

# SELECCIÓN DE EVENTOS DENTRO DE UN POLÍGONO Y EN LA SUPERFICIE DE LA TIERRA CON *MATLAB*

José Frez C.

División de Ciencias de la Tierra, CICESE

## RESUMEN

Sea un catálogo de puntos en la superficie de la Tierra del cual necesitamos seleccionar un subconjunto con el criterio de que los puntos estén dentro de un polígono. El problema computacional tiene solución conocida a distancias locales y regionales ( $< \sim 500$  km) ya que, entonces, la superficie de la Tierra se puede aproximar como plana. El propósito de este trabajo es entregar un programa de cómputo que aprovecha funciones de *MATLAB* para resolver compactamente el problema. Al mismo tiempo, se divulgan técnicas de programación; en particular, de código vectorizado. El procedimiento se comprueba con un catálogo de 76545 hipocentros, cada uno descrito por 23 parámetros (columnas, en el catálogo). El catálogo contiene relocalizaciones de sismos que ocurren en la región fronteriza de ambas Californias (Shearer *et al.*, 2003). Un resultado lateral es que un nido con 17 epicentros ubicados en la falla Agua Blanca tiene un error sistemático fuerte y no señala una actividad de la falla.

**Palabras clave:** Mapa de epicentros, polígono, *MATLAB*.

## ABSTRACT

Consider a catalog of points on the surface of the Earth, from which we need to select a subset according to the criterion that the points be inside a polygon. The computational problem has a known solution for local and regional distances ( $< \sim 500$  km) since, for these ranges, the Earth's surface may be approximated by a plane. The purpose of the present work is to present a program that solves this program in a compact form, using *MATLAB* functions. At the same time, some programming techniques, particularly for vectorized code, are presented. The procedure is applied to a catalog containing 76545 hypocenters, each one described by 23 parameters (catalog columns). The catalog contains relocations of earthquakes occurring in the border region between both Californias (Shearer *et al.*, 2003). An incidental result is that 17 epicenters located on the Agua Blanca fault have a strong systematic error and do not indicate fault activity.

**Keywords:** Epicenter mapping, polygon, *MATLAB*.

## INTRODUCCIÓN

Un problema clásico de Geometría Computacional en dos dimensiones cartesianas es determinar si un punto está o no dentro de un polígono. Una solución puede encontrarse, por ejemplo en Sedgewick (1992); el lector debe cuidarse de errores en los códigos (escritos en C) de la versión en castellano (Sedgewick, 1995). Con estos resultados, podemos resolver el problema cuando los puntos están situados en la superficie de la Tierra, si es que aplicamos una aproximación plana que sea suficientemente exacta. Lo anterior nos permite obtener resultados correctos para mapeos regionales y locales (distancias menores a unos 500 km). A pesar de lo dicho, es común encontrar que los catálogos sísmicos no se acompañan con la herramienta computacional que mencionamos. Para mi trabajo, escribí hace años un largo programa en *Fortran*, con uso de la biblioteca de graficado *Plot88* (Young y Van Woert, 1992) y basado en Sedgewick (1992). Sin embargo, ahora, resulta más atractivo utilizar *MATLAB* (incluyendo la caja de herramientas para hacer mapas)

ya que este paquete de programas (doble precisión implícita) contiene la función *inpolygon* (que soluciona el problema en coordenadas cartesianas), sin contar con la facilidad de que se pueden codificar muy compactamente las lecturas, escrituras, cálculos y dibujos de mapas. Los programas resultantes son cortos, así como fáciles de leer y de adaptar a distintos objetivos.

El objetivo de esta publicación es divulgar técnicas de programación y entregar el listado de un programa (Figura 1) además de comentarios para programadores; también, se incluye un ejercicio de comprobación (figuras 2 y 3) que usa un catálogo grande de eventos sísmicos. Espero que el programa sea útil a investigadores y estudiantes de Ciencias de la Tierra y que incentive la escritura de programas más complicados.

