) / Ln(10.0) + 1.0);  
  end; { GetExponent }  
  
procedure DrawNum(X1, Y1, MaxExponent : integer; Number : real);  
var  
  I : integer;  
  StrNumber : WrkString;  
begin  
  StrNumber := StringNumber(Number, MaxExponent);  
  Y1 := Y1 - 3;  
  for I := 1 to 5 do  
    DrawAscii(X1, Y1, I, Ord(StrNumber[I]));  
  end; { DrawNum }  
  
function Balance : integer;  
begin  
  Balance := 0;  
  S := S + Fract;  
  if S >= 0 then  
    begin  
      S := S - 1.0;  
      Balance := 1;  
    end;  
end;
```

```

end; ( Balance )

procedure DrawExponent(X1, Y1, MaxExponent : integer);
var
  I      : integer;
  StrNumber : WrkString;
begin
  Y1 := Y1 - 3;
  X1 := X1 + 1;
  DrawAscii(X1, Y1, 1, 49);
  DrawAscii(X1, Y1, 1, 48);
  Str(MaxExponent:3, StrNumber);
  Y1 := Y1 - 3;
  X1 := X1 - 7;
  for I := 1 to 3 do
    DrawAscii(X1, Y1, 1, Ord(StrNumber[I]));
  end; ( DrawExponent )

begin ( DrawAxis )
  LineStyleLoc := LineStyleGlb;
  SetLineStyle(0);
  DirectModeLoc := DirectModeGlb;
  DirectModeGlb := true;
  with Window[WindowNdxGlb] do
  begin
    X1RefLoc := X1;
    X2RefLoc := X2;
    Y1RefLoc := Y1;
    Y2RefLoc := Y2;
    RedefineWindow(WindowNdxGlb, X1 + XLeft, Y1 + YTop,
                   X2 - XRight, Y2 - YBottom);
    SelectWindow(WindowNdxGlb);
  end;
  if (XDens < 0) xor (YDens < 0) then
  begin
    HeaderLoc := HeaderGlb;
    HeaderGlb := false;
    DrawBorder;
    HeaderGlb := HeaderLoc;
  end;
  XDens := abs(XDens);
  YDens := abs(YDens);
  if XDens > 9 then
    XDens := 0;
  if YDens > 9 then
    YDens := 0;

  Xk0 := (X1RefGlb) shl 3;

  Yk0 := Y2RefGlb - 14;
  Yk1 := Y1RefGlb + 6;
  Xk1 := Xk0;
  Yk2 := Yk0;
  Xk2 := (X2RefGlb - 2) shl 3 + 7;
  if (XAxis >= 0) or (YAxis >= 0) then

```

```

begin
  ClippingLoc := ClippingGlb;
  ClippingGlb := true;
  with Window[WindowNdxGlb] do
  begin
    X1RefLoc2 := X1;
    X2RefLoc2 := X2;
    Y1RefLoc2 := Y1;
    Y2RefLoc2 := Y2;
  end;
  ReDefineWindow(WindowNdxGlb, X1RefLoc2, Y1RefLoc2 + 6,
    X2RefLoc2, Y2RefLoc2 - 14);
  SelectWindow(WindowNdxGlb);
  DirectModeGlb := false;
  if (XAxis >= 0) then
  begin
    SetLineStyle(XAxis);
    DrawLine(X1WldGlb, Y1WldGlb + Y2WldGlb, X2WldGlb, Y1WldGlb + Y2WldGlb);
    SetLineStyle(0);
  end;
  if (YAxis >= 0) then
  begin
    SetLineStyle(YAxis);
    DrawLine(0, Y1WldGlb, 0, Y2WldGlb);
    SetLineStyle(0);
  end;
  ClippingGlb := ClippingLoc;
  DirectModeGlb := true;
  ReDefineWindow(WindowNdxGlb, X1RefLoc2, Y1RefLoc2, X2RefLoc2, Y2RefLoc2);
  SelectWindow(WindowNdxGlb);
end;
if XDens >= 2 then
begin
  DrawLine(Xk0, Yk0, Xk2 + 1, Yk2);
  if Arrows then
  begin
    DrawLine(Xk2, Yk2, Xk2 - 4, Yk2 - 4);
    DrawLine(Xk2, Yk2, Xk2 - 4, Yk2 + 4);
  end;
end;
if (abs(Yk0 - Yk1) >= 35) and (abs(Xk2 - Xk1) >= 150) then
begin
  DrawLine(Xk0, Yk0, Xk0, Yk0);
  DrawLine(Xk0, Yk0, Xk0, Yk0 + 4);
  Delta := Y2RefGlb - Y1RefGlb - 20;
  NPoints := Delta div 7;
  NDiff := Delta - (NPoints shl 3) + NPoints;
  if YDens >= 4 then
  begin
    if abs(Y2WldGlb) > abs(Y1WldGlb) then
      MaxExponentY := GetExponent(Y2WldGlb)
    else
      MaxExponentY := GetExponent(Y1WldGlb);
    if MaxExponentY <> 0 then

```

```

ends;
Fract := NDiff / NPoints;
S := -Fract;
Ys := Yk0;
Difference := (Y2WldG1b - Y1WldG1b) / NPoints;

if XDens >= 4 then
begin
  if abs(X2WldG1b) > abs(X1WldG1b) then
    MaxExponentX := GetExponent(X2WldG1b)
  else
    MaxExponentX := GetExponent(X1WldG1b);
  DrawNum(Xk0 - 14, Yk0 + 10, MaxExponentX, X1WldG1b);
  if MaxExponentX <> 0 then
    DrawExponent(Xk2 - 5, Yk0 + 10, MaxExponentX);
  end;
  Delta := abs(X2RefG1b - X1RefG1b) shl 3 - 41;
  NPoints := Delta div 30;
  NDiff := Delta - (NPoints shl 5) + (NPoints shl 1);
  Fract := NDiff / NPoints;
  S := -Fract;
  Xs := Xk0 - 1;
  Difference := (X2WldG1b - X1WldG1b) / NPoints;
  for I := 1 to NPoints do
  begin
    Xs := Xs + 32 + Balance;
    if (XDens > 2) and (Xs < X2RefG1b shl 3 + 7 - 24) then
    begin
      Number := X1WldG1b + I * Difference;
      DrawLine(Xs, Yk0, Xs, Yk0 + 4);
      if XDens >= 4 then
        if I mod (10 - XDens) = 0 then
          DrawNum(Xs - 14, Yk0 + 10, MaxExponentX, Number);
        end;
      end;
    end;
  end;
  RedefineWindow(WindowNdxG1b, X1RefLoc, Y1RefLoc, X2RefLoc, Y2RefLoc);
  SelectWindow(WindowNdxG1b);
  DirectModeG1b := DirectModeLoc;
  SetLineStyle(LineStyleLoc);
  AxisG1b := true;
  X1G1b := XLeft;
  X2G1b := XRight;
  Y1G1b := YTop;
  Y2G1b := YBottom;
end; ( DrawAxis )

procedure ResetAxis;
begin
  AxisG1b := true;
end; ( ResetAxis )

```

```
( ARCHIVO VARTYPE;  
  VARIABLES Y CONSTANTES PREDEFINIDAS )
```

```
const
```

```
pi=3.1415926;  
  
