**SEISPLOT: UN CÓDIGO INTERACTIVO PARA MOSTRAR Y SELECCIONAR LAS PRIMERAS LLEGADAS EN SÍSMICA DE GRAN ÁNGULO**

Mariano S. Arnaiz-Rodríguez1, Maximiliano Bezada2, Michael Schmitz3

*1Departmento de Geofísica, Facultad de Ingeniería, Escuela de Geología, Minas y Geofísica, Universidad Central de Venezuela, Venezuela (*[*marianoarnaiz@gmail.com*](mailto:marianoarnaiz@gmail.com)*, 0058212-4819006).*

*2 Department of Earth Sciences, University of Minnesota.*

*3Fundación de Investigaciones Simológicas (FUNVISIS), Caracas, Venezuela.*

[\*marianoarnaiz@gmail.com](mailto:*marianoarnaiz@gmail.com)

|  |
| --- |
| RESUMEN |

Utilizando funciones de acceso abierto, así como las poderosas capacidades gráficas de Matlab, desarrollamos un código rápido, eficiente y flexible que permite al usuario el fácil manejo de dato de sísmica de gran ángulo. El código guía al usuario a través de varios pasos en una interfaz humano-maquina (HMI); generará una imagen de alta resolución del dato, así como un conjunto de archivos que contienen las llegadas seleccionadas. El código diseñado es eficiente, rápido y flexible. Contempla la permanente actualización para optimizarlo al añadirle otras características que ayuden al usuario a hacer la selección de las llegadas de manera rápida, amigable e interactiva, dentro de un ambiente que no se encuentre limitado por el tipo de sistema operativo, ni que quiera ser compilado. El código se desarrolla en el marco del proyecto GIAME (Geociencia Integral de los Andes de Mérida).

*Palabras Clave: sísmica de refracción, sismología, Matlab, Código libre*

|  |
| --- |
| ABSTRACT |

Using pre-coded functions available online, as well as Matlab’s powerfull graphical capabilities and functions, we developed a fast, efficient and flexible code that allows the user to easily handle wide-angle seismic data. The code guides the user through several steps in a HMI that will eventually generate a high-resolution image of the data, as well as several files with the hand picked phases. The designed code envisages the permanent and further update to optimize it by adding other features that help the user to select the arrivals in a fast, friendly and interactive manner, within an environment not limited by the type of operating systems. Also, it does not require to be compiled every time it needs to be installed in a different computer. The code is developed within the framework of GIAME project (Geociencia Integral de los Andes de Mérida).

*Keywords: seismic refraction, seismology, Open Source*

|  |
| --- |
| INTRODUCCIÓN |

La sísmica de refracción profunda es una de las técnicas primordiales para el estudio de las características de la corteza de la Tierra. En Venezuela, esta técnica ha sido utilizada a través de décadas con este fin (e.g. Castejón et al., 1986; Schmitz et al., 2002; Schmitz et al., 2008). En general, ha sido exitosa en cumplir los objetivos científicos en relación con el desarrollo de estas investigaciones. Sin embargo, para lograrlo siempre se ha dependido de programas y códigos que han sido desarrollados fuera del país. Inclusive, se han empleado otros desarrollados para otros fines. En tal sentido, para solventar esta dependencia, en el marco del proyecto **GIAME (Geociencia Integral de los Andes de Mérida**), nos hemos propuesto crear el código **SEISPLOT:** un código interactivo para mostrar y seleccionar las primeras llegadas en sísmica de gran ángulo.

Las primeras etapas del código consisten en el filtrado de los datos, presentación de la sección sísmica adquirida y la selección de los tiempos de llegada de las ondas corpóreas a la superficie. Son estas etapas que pueden representar las más críticas de la investigación. Con el propósito de facilitar estas etapas, el objetivo de este trabajo ha sido el desarrollo de un código flexible y rápido, que consta de una interfaz sencilla y ligera en Matlab 2012b.

El código se encuentra diseñado exclusivamente para el manejo de datos de refracción de gran ángulo; permite realizar el preprocesamiento y procesamiento básico del dato, así como distintos tipos de graficaciones y un módulo completo de selección de tiempos de llegadas. Adicionalmente, permite la exportación de las secciones en un formato de alta resolución (Postscript, .ps) y de los tiempos selecconados para la llegada de las ondas sísmicas.

|  |
| --- |
| METODOLOGÍA |

Para generar el código, primero fue necesario recolectar una serie