En la sección siguiente, describiremos el programa; luego, la aplicación; finalmente, discutiremos un detalle interesante relacionado con la sismotectónica del norte de Baja California.

```

% -----
% Obtiene epicentros dentro de una ventana poligonal; hace el mapa
% correspondiente.
% Entradas:
% temp05.cat: Región 5 del catálogo de Shearer et al.(2003)
% poly.dat: Coordenadas geográficas de los vértices del polígono.
% Salida:
% temp06.cat: Subcatálogo de eventos, con epicentros dentro del
% polígono.
% Funciones externas:
% distaz: Calcula distancias y acimuts con el método de Richter
% linfallas: Lee y dibuja líneas (fallas, lagos), excepto costas.
% Todas las demás funciones son de MATLAB.
% Historia: J. Frez C. Febrero 2004, CICESE.
% -----
clear all,close all,
path(path,'c:\usur\1\MATLAB\mapas\mapsis') % Ruta de archivos a utilizar.
% Coordenadas del cuadro del mapa
lonul=-118.0; lonlr=-114.5; avlon=(lonul+lonlr)/2;
latul=33.5; latlr=31.00; avlat=(latul+latlr)/2;
% Calcula escala para transformación grados -> km.
[distx,az]=distaz(avlat,lonlr,avlat,lonul);
dglntkm=abs(distx/(lonlr-lonul));
[disty,az]=distaz(latul,avlon,latlr,avlon);
dgltkm=abs(disty/(latul-latlr));
% Archivo con coordenadas geográficas de los vértices del polígono;
% los reduce a coordenadas planas.
load 'poly.dat'
ypr=dgltkm*abs(poly(:,1)-latlr); xpr=dglntkm*abs(poly(:,2)-lonul);
% Carga el catálogo de eventos
load temp05.cat;
% Reduce las coordenadas de epicentros a planas.
epx=dglntkm*abs(temp05(:,9)-lonul);
epy=dgltkm*abs(temp05(:,8)-latlr);
% Aplica el filtro poligonal
in=inpolygon(epx,epy,xpr,ypr);
% Pasa el subcatálogo a temp06
temp06=temp05(in>0,:);
clear('temp05')
% Parametrización del mapa
hm=axesm('MapProjection','mercator',...
'MapLatLimit',[latlr latul], 'MapLonLimit',[lonul lonlr], ...
'AngleUnits','degrees',...
'Origin',[0 0 0], 'ScaleFactor',[1.0], ...
'frame','on','grid','on',...
'flatlim',[latlr latul], 'flonlim',[lonul lonlr], 'flinewidth',2,...
'mlinelimit',[latlr latul], 'plinelimit',[lonul lonlr],...
'MLabelLocation',[118 -117 -116 -115],
'PLabelLocation',[31 32 33], ...
'PLabelRound',[-1], 'MLabelRound',[-1],...
'FontSize',[11], ...
'ParallelLabel','on', 'MeridianLabel','on', ...
'MLineLocation',0.5, 'PLineLocation',0.5, ...
'GLineWidth',[0.6]);
hidem(gca) % "Limpia" la figura del marco entre ventanas.
textm(30.85,-117.0,'Longitud','FontSize',[13])
textm(32.80,-114.35,'Latitud','FontSize',[13],'Rotation',-90)
% Dibuja sismicidad; tamaño de símbolos proporcionales a la magnitud
% (columna 11)
scatterm(temp06(:,8),temp06(:,9),2*(temp06(:,11)+0.1),'b')
scaleruler % Dibuja escala del mapa
% Dibuja el polígono
plotm(poly(:,1),poly(:,2),'r','LineWidth',1.5);
% Dibuja líneas de costa y límite internacional que provienen de NGDC.
load c15854.dat;
[nl,nc]=size(c15854);
plotm(c15854(:,2),c15854(:,1),'k','LineWidth',1.2);
clear('c15854');
% Dibuja fallas, río Colorado y otras líneas geográficas.
linfallas;
% Escribe resultado en archivo de texto y formateado (primeras 15
% columnas)
nom(1:6)='temp06'; ext(1:4)='(.cat)';
filename=[nom,ext]; fid=fopen(filename,'w');
if(fid<0), 'Error: No pude abrir el archivo',filename; end
nx=length(temp06(:,1));
for k=1:nx,
fprintf(fid,'%4d %2d %2d %2d %2d %5.2f %9d %8.5f %9.5f %5.2f %5.2f
%3d %3d %5.2f %5.2f\n', ...
temp06(k,1:15));
end
fclose(fid);