MaxNumColumn      = 100;  
MaxNumColumnExp   = 200;  
MaxNumReng        = 40;  
MaxNumF11         = 15;  
MaxNumPts         = 20;
```

```
type
```

```
MtrzPoint = ^matriz;  
matriz = record  
    x,  
    y: array[1..MaxNumReng,1..MaxNumColumn] of real;  
end;  
  
MtrzExpPoint = ^matrizExpan;  
matrizExpan = record  
    x,  
    y: array[1..MaxNumReng,101..MaxNumColumnExp] of real;  
end;  
  
Mtrz2Point = ^matriz2;  
matriz2 = record  
    x,  
    y: array[1..MaxNumF11,1..MaxNumColumnExp] of real;  
end;  
  
vector = array[1..MaxNumPts] of real;  
vector2 = array[1..MaxNumReng] of real;  
vector3 = array[1..MaxNumF11] of integer;
```

```
var
```

```
( Lista de variables usadas en el programa )
```

```
( Variables generales del programa: )
```

```
PtoX,PtoY      , ( Coordenadas de los puntos de control que constituyen las )  
                ( fallas. )  
Rfx,Rfy        , ( Vectores unitarios de desplazamiento asociados a los )  
                ( segmentos rectos de las fallas. )  
CizX,CizY      , ( Puntos de control de cizalla. )  
PenBis;vector  , ( Pendientes de las bisectrices de las inflexiones de )  
                ( las fallas. )
```

```

IncZiv:vector2; ( Incremento en el desplazamiento por cizalla. )
NumPti:vector3; ( Vector que almacena el numero de puntos interpolados pa- )
                ( ra cada sistema de fallas. )
Falla:Mtrz2Point; ( Sistema de puntos que constituyen a las fallas. )
Red:MtrzPoint; ( Sistema de puntos de referencia que constituyen una red. )
Red2:MtrzExpPoint; ( Sistema ampliado de puntos de referencia. )

```

```

i, j , ( Variables de ciclos anidados, se conserva esta nomenclatura a lo largo de las subrutinas. )
SumInc , ( Contador del numero de etapas. )
NumFl1 , ( Contador de numero de fallas. )
Np , ( Numero de puntos de que consta el sistema de fallas. )
Ne , ( Numero de etapas intermedias de deformacion. )
Nc , ( Numero de puntos de control de que consta la cizalla. )
m,n , ( Rango de la matriz de puntos de las capas. )
Reg , ( Variable que designa la region a la que pertenece cada punto de referencia. )
Ind , ( Variable que designa si se despliega o no el menu de opciones. )
NumWin , ( Numero de ventanas horizontales totales. )
Windpg , ( Ventana desplegada actualmente. )
H1,H2 , ( Variables para calcular el tiempo de procesado. )
M1,M2 , ( "" )
S1,S2:integer; ( "" )
Test,Test2: boolean; ( Variables booleanas de verificacion del movimiento. )
Hor,Ver , ( Separacion horizontal y vertical entre puntos adyacentes. )
Desliz , ( Deslizamiento sobre el sistema de fallas. )
DesTot , ( Desplazamiento total. )
Inc , ( Incremento en el desplazamiento. )
Dif:real; ( Distancia entre un nodo y el eje axial. )

```

(Variables de control de ejecucion:)

```

Ch : char; ( Caracter de control. )
Dato , ( Verificacion de la entrada de datos. )
Procesado , ( Verificacion de la ejecucion de procesado. )
Graficado : boolean; ( Verificacion de graficacion. )

```

(Variables de configuracion del graficado:)

```

Yinf , ( Limite inferior del intervalo de graficacion. )
Ysup , ( Limite superior del intervalo de graficacion. )
Xinf , ( Valor de inicio de graficado de las abscisas. )
LargoPantalla , ( largo de la pantalla. )
MaxElongX , ( Elongacion horizontal permitida para graficar. )
MaxElongY , ( Elongacion vertical permitida para graficar. )
Traslape : real; ( Traslape entre ventanas adyacentes. )

```

```

label
101 ( Etiqueta. )

```

(ARCHIVO MENU-ERR.MSB
 MENSAJES DE ERROR Y LETREROS.)

```

procedure VentanaMenu;
begin
  entergraphic;
  selectscreen(2);
  definewindow(15,0,0,79,8);
  selectwindow(15);
  loadwindow(1,-1,-1,'menu-sup');      ( Ventana predefinida, ver archivo )
  storewindow(15);                      ( TIT.PAS )
  selectscreen(1);
  leavegraphic;
end;

procedure menu(ind1,ind2:integer);
var numero:string[5];
label 10,20;
begin
  case ind1 of
    1 : begin
      gotoxy(1,3);
      writeLn('                MENU DE OPCIONES                ');
      writeLn('          ');
      if suminc <> 0 then
        begin
          writeLn('          [ F ]          | Siguiente falla.          ');
        end
      else
        begin
          writeLn('          [ E ]          | Entrada de Datos.          ');
          writeLn('          [ B ]          | Graficado de la secuencia.          ');
          writeLn('          [ C ]          | Configuracion de la Graficacion.          ');
          writeLn('          [ P ]          | Procesado de la Secuencia.          ');
          if suminc <> 0 then
            begin
              writeLn('          [ I ]          | Continuar procesado.          ');
            end
          if numfil >1 then
            begin
              writeLn('          [ A ]          | Formar un archivo.          ');
              writeLn('          [ S ]          | Salir del Programa.          ');
            end
          end;
    end;
  end;

```

```
writeln('
writeln('
writeln('
writeln('
writeln('
exit;
end;
```



```
);
);
);
);
);
```

```
2 : begin
if ind2 <> 0 then goto 20;
```

```
gotoxy(1,4);
```

```
writeln('
```

```
exit;
```

```
20:
```

```
gotoxy(11,ind2-4);writeln('
```

```
gotoxy(11,ind2-3);writeln('
```

```
gotoxy(11,ind2-2);writeln('
```

```
gotoxy(11,ind2-1);writeln('
```

```
gotoxy(11, ind2 );writeln('
```

```
gotoxy(11,ind2+1);writeln('
```

```
exit;
```

```
end;
```

Parametros de graficacion

I intervalo de Graficacion de ordenadas.
L largo de la ventana.
T raslape entre ventanas.
A bsisa inicial.
M axima distorcion.

EBC Para salir del menu.

Valor en uso

Nuevo valor

```
3 : begin
gotoxy(1,4);
```

```
writeln('
```

Opciones de proceso

procesado Total

graficado por Etapas

EBC Para salir del menu.

C Para continuar.

```

exit;
end;

4 | begin
  gotoxy(7,15);writeln('          procesando...');
  gotoxy(7,16);writeln('          r          c');
  gotoxy(7,17);writeln('          falla->');
  gotoxy(7,18);writeln('          fl1 ----> 0 , 0;');
  gotoxy(7,19);writeln('          inc ---->');
  gotoxy(7,20);writeln('          inct ---->');
  gotoxy(7,21);writeln('          ');
  gotoxy(7,22);writeln('          ');
  gotoxy(40,17);write(numfl1:2);
  gotoxy(40,18);write(suminc:2);
  exit;
end;

5 | begin
  gotoxy(1,1);
  writeln('          FALLA No : ');
  writeln('          ');
  exit;
end;

6 | begin
  gotoxy(36,20);writeln('          tiempo de p. ');
  gotoxy(36,21);writeln('          ');
  gotoxy(36,22);writeln('          ');
  exit;
end;

7 | begin
  if ind2 = 1 then goto 10;
  restorewindow(15,0,0);

  str(round(suminc#inc):4,numero);
  drawtext(5:2,3,1,numero);

  str(windpgs:2,numero);
  drawtext(38:3,3,1,numero);

10|exit;
end;

8 | begin
  gotoxy(44,ind2);writeln('          Nombre del Archivo : ');
  gotoxy(44,ind2+1);writeln('          ');
  gotoxy(44,ind2+2);writeln('          ');
  gotoxy(44,ind2+3);writeln('          ');
  exit;
end;

9 | begin
  gotoxy(44,ind2+4);writeln('          ');

```

```

gotoxy(44,ind2+5);writeln(' Inserte Disco y Presione <ENTER> ');
gotoxy(44,ind2+6);writeln(' para Continuar ');
gotoxy(44,ind2+7);writeln(' ');
read;
exit;
end;

```

```

end; ( fin de la case )
end; ( fin de la subrutina )

```

```

procedure EKR(ind:integer);
var ch : char;
begin
  case ind of
    1 : begin
        writeln;
        writeln('----> Numero de Columnas y/o Renglones Fuera de Rango');
        halt;
        end;
    2 : begin
        writeln;
        writeln('----> Numero de Puntos de Control Fuera de Rango');
        halt;
        end;
    3 : begin
        gotoxy(1,24);
        write('----> Abortar Procesado... ? (S/N) ');
        ($C,U+)
        repeat
          read(kbd,ch); ch := upcase(ch);
          if ch = 'S' then halt;
        until ch = 'N';
        ($C,U-)
        gotoxy(1,24); clrscr;
        end;
    4 : begin
        gotoxy(1,21);
        writeln('### PRECAUCION ');
        writeln('----> Alterar estos parametros puede generar distorsion en');
        writeln('----> el graficado ... ');
        end;
    5 : begin
        writeln;
        writeln('----> Numero de fallas sobrepasado...');
        end;
    6 : begin
        gotoxy(1,24);

```

```
write('----> Interruption graficado... ? (S/N) ');  
end;
```

```
end;  
end;
```

```
( ARCHIVO FUNC.INC:
  FUNCIONES.      )
```

```
(***** PAQUETE DE FUNCIONES *****)
```

```
function tan(ang:real):real;    ( Calculo de la tangente de un angulo )
begin
  tan:=sin(ang)/cos(ang);
end;

function norma(x1,y1,x2,y2:real):real;    ( Calculo de la norma de un segmento
dirigido entre dos puntos
(x1,y1) y (x2,y2) )
begin
  norma:=sqrt(sqr(abs(x2-x1))+sqr(abs(y2-y1)));
end;

function pend(y2,y1,x2,x1:real):real;    ( Funcion para calcular la pendiente
de la recta que pasa por dos
puntos conocidos (x1,y1), (x2,y2) )
begin
  if abs(x2-x1)<=1E-5 then pend:=1E17
  else
    pend:=(y2-y1)/(x2-x1);
  end;

function bisectriz(y2,y1,ang,x1:real):real;
( Funcion que calcula la ordenada de la bisectriz conociendo la abscisa (y2),
el angulo de la pendiente de la bisectriz y un punto (x1,y1) )
begin
  bisectriz:=(y2-y1)/tan(ang)+x1;
end;

function despeque(x2,x1,m,y):real;
( Funcion que calcula la abscisa del sistema de fallas conociendo la ordenada
(x2), un punto (x1,y1) y la pendiente del segmento recto sobre el que se
encuentra el punto )
begin
  despeque:=(x2-x1)#m+y1;
end;
```

```
( ARCHIVO SUBNUME.INC
  SUBRUTINAS NUMERICAS )
```

```
(***** PAQUETE DE SUBRUTINAS *****)
```

```
procedure malla(m,n:integer;hor,ver:real;var red1:MtrzPoint;
               var red2:MtrzExpPoint);
```

```
( Subrutina para generar una malla regularmente espaciada. Se introduce el
  rango (m*n) y la separacion horizontal y vertical entre punto y punto.
  La genera a partir de sumatorias )
```

```
var i,j:integer;
    sumx,sumy:real;
begin
  suay:=0;
  for i:=1 to m do
    begin
      sumy:=sumy+ver;
      sumx:=0;
      for j:=1 to n do
        begin
          suax:=suax+hor;
          if j <= 100 then
            begin
              red1.x[i,j]:=sumx;
              red1.y[i,j]:=sumy;
            end
          else
            begin
              red2.x[i,j]:=sumx;
              red2.y[i,j]:=sumy;
            end;
          end; { fin del ciclo interno }
        end; { fin del ciclo externo }
      end; { fin de la subrutina }
```