```

Figura 1. Listado del programa de computación (MATLAB) utilizado para generar la Figura 3.

## INDICACIONES DE IMPLEMENTACIÓN

El programa (Figura 1) se inicia con la lecturas de parámetros, para pasar a la de dos archivos: uno contiene las coordenadas geográficas de los vértices de un polígono; el otro, el catálogo sísmico. En la aplicación, se leen los parámetros hipocentrales de 76545 eventos (filas), en 23 columnas. Después de reducir el problema a uno de superficie plana, el programa selecciona los eventos que están dentro del polígono, vuelve a coordenadas geográficas y dibuja el mapa correspondiente. El resultado es que un programa de menos de 100 líneas de un código muy transparente realiza la tarea; mi programa anterior (Fortran y Plot88) tenía varias veces ese número de líneas.

Con la recomendación de leer la documentación de las funciones de *MATLAB* que utiliza el programa, agregué algunas indicaciones que, por mi experiencia, me parecen útiles. Las variables y funciones se escriben en cursiva.

1. El catálogo se lee con la función *load* que lo despliega como una matriz. La limitación de ello es que la lectura es para formato libre y sólo para números. Esto implica que el catálogo debe estar libre de letras y símbolos no-numéricos. Además, todas las columnas deben estar llenas con valores numéricos (los blancos no deben reemplazar ceros), al menos con un hueco que los separen. Si estas condiciones no se cumplen, se puede reemplazar las letras o símbolos por números, introducir explícitamente los ceros y separar "a mano" los números que quedan juntos. Una opción más drástica es reemplazar *load* por una función que lea con formato (quizás, sólo a variables de interés); recomiendo, en ese caso, el uso vectorizado de *textread*.
2. Para transformar las coordenadas geográficas a planas, las coordenadas cartesianas son referidas al extremo izquierdo inferior del cuadrángulo del mapa. Las distancias en la superficie de la Tierra (en km) se calculan con el algoritmo de Richter (Lee y Stewart, 1981) que es usado en *Hypo71* (Lee y Lahr, 1975) y que incluye el efecto elipsoidal. La función *distaz* es una traducción a *MATLAB* de las fórmulas de Lee y Stewart (1981). Si al usuario no le gustan las cajas negras, puede tomar de algún texto de Geodesia, por ejemplo Pearson (1990), expresiones para calcular las distancias a lo largo de meridianos y paralelos. Utilizando el elipsoide WGS-72, se comprueba que las expresiones de Pearson (1990) entregan resultados que difieren en 3 m del algoritmo de Richter, que tiene integrado un elipsoide de referencia no explícito; esta diferencia se encuentra para valores de 1° a latitudes de 43° y 30°. Por supuesto que la aproximación plana está en el subsiguiente uso del teorema de Pitágoras para calcular la distancia. Un uso de esta aproximación plana para distancias de 500 km o más debe ser comprobado con experimentos numéricos.
3. No es necesario entregar la función *linfallas*. Ella sirve para dibujar las trazas de las fallas en la superficie y está compuesta por líneas de código semejantes a las cuatro usadas para dibujar las líneas de costa y que aparecen en el listado (Figura 1). Por lo tanto, es trivial escribir la versión de *linfallas* que se nece-