```
procedure vectores_unit(n:integer;px,py:vector;var vecx,vecy:vector);
```

```
( Subrutina que calcula los vectores unitarios de desplazamiento asociado a
  cada segmento recto del sistema de fallas, de acuerdo a la expresion :
```

$$e(x,y) = (a(x,y) - b(x,y)) / \sqrt{a(x,y)^2 + b(x,y)^2}$$

```
vecx y vecy son los argumentos imaginarios de rx y ry )
```

```
var i:integer;
begin
  for i:=1 to n do
    begin
      vecx[i]:= (px[i+1]-px[i])/norma(px[i+1],py[i+1],px[i],py[i]);
      vecy[i]:= (py[i+1]-py[i])/norma(px[i+1],py[i+1],px[i],py[i]);
    end;
```

```

end; ( fin del procedimiento )

procedure fallas(nf,np:integer;px,py:vector;var npf:vector3;hor:real;
                var falla:Mtr2Point);
( Subrutina para interpolar puntos (a manera de referencia, para poder -
deformarlos posteriormente) entre los puntos de control de las fallas )
var i,j,k:integer;
    x:real;

begin
  x:=px[i];
  k:=0;

  for i:=1 to np do
    for j:=1 to trunc((px[i+1]-px[i])/hor) do
      begin
        k:=k+1;
        if (k = 201) or (x > px[np+1]) then exit; ( Algoritmo de interpolacion de )
        npf[nf]:=k; ( acuerdo a una recta )
        x:=x+hor;
        falla^.x[nf,k]:=x;
        falla^.y[nf,k]:= (x-px[i])$pend(py[i+1],py[i],px[i+1],px[i])+py[i];
      end;
    end;
  end; ( Fin de la subrutina )

procedure bisectrices(n:integer;px,py:vector;var bis:vector);
( Subrutina para calcular la pendiente de las bisectrices entre dos tramos
rectos del sistema de fallas.
Donde: teta1, teta2 y beta son variables temporales que representan el
angulo de las pendientes de los segmentos rectos (teta1 y teta2) y
el angulo bisector entre estos (beta). )
var
  teta1,teta2,beta:real;
  i:integer;
const pi=3.1415926;
begin
  for i:=2 to n do
    begin
      teta1:=arctan(pend(py[i],py[i-1],px[i],px[i-1]));
      teta2:=arctan(pend(py[i+1],py[i],px[i+1],px[i]));
      beta:=teta1+pi-teta2;
      bis[i]:=teta2+beta/2;
    end;
    bis[1]:=pi/2; bis[n+1]:=pi/2;
  end; ( fin del procedimiento )

procedure desliz_c_c(red:Mtr2Point;ms:integer;cx,cy:vector;n:integer;
                    var ciz:vector2);

```

```

( Subrutina que genera incrementos adicionales por deslizamiento capa por
  capa )
var i,j:integer;          ( Los genera de acuerdo a triangulos )
begin                    ( congruentes )
  for i:=1 to n do
    for j:=1 to n-1 do
      begin
        if (red^.y[i,j] > cy[j]) and (red^.y[i,j] <= cy[j+1])
          then
            cz[i]:=cx[j]+(red^.y[i,j]-cy[j])*(cx[j+1]-cx[j])/(cy[j+1]-cy[j]);
          end;
        end;
      end;
    end;
  end; ( Fin de la subrutina )

```

```

procedure select_posicion(xp,yp:real;n:integer; px,py,bis:vector;
  var pos:integer;var pi:boolean);
( Verifica si un punto es desplazable y la region a la que pertenece )
( donde xp y yp es el punto a probar )
var i:integer;
    pendiente:real; ( Variable temporal para calcular las pendientes de las )
                    ( fallas )
begin
  for i:=1 to n do
    begin
      pendiente:=pend(py[i+1],py[i],px[i+1],px[i]);
      pi:=false;
      if ((bisectriz(yp,py[i],bis[i],px[i]) <= xp) ( Encuentra la region a )
          and (xp < bisectriz(yp,py[i+1],bis[i+1],px[i+1])) ( la que pertenece (xp, )
          and (yp >= despegue(xp,px[i],pendiente,py[i])) ( yp), limitada por las )
          and (bisectriz(yp,py[i],bis[i],px[i]) <= xp) ( bisectrices del sistema)
          and (yp >= despegue(xp,px[i],pendiente,py[i])) ( de fallas, y si se en- )
          and (bisectriz(yp,py[i],bis[i],px[i]) <= xp) ( cuentra sobre esta )
          and (yp >= despegue(xp,px[i],pendiente,py[i])) (
            then
              begin
                ( Si es verdadero el "if" entonces: )
                pos:=i; ( Asigna a "pos" la region donde se encuentra el punto )
                pi:=true; ( La prueba es verdadera )
                exit ( Da fin a la subrutina )
              end;
            end;
          end;
        end;
      end;
    end;
  end; ( Fin de la subrutina )

```

```

procedure cambio_desplazamiento(x1,y1,px,py,b,delta,rx,ry:real;var direal;
  var p2:boolean);
( La siguiente subrutina verifica si un punto de la red se encuentra )
( adyacente a un eje axial, para modificar su trayectoria. Donde (x1,y1) es )
( el punto a probar )
var
  x2,y2, ( Posicion a la que debe desplazarse (x1,y1) )
  m1,m2, ( Pendiente de la recta que pasa por (x1,y1) y (x2,y2) )
  c1,c2, ( y pendiente de la recta bisectriz )
          ( Terminos independientes de las ecuaciones de las )

```

```

x,y:real;          ( rectas a anteriores )
                   ( Punto de interseccion de las rectas )
begin
  p2:=false;      ( Valor por default de la prueba )
  x2:=x1+deltax;  ( Calculo de la posicion desplazada )
  y2:=y1+deltay;
  m1:=pend(y2,y1,x2,x1);
  m2:=tan(b);     ( Calculo de las pendientes )
  c1:=y1-x1*m1;   ( Terminos independientes )
  c2:=py-px*m2;
  x:=(c2-c1)/(m1-m2);
  y:=(c2*m1-c1*m2)/(m1-m2);
  ( Calculo del punto de interseccion )
  if norma(x1,y1,x,y) < delta
  then
  begin
    p2:=true;
    d:=norma(x1,y1,x,y);
    end;
  ( Si la distancia entre los puntos )
  ( (x1,y1) y (x2,y2) es menor que la )
  ( distancia que debe desplazarse, )
  ( la prueba es verdadera, y se asig-)
  ( na a "d" la diferencia de las dis-)
  ( tancias )
end; ( fin de la subrutina )

procedure traslacion(var x,y:real;DeltaX,DeltaY:real);
( Subrutina para calcular el desplazamiento que sufre un punto, a partir, )
( de los incrementos deltax,deltay )
begin
  x:=x+DeltaX;
  y:=y+DeltaY;
end; ( fin de la subrutina )

procedure Tiempo(var h,a,s:integer);
type
  repack = record
    ax,bx,cx,dx,bp,si,di,de,es,flags: integer;
  end;
var
  repack1: repack;
  ah,ai,ch,ci,dh: byte;
  hour,min,sec: string[23];
  vt: integer;
begin
  ah := $2c;
  with repack do
  begin
    ax := ah shl 8 + ai;

```

```

end;
intr (@Z1,recpack);
with recpack do
begin
  str(cx shr 8, hour);
  str(cx mod 256, min);
  str(dx shr 8, sec);
end;
val (hour, h, vt);
val (min, m, vt);
val (sec, s, vt);

end;

procedure TiempoProc;
begin
  menu(6,0);
  if (m1 > m2) then
    if (m1 > m2) then
      begin
        gotoxy(51,21);writeLn(h2-h1-1;1,';',m2-m1+60;2,';',m2-m1+60;2);
      end
    else
      begin
        gotoxy(51,21);writeLn(h2-h1;1,';',m2-m1-1;2,';',m2-m1+60;2);
      end
    else
      if m1 > m2 then
        begin
          gotoxy(51,21);writeLn(h2-h1-1;1,';',m2-m1+60;2,';',m2-m1;2);
        end
      else
        begin
          gotoxy(51,21);writeLn(h2-h1;1,';',m2-m1;2,';',m2-m1;2);
        end;
      delay(2000);
    end;
end;

```

(ARCHIVO SUBBKAF.INC)

(***** SUBRUTINAS DE GRAFICADO *****)

```
procedure portada;
var
  i,j,k:integer;

function AleatorioX(Intervalo:integer):integer;
var x:integer;
begin
  repeat
    x:=random(81);
  until (x >= 40-intervalo) and (x <= 40+intervalo);
  aleatoriox:=x;
end;

function AleatorioY(Intervalo:integer):integer;
var y:integer;
begin
  repeat
    y:=random(26);
  until (y >= 13-intervalo) and (y <= 13+intervalo);
  aleatorioy:=y;
end;

begin

  entergraphic;
  definewindow(1,0,0,XmaxG1b,YmaxG1b);
  defineworld(1,0,700,1000,0);
  selectworld(1);
  selectwindow(1);
  setaspect(1);

  drawcircle(300,300,1.1);    ( CINTURON )
  drawcircle(300,300,0.8);

  drawline(180,300,420,300);
  drawline(180,298,420,298);
  drawline(185,250,415,250);
  drawline(185,248,415,248);    ( INTERIOR )
  drawline(310,180,300,300);
  drawline(180,300,230,330);
  drawline(230,330,420,300);

  delay(200);

  drawline(260,190,340,190);
  drawline(260,130,340,130);
  drawline(260,190,260,130);    ( EVILLA )
  drawline(340,190,340,130);
```

```

drawline(260,190,230,150);
drawline(260,130,230,90);
drawline(230,150,230,90);
( cinturón )

delay(200);

drawline(400,500,410,550);
drawline(400,500,590,400);
drawline(410,540,590,400);
( PICO )

drawline(200,500,45,400);
drawline(165,540,8,435);
drawline(200,500,165,540);
drawline(45,400,8,435);
( MARTILLO )

delay(200);

drawline(115,445,161,400);
drawline(445,115,400,160);
drawline(135,460,177,418);
drawline(465,125,410,178);
drawline(465,125,445,115);
( MANGO MARTILLO )

drawline(450,465,408,423);
drawline(471,456,427,413);
drawline(150,115,201,168);
drawline(130,125,178,178);
drawline(150,115,130,125);
( MANGO PICO )

delay(200);

drawline(600,100,600,200);
drawline(600,200,660,200);
drawline(600,150,650,150);
( 'F' )

delay(200);

drawline(680,100,720,200);
drawline(720,200,760,100);
drawline(700,150,740,150);
( 'A' )

delay(200);

drawline(780,100,780,200);
drawline(780,100,830,100);
( 'L' )

delay(200);

drawline(850,100,850,200);
drawline(850,100,900,100);
( 'L' )

delay(200);

drawline(920,100,960,200);
drawline(960,200,1000,100);
( 'A' )

```

```

drawline(940,150,980,150);
delay(200);
drawborder;
delay(1000);
for i:= 1 to 40 do
for j:= 1 to 12 do
if (i=j) or (i>12) then
begin
for k:=1 to 30 do
begin
gotoxy(aleatorioX(i),aleatorioY(j));
write(chr(178));
end;
end;
delay(2000);
leavegraphic;
end; { fin de la subrutina }

procedure selec_coord_ventana(k,inds:integer);
begin
defineworld(1,xinf+k*largoPantalla-(k-1)*straslape-largoPantalla,ysup,
xinf+k*largoPantalla-(k-1)*straslape,yinf);
if ind <> 0 then definewindow(k,0,0,XmaxBib,trunc((9/10)*YmaxBib))
else definewindow(k,0,0,XmaxBib,YmaxBib);
selectworld(1);
selectwindow(k);
end;

procedure graficado_fallas; forward;
procedure graficado_bisectrices; forward;

procedure graficado_malla;
{ Subrutina de graficado de la red de puntos de referencia }
var i,j,k:integer;
npr,
maxvalx:real;
cht:char;
begin
(0C-,U-)
maxvalx:=0;

for i:=1 to m do
for j:=1 to n do
begin
case j of
1..100 : if red^.x[i,j] > maxvalx then maxvalx:=red^.x[i,j];

```