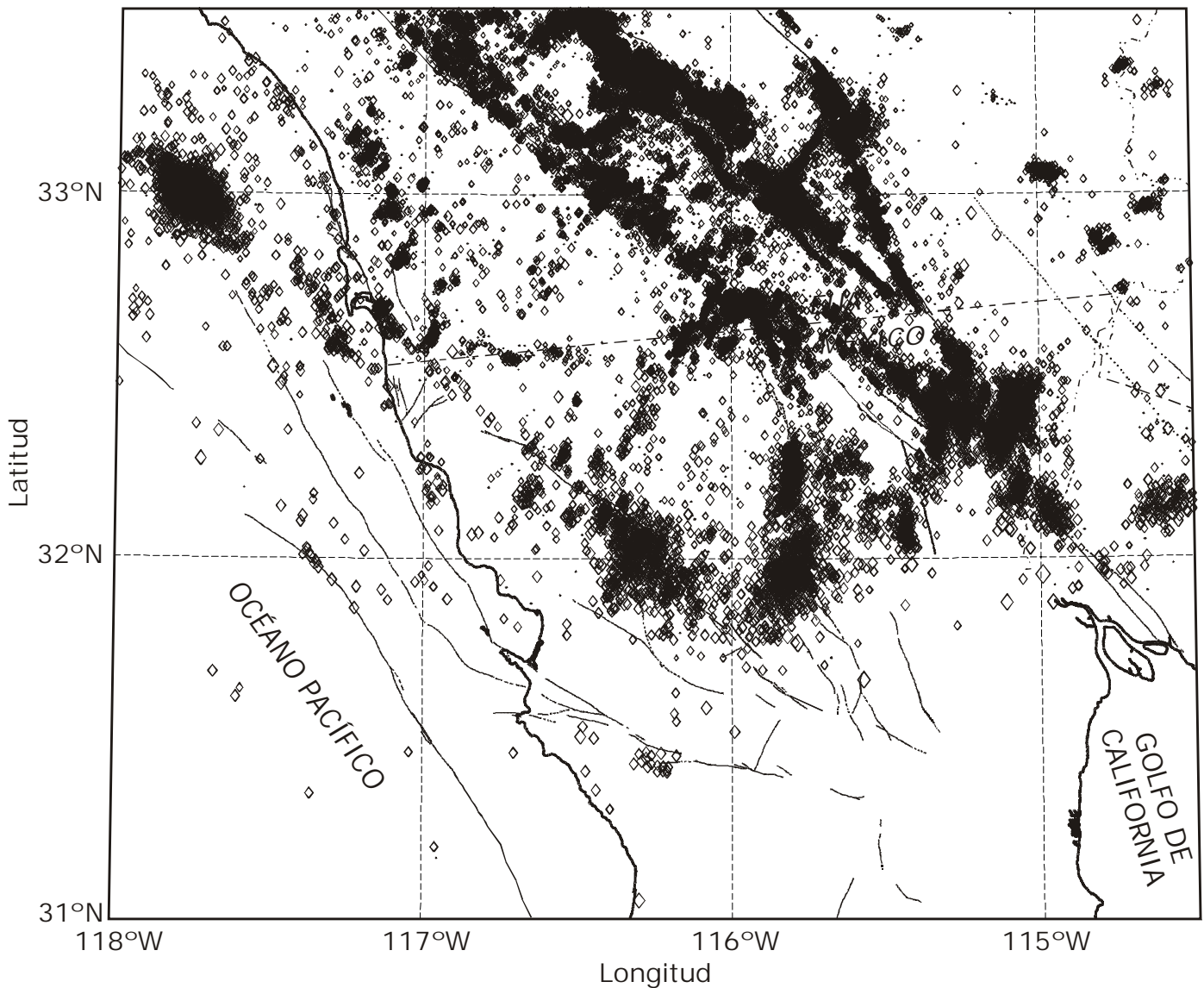


Figura 2. Mapa con los epicentros de la región 5, tomados del catálogo de Shearer *et al.* (2003). El mapa corresponde a la región fronteriza de ambas Californias. Las líneas gruesas representan la costa y el límite internacional; las líneas delgadas, fallas, el río Colorado y las depresiones del Salton y Laguna Salada. Ver, Frez y Frías-Camacho (1999) para el marco tectónico y nombres de sus características.