```

101..200 i if red2^.x[i,j] > maxvalx then maxvalx:=red2^.x[i,j]
end; { fin del "case" }
end;

numwin:=round((maxvalx-xinf)/largopantalla)+1;
if (maxvalx-xinf) < largopantalla then numwin:=1;
numwin:=round(numwin+(traslape#numwin/largopantalla));
if numwin > 10 then numwin:=10;

k:=0;
repeat

  k:=k+1;
  clearscreen;
  selec_coord_ventana(k,1);

  for j:=1 to n do
  for i:=1 to m do
  begin

    if keypressed then
    begin
      ERR(6);
      repeat
        read(kbd,ch);
        if upcase(ch) = 'S' then
        begin
          numwin:=k-1;
          exit;
        end;
      until upcase(ch) = 'N';
      gotoxy(1,24);writeln(' ');
    end;

    if j <= 100 then
      drawpoint(red^.x[i,j],red^.y[i,j])      ( Grafica los puntos )
    else
      drawpoint(red2^.x[i,j],red2^.y[i,j]);  ( de referencia )

    case j of
    1..99 :begin
      ( Restriccion entre los puntos verticales adyacentes )
      ( Si sufre una elongacion despues de la deformacion )
      ( mayor al 33% , no se grafica )

      if i < m then
      begin
        nor:=norma(red^.x[i,j],red^.y[i,j],red^.x[i+1,j],red^.y[i+1,j]);
        if nor <= maxalongsver then
          drawline(red^.x[i,j],red^.y[i,j],red^.x[i+1,j],red^.y[i+1,j]);
      end;
    end;
  end;

```

```

( Restriccion entre los puntos horizontales adyacentes )
( Si sufre una elongacion despues de la deformacion )
( mayor al 33% , no se grafica )

if j < n then
begin
nor:= norma(red^.x[i, j],red^.y[i, j],red^.x[i, j+1],red^.y[i, j+1]);
if nor <= maxelongx$hor then
drawline(red^.x[i, j],red^.y[i, j],red^.x[i, j+1],red^.y[i, j+1]);
end;
end;

101..200:begin
if i < m then
begin
nor:= norma(red2^.x[i, j],red2^.y[i, j],red2^.x[i+1, j],
red2^.y[i+1, j]);
if nor <= maxelongx$ver then
drawline(red2^.x[i, j],red2^.y[i, j],red2^.x[i+1, j],
red2^.y[i+1, j]);
end;
if j < n then
begin
nor:= norma(red2^.x[i, j],red2^.y[i, j],red2^.x[i, j+1],
red2^.y[i, j+1]);
if nor <= maxelongx$hor then
drawline(red2^.x[i, j],red2^.y[i, j],red2^.x[i, j+1],
red2^.y[i, j+1]);
end;
end;

100:begin
if n > 100 then
begin
nor:= norma(red^.x[i, j],red^.y[i, j],red2^.x[i, j+1],
red2^.y[i, j+1]);
if nor <= maxelongx$hor then
drawline(red^.x[i, j],red^.y[i, j],red2^.x[i, j+1],
red2^.y[i, j+1]);
if i < m then
begin
nor:= norma(red^.x[i, j],red^.y[i, j],red^.x[i+1, j],
red^.y[i+1, j]);
if nor <= maxelongx$ver then
drawline(red^.x[i, j],red^.y[i, j],red^.x[i+1, j],
red^.y[i+1, j]);
end;
end;
end;

end; ( Fin del "case" )

```

```

end; ( Fin de los ciclos anidados )

selec_coord_ventana(k,0);
drawaxis(8,0,0,0,0,0,-1,-1,false);
selec_coord_ventana(k,1);
graficado_fallas;
graficado_bisectrices;
selec_coord_ventana(k,0);
storewindow(k);

until k = numwin;
{C+,U+}
end; ( fin de la subrutina )

procedure graficado_fallas;
( Subrutina para graficar el sistema de fallas )
var i,j,vt:integer;
    t1,t2:boolean;
begin
  for i:=1 to np do
    drawline(ptox[i],ptoy[i],ptox[i+1],ptoy[i+1]);

    for j:=1 to numf11-1 do ( Si dos puntos contiguos ocupan la )
      for j:=1 to numptf[i]-1 do ( pertenecen al mismo bloque (aloc- )
        begin ( no o autoctono) respecto de la fa- )
          ( lla en "uso" se unen con una linea )
          ( de lo contrario no se grafica )
          select_posicion(falla^.x[i],j],falla^.y[i],j],np,ptox,ptoy,penbis,vt,t1);
          select_posicion(falla^.x[i],j+1],falla^.y[i],j+1],np,ptox,ptoy,penbis,vt,
            t2);

          if t1 = t2 then
            drawline(falla^.x[i],j],falla^.y[i],j],falla^.x[i],j+1],falla^.y[i],j+1]);
          end;
        end;
      end;
    end;
  end; ( fin de la subrutina )

procedure graficado_bisectrices;
( Subrutina para graficar las bisectrices asociadas a las inflexiones del )
( sistema de fallas. )
( La variable 'x' se genera a partir de la ecuacion de una recta, que toma )
( como puntos conocidos a px[i],py[i] y la pendiente de la bisectriz (bis[i]) )
var x:real;
    i:integer;
begin
  for i:=1 to np-1 do
    begin
      x:=(5000-ptoy[i])/tan(penbis[i])+ptox[i];
      drawline(ptox[i],ptoy[i],x,5000);
    end;
    delay(1000);
  end; ( fin del procedimiento )

```

(ARCHIVO SUBPRINC. INC)

(===== SUBRUTINAS PRINCIPALES =====)

Procedure Inicializacion;

```
begin
  initgraphic;
  VentanaMenu;
  Portada;

  new(red);
  new(falla);

  for i:=1 to m do
    inciz[i] := 0;
    numfil := 1;
    suminc := 0;
    destot := 0;
    ind := 0;

  Yinf := 0;
  Ysup := 675;
  Xinf := 0;
  LargoPantalla:=1000;
  Traslape := 250;
  MaxElongX :=1.33;
  MaxElongY :=1.33;

  procesado := true;
  datos := false;
  graficado := false;
end;
```

(=====)

procedure EntradaDatos;

```
label 10;
begin
  if numfil > 1 then goto 10;

  clrscr;
  writeln;
  write(' Rango de la malla (renglones#columnas): ');read(m,n);
  if not( a in[1..MaxNumReng]) then ERR(1);
  if not( n in[1..MaxNumColumnExp]) then ERR(1);

  if n > 100 then new(red2);

  writeln;
  write(' Espaciamiento horizontal y vertical entre punto y punto: ');
  readln(hor,ver);
```