- site. Cada línea de los archivos de líneas de costas y de fallas contiene la latitud y la longitud de cada punto, consecutivamente; los segmentos discontinuos se separan con la línea (NaN, NaN).
- 4. Es fácil agregar otras características en el mapa, como señalización de sitios geográficos (use *plotm*), letreros (use *textm*), barras de error de la localizaciones catalogadas, etc.
- 5. El subcatálogo resultante se escribe en un archivo de texto, utilizando *fprintf*, el usuario puede introducir el formato adecuado a sus propósitos. Esta opción es más general que el comando *save* ya que es más portable y legible; esto último, aunque *save* sea usado con la opción *ASCII*.

- 6. *MATLAB* permite aplicar ventanas con código vectorizado. Por ejemplo, el programa (Figura 1) utiliza, en dos líneas inmediatamente después de llamar a *inpolygon*, al vector *in* (salida de *inpolygon*) para filtrar el catálogo de entrada y formar el de salida. Si se necesita el catálogo que no está dentro de la región limitada por el polígono, reemplazamos la línea *temp06=temp05(in>0.;*) por *temp06=temp05(~in>0.;*). Con un código vectorizado podemos también filtrar en magnitudes, profundidades u otras características. Por ejemplo, las dos líneas siguientes filtran en latitud y longitud, columnas 9 y 10 respectivamente del catálogo *temp05*; el subcatálogo resultante se guarda como *temp06*.

```
mwl=(temp05(:,8) < 31.5 & temp05(:,8) > 31.4 & temp05(:,9) > -116.35 & temp05(:,9) < -116.15);
```