```

malla(m,n,hor,ver,red,red2);      ( Llamado a la subrutina para obtener los )
                                  ( puntos de la red )
10:clrcsr;
menu(5,0);gotoxy(16,2);write(numf1);
gotoxy(1,5);
write(' Numero de puntos de que consta el despegue basal: ');
read(np);
if not( np in [1..MaxNumPts] ) then ERR(2);

writeln;
writeln;

for i:=1 to np do
begin
  write('Px(',i,')= ');read(ptox[i]);
  write(', Py(',i,')= ');read(ptoy[i]);
  writeln;
end;

ptox[np+1]:=ptox[np]+(ptox[np]-ptox[1]);      ( Se genera un punto extra para )
ptoy[np+1]:=ptoy[np];                        ( asegurar el desplazamiento )

writeln;
write(' Deslizamiento sobre el despegue: ');read(desliz);
writeln;writeln;
write(' Numero de etapas: ');read(ne);
writeln;
writeln;

writeln('Cizalla simple (s/n) ?');
read(kbd,ch);
writeln;

ch:=upcase(ch);
if ch = 'B' then
begin
  for i:=1 to a do
  inciz[i] := 0;
  write(' Numero de puntos de control de deslizamiento: ');
  read(nc);
  writeln;
  writeln;
  for i:=1 to nc do
  begin
    write('Px(',i,')= ');read(cizx[i]);
    write(', Py(',i,')= ');read(cizy[i]);
    writeln;
  end;
end; ( Fin del 'then' )

if ch = 'B' then desliz_c_c(red,m,cizx,cizy,nc,inciz);

clrcsr;

end; ( fin de la subrutina )

```

(=====)

procedure ProcesadoDatos;

begin

(OK-)

for j:=1 to n do

for i:=1 to m do

begin

inc:=(desliz+inciz[i])/ne;

if keypressed then ERR(3);

gotoxy(19,19);writeln(i:3);

if i = 1 then begin gotoxy(25,19);writeln(j:3) end;

if (i=1) and (j=1) then

begin

destot:=destot+(suminc#inc)-((suminc-1)#inc);

gotoxy(40,19);write(suminc#inc:4:2);

gotoxy(40,20);write(destot:4:2);

end;

case j of

1..100:begin

select_posicion(red^.x[i,j],red^.y[i,j],np,ptox,ptoy,penbis,reg,test);

if test = true then

begin

cambio_desplazamiento(red^.x[i,j],red^.y[i,j],ptox[reg+1],ptoy[reg+1],
penbis[reg+1],inc,rx[reg],ry[reg],dif,test2);

if test2 = true then

begin

(Si es verdadera la prueba entonces:)

(El primer desplazamiento se efectua hasta el eje axial)

(y el segundo a partir de este con el nuevo vector de)

(desplazamiento)

traslacion(red^.x[i,j],red^.y[i,j],dif#rx[reg],dif#ry[reg]);

traslacion(red^.x[i,j],red^.y[i,j],(inc-dif)#rx[reg+1],
(inc-dif)#ry[reg+1]);

end

else

(Si es falsa ...)

(El punto no es adyacente a un eje y se desplaza a lo largo del)

(vector asociado al segmento de falla)

traslacion(red^.x[i,j],red^.y[i,j],inc#rx[reg],inc#ry[reg]);

```

end; ( fin del primer if )
end; ( fin de la primera sentencia compuesta del "case" )

101..200:begin
  select_posicion(red2^.x[i,j],red2^.y[i,j],np,ptox,ptoy,penbis,reg,test)
  if test = true then
  begin
    cambio_desplazamiento(red2^.x[i,j],red2^.y[i,j],ptox[reg+1],ptoy[reg],
      penbis[reg+1],inc,rx[reg],ry[reg],dif,test2)
    if test2 = true then
    begin
      traslacion(red2^.x[i,j],red2^.y[i,j],dif#rx[reg],dif#ry[reg]);
      traslacion(red2^.x[i,j],red2^.y[i,j],(inc-dif)#rx[reg+1],
        (inc-dif)#ry[reg+1]);
    end
    else
      traslacion(red2^.x[i,j],red2^.y[i,j],inc#rx[reg],inc#ry[reg]);
    end; ( fin del primer "if" )
  end; ( fin de la segunda sentencia compuesta del "case" )
end; ( fin del case )
end; ( fin de los ciclos anidados i,j )

```

(Repite el paquete para los sistemas de fallas)

```

for i:=1 to numfl-1 do
for j:=1 to numptf[i] do
begin
  if keypressed then ERR(3);
  gotoxy(19,20);writeln(i:3);
  if i = 1 then begin gotoxy(25,20);writeln(j:3) end;
  inc:=(desliz+inciz[i])/ne;
  select_posicion(falla^.x[i,j],falla^.y[i,j],np,ptox,ptoy,penbis,
    reg,test);
  if test = true then
  begin
    cambio_desplazamiento(falla^.x[i,j],falla^.y[i,j],ptox[reg+1],
      ptoy[reg+1],penbis[reg+1],inc,rx[reg],ry[reg],dif,test2);
  end;
end;
end;
end;

```

```

    if test2 = true then
    begin
        traslacion(falla^.x[i],j],falle^.y[i],j],dif#rx[reg],dif#ry[reg]);
        traslacion(falla^.x[i],j],falle^.y[i],j],(inc-dif)#rx[reg+1],
            (inc-dif)#ry[reg+1]);
    end
    else
        traslacion(falla^.x[i],j],falle^.y[i],j],inc#rx[reg],inc#ry[reg]);
    end; ( fin del primer if )

end; ( fin de los ciclos anidados )
(*K+)
end; ( fin de la subrutina )

```

(=====)

```

procedure Opciones(ops:integer);
const
    ESC = #27;
    ENTER = #13;
var vt:char;
    nombre:string[80];
procedure archivo_salida;
var i,j:integer;
    archivo:text;
begin
    menu(8,10);
    gotoxy(47,12);
    read(nombre);
    menu(9,9);
    assign(archivo,nombre);
    rewrite(archivo);
    writeIn(archivo,'Nombre del archivo: ',nombre);
    writeIn(archivo,'Consta de: ',#n:4,' puntos');
    writeIn(archivo,' ');
    for i:=1 to n do
        for j:=1 to m do
            if j <= 100 then
                writeIn(archivo,red^.x[i],j]:4:2,red^.y[i],j]:4:2)
            else
                writeIn(archivo,red2^.x[i],j]:4:2,red2^.y[i],j]:4:2);
        delay(1000);
    end; ( fin de la subrutina )

procedure pantalla_salida;
begin
    menu(8,12);
    gotoxy(47,14);
    read(nombre);
    menu(9,11);

```