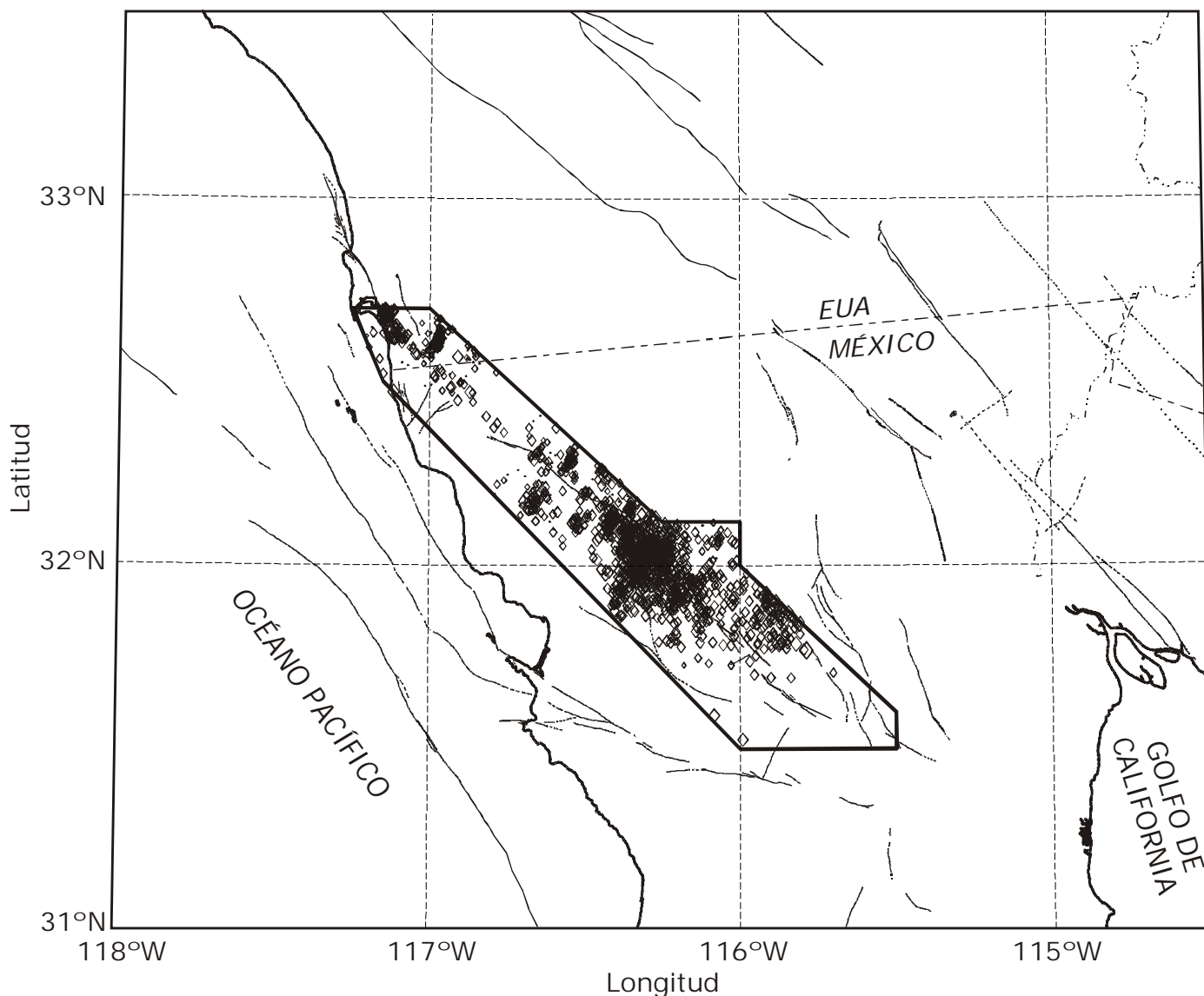


Figura 3. Resultado de aplicar una ventana poligonal al catálogo de hipocentros ilustrado en la Figura 2. Note que el polígono no es convexo.

temp06=temp05(mwI > 0,:);

7. La documentación de la función *axesm* debe ser estudiada con cuidado. Los parámetros modificados en el listado de la Figura 1 son los que nos parecen importantes destacar.
8. Las líneas de costas y del límite internacional fueron tomadas del sitio web de la National Geophysical Data Center (NGDC) de NOAA.
9. Puede que se tengan problemas de memoria con catálogos muy grandes. Además de aumentar la memoria virtual de la computadora, se puede leer línea a línea, aplicar una ventana y, si el evento está dentro de la ventana, guardar el resultado antes de pasar a la línea siguiente. Este procedimiento es costoso en tiempo pero usa muy poca memoria. Otra opción es reducir el catálogo a otro que tenga sólo los parámetros necesarios para

el propósito del trabajo, por ejemplo, sólo las coordenadas geográficas y la magnitud, si sólo ellas se necesitan para hacer un mapa.

10. La función *inpolygon* es general, es decir, discrimina eficientemente los puntos que están dentro, sobre y fuera de cualquier tipo de polígono, sea o no convexo; la documentación alienta al usuario a trabajar con polígonos con seis coordenadas aleatorias para los vértices, con lo cual resultan polígonos de todos tipos, incluso con lados que se cruzan. El lector interesado puede considerar también el algoritmo que aparece en el paquete ZMAP (Wiener, 2001) que es más simple, corto y con resultados indistinguibles a los que produce *inpolygon*, excepto para polígonos tan distorsionados que no son de uso práctico.



## APLICACIÓN

El programa anterior se aplica al catálogo de eventos sísmicos *temp05.cat*. Este corresponde a la región 5 (región fronteriza de ambas Californias), con 76545 relocalizaciones del catálogo del Sur de California entre 1984 y 2002, hechas por Shearer *et al.* (2003). La Figura 2 contiene todos los eventos localizados dentro del marco de la figura; la Figura 3 entrega un polígono y los epicentros ubicados en su interior. Las relocalizaciones están basadas en un algoritmo que identifica agrupamientos ("clusters") y usa, en un segundo paso, diferencias de tiempos de arribo. El mapa llama la atención por ciertas propiedades de la distribución de la sismicidad. En particular, la doble alineación que se observa a lo largo de algunas fallas en territorio de EEUU ha sido discutida por Shearer *et al.* (2002). Shearer *et al.* (2003) destaca la existencia de alineaciones que indican "un número sorprendente" de fallas conjugadas; vale la pena anotar que el carácter ortogonal de las alineaciones de las réplicas del sismo de Superstition Hill de 1987 ha sido tratado, entre otros, por Hudnut *et al.* (1989). También al respecto, Frez *et al.* (2004) han determinado que la dirección de fractura de varios sismos del norte de Baja California es perpendicular a la de las trazas cartografiadas de las fallas cercanas. Otra característica que llama la atención es la existencia de una agrupación de eventos cerca de la falla Agua Blanca, alrededor del punto (31.45; -116.27). Hemos utilizado nuestro programa para aislar el grupo, el que resulta estar compuesto de 17 sismos que ocurrieron entre 1984 y 1987. Ningunos de los dos catálogos regionales, el del Sur de California y del Noroeste de México, muestran esta aglomeración; ver, por ejemplo, Frez y Frías-Camacho (1999). Los tiempos de arribo correspondientes pueden obtenerse de las bases de datos de la Red del Noroeste de México (RESNOM) y del Southern California Earthquake Center (SCEC). Los tiempos de RESNOM no son compatibles con las localizaciones en la agrupación señalada. Una relocalización que utiliza datos de las dos redes regionales, *Hypo71* (Lee y Lahr, 1975) y la estructura de Nava y Brune (1982) coloca los 17 eventos en otro agrupamiento, esta vez ubicado cerca del punto (31.7 - 115.95).

La Figura 2 maneja 76545 eventos con 23 columnas. Con una computadora personal Pentium 4, 256 Mb de memoria de trabajo y una memoria virtual sin un máximo declarado, se necesitan alrededor de dos minutos para formar la Figura 1 y algunos segundos para la Figura 3. Ello se explica porque la mayor parte del tiempo de ejecución se gasta en escalar los símbolos de los epicentros tomando en cuenta la magnitud.

## CONCLUSIONES

Entregamos un programa que selecciona sismos cuyos epicentros están dentro de un polígono cuyos vértices están marcados por coordenadas geográficas en la superficie de la Tierra; además, hace el mapa que incluye los sismos seleccionados. El programa escrito en *MATLAB*, es compacto, fácil de comprender y debe ayudar a escribir programas semejantes. La ventaja con respecto a utilizar un buen paquete como ZMAP (Wiener, 2001), con más de 50000 líneas de código, es obvia. Un programa relativamente pequeño y

con control personal es simplemente más fácil manejar; entre otras cosas, puede fácilmente ser a) puesto al día con respecto a las versiones de *MATLAB*, b) ampliado y modificado para propósitos específicos de la investigación del usuario y c) transportable. Dada la popularidad que ha alcanzado *MATLAB* en los centros de educación superior, creo que su acceso no es una limitación para estudiantes de Ciencias de la Tierra. El caso de aplicación que entregamos demuestra que se pueden manipular catálogos relativamente grandes con computadoras modernas. Si ello no es así por limitaciones de la memoria de trabajo de la computadora, además de un aumento de la memoria virtual, la alternativa es reducir el tamaño del catálogo por eliminación de columnas que no se utilizan; otra es modificar el programa de tal modo que realice el trabajo, incluyendo leer y guardar resultados, para cada línea del catálogo. De este modo, no se necesita mantener todo el catálogo en memoria.