```

restorewindow(windpg,0,0);
savescreen(ncmbre);
delay(1000);

end; ( fin de la subrutina )

begin ( Subrutina Opciones )
case ops of
1 : begin
  clrscr;
  repeat
  menu(1,0);
  if suminc = 0 then
  repeat read(kbd,ch) until upcase(ch) in ['E','G','C','P','S'];
  else
  repeat read(kbd,ch) until upcase(ch) in ['F','G','C','P','S','I','A'];
  ch:=upcase(ch);
  case ch of
'E': begin
  Datos:=true;
  EntradaDatos;
  end;

'F': begin
  if suminc <> 0 then numfil:=numfil+1;
  if not(numfil in [1..MaxNumFil]) then ERR(5)
  else
  begin
  EntradaDatos;
  suminc:=0;
  end;
  end;

'G': begin
  Procesado:=false;
  graficado:=true;
  end;

'C': Opciones(2);

'P': Opciones(3);

'I': begin
  clrscr;
  if suminc <> 0 then procesado:=true else ch:=' ';
  end;

'A': begin
  archivo_salida;
  menu(1,0);
  end;

'S': Halt;

end; ( fin del case)

```

```

    until ((ch = 'B') or (ch = 'P')) or (ch = 'I');
    exit;
end;

2 : begin
  (SC-,U-)
  menu(2,0);
  repeat read(kbd,ch) until upcase(ch) in ['I','L','T','A','M',ESC];
  ch:=upcase(ch);
  case ch of
    'I': begin
      ERR(4);
      menu(2,8);
      gotoxy(24,5);write('Yinf');
      gotoxy(13,6);write(yinf:7:2);
      gotoxy(13,8);read(yinf);
      menu(2,8);
      gotoxy(24,5);write('Ysup');
      gotoxy(13,6);write(ysup:7:2);
      gotoxy(13,8);read(ysup);
      gotoxy(1,21);clrrow;
      gotoxy(1,22);clrrow;
      gotoxy(1,23);clrrow;
      menu(2,0);
      end;

    'L': begin
      ERR(4);
      menu(2,10);
      gotoxy(13,8);write(largopantalla:7:2);
      gotoxy(13,10);read(largopantalla);
      gotoxy(1,21);clrrow;
      gotoxy(1,22);clrrow;
      gotoxy(1,23);clrrow;
      menu(2,0);
      end;

    'T': begin
      menu(2,12);
      gotoxy(13,10);write(traslape:7:2);
      gotoxy(13,12);read(traslape);
      menu(2,0);
      end;

    'A': begin
      menu(2,14);
      gotoxy(13,12);write(xinf:7:2);
      gotoxy(13,14);read(xinf);
      menu(2,0);
      end;

    'M': begin
      menu(2,16);
      gotoxy(24,13);write(' (X) ');

```

```

        gotoxy(13,14);write(MaxElongX;7;2);
        gotoxy(13,16);read(MaxElongX);
        menu(2,16);
        gotoxy(24,13);write(' (Y)');
        gotoxy(13,14);write(MaxElongY;7;2);
        gotoxy(13,16);read(MaxElongY);
        menu(2,0);
        end;

end; { fin del case }
until ch = ESC;
{EC+,U+}
ch:='C';
exit;
end;

3 : begin
{EC-,U-}
menu(3,0);
repeat
repeat read(kbd,ch) until upcase(ch) in ['T','E',ESC,'C'];
ch:=upcase(ch);
case ch of
'T': begin
graficador:=false;
gotoxy(23,7);write(chr(219));
gotoxy(26,9);write('E');
gotoxy(23,7);
end;

'E': begin
graficador:=true;
gotoxy(23,7);write('T');
gotoxy(26,9);write(chr(219));
gotoxy(26,9);
end;

ESC: begin
menu(1,0);
exit;
end;

end; { fin del case }
until ch = 'C';
procesador:=true;
{EC+,U+}
ch:='P';
exit;
end;

4 : begin
{EC-,U-}
repeat
if numwin <> 0 then restorewindow(windpg,0,0);
menu(7,ind);

```

```

repeat read(kbd,ch) until upcase(ch) in ['I','P','H','R','C','Z','M'];
ch:=upcase(ch);
case ch of
  'I': hardcopy(false,1);
  'P': pantalla_salida;
  'H': if ind = 0 then ind := 1
        else ind := 0;
  'R': leavegraphic;
  'Z': begin
        if windpg > 1 then windpg:=windpg-1;
        restorewindow(windpg,0,0);
        end;
  'M': begin
        if windpg < numwin then windpg:=windpg+1;
        restorewindow(windpg,0,0);
        end;
  end; ( fin del "case" )
until (ch = 'C') or (ch = 'R');
($C+,U+)
exit;
end;

end; ( fin del "case" externo )

end; ( Fin de la subrutina )

```

UN

(ARCHIVO TIT.PAS)

program tit;

(\$i typedef.sys)
(\$i graphix.sys)
(\$i kernel.sys)
(\$i windows.sys)

begin

 initgraphic;
 definewindow(1,0,0,79,8);
 defineworld(1,0,8,640,0);
 selectworld(1);
 selectwindow(1);

 drawtext(5,3,1,'H Rm Men');
 drawline(0,0,64,0);
 drawline(0,8,64,8);
 drawline(0,0,0,8);
 drawline(64,0,64,8);

 drawtext(75,3,1,'R Rtr Men p');
 drawline(70,0,145,0);
 drawline(70,8,145,8);
 drawline(70,0,70,8);
 drawline(145,0,145,8);

 drawtext(155,3,1,'P Sv Pant');
 drawline(150,0,215,0);
 drawline(150,8,215,8);
 drawline(150,0,150,8);
 drawline(215,0,215,8);

 drawtext(225,3,1,'I Imprimir');
 drawline(220,0,290,0);
 drawline(220,8,290,8);
 drawline(220,0,220,8);
 drawline(290,0,290,8);

 drawtext(300,3,1,'<-Z,M->');
 drawline(295,0,345,0);
 drawline(295,8,345,8);
 drawline(295,0,295,8);
 drawline(345,0,345,8);

 drawtext(355,3,1,'VENTI');
 drawline(350,0,412,0);
 drawline(350,8,412,8);
 drawline(350,8,350,0);
 drawline(412,0,412,8);

```
drawtext(422,3,1,'DEGLIZAMIENTO:');  
drawline(417,0,565,0);  
drawline(417,8,565,8);  
drawline(417,0,417,8);  
drawline(565,0,565,8);
```

```
savewindow(1,'menu-sup'); ( subrutina para gravar en disco la ventana )
```

```
repeat until keypressed;
```

```
end.
```