Hemos aprovechado el caso de aplicación para discutir brevemente algunas características de la sismicidad que se muestra en la Figura 2. Una, no comentada en la literatura, es la existencia de un agrupamiento o nido de sismos en la falla Agua Blanca, que se puede fácilmente identificar con nuestro programa. La relocalización de los hipocentros con los tiempos de arribo de RESNOM indica que es un artefacto del método, probablemente producto de la lejanía de las estaciones receptoras de la Red del Sur de California y el agrupamiento, que es real. Concluimos que la falla Agua Blanca no tiene una clara sismicidad regional asociada ( $M_L > 2.1$ ); tal como se ha señalado, por ejemplo, en Frez y Frías-Camacho (1998).

## AGRADECIMIENTOS.

Agradezco al Dr. P. Shearer por darme acceso a su catálogo; igualmente, a RESNOM, SCEC y la Red del Sur de California por utilizar sus bases de datos. Los comentarios en el arbitraje nos fueron de utilidad para mejorar la presentación del artículo.

## REFERENCIAS

- Frez, J. y V.M. Frías-Camacho, 1998. Mapas anuales de sismicidad para la región fronteriza de ambas Californias, *GEOS*, 18, 112-130.
- Frez, J., F.A. Nava and J. Acosta, 2004. Source rupture plane determination from Doppler effect for small earthquakes recorded by local networks, *Bulletin of the Seismological Society of America* (sometido).
- Hudnut, K.W., L. Seeber and J. Pacheco, 1989. Cross-fault triggering in the November 1987 Superstition Hills earthquake sequence, *Southern California, Geophysical Research Letters*, 16, 199-202.
- Lee, W.K.H. and S.W. Stewart, 1981. *Principles and Applications of Microearthquake Networks*, Academic Press, Londres, Inglaterra.
- Lee, W.K.H. and J.C. Lahr, 1975. *HYP071* (revised): A computer program for determining hypocenter, magnitude, and first motion pattern of local earthquakes, U.S. Geological Survey, Open-File Rep. (U.S.) 75-311.

- Nava, A. and J.N. Brune, 1982. An earthquake-explosion reversed refraction line in the Peninsular Ranges of Southern California and Baja California, *Bulletin of the Seismological Society of America*, 72, 1195-1206.
- Pearson, F., 1990. *Map Projections: Theory and Applications*, CRC Press, Florida, E.U.A.
- Sedgewick, R., 1992. *Algorithms in C++*, Addison-Wesley, Reading, Massachusetts, E.U.A.
- Sedgewick, R., 1995. *Algoritmos en C++*, Addison-Wesley/Días de Santos, Wilmington, Delaware, E.U.A.
- Shearer, P.M., 2002. Parallel fault strands at 9-km depth resolved on the Imperial fault, Southern California, *Geophysical Research Letters*, 29, doi: 10.129.2002/2002GL015302.
- Shearer, P., E. Hauksson, G. Lin and D. Kilb, 2003. Comprehensive waveform cross-correlation of southern California seismograms: Part 2. Events locations obtained using cluster analysis, *EOS Trans. AGU*, 84(46), Fall Meeting. Suppl., Abstract S21D-0326.
- Young, T.L. and M.L. Van Woert, 1992. *Plot88, Software Library Reference Manual*, Plotworks Inc., Ramona, California, E.U.A.
- Wiener, S., 2001. A software package to analyze seismicity: ZMAP, *Seismological Research Letters*, 72, 273-382.

Recepción del manuscrito: 12 de febrero, 2004

Recepción del manuscrito corregido: 25 de junio, 2004

Aceptación del manuscrito: 10 de agosto, 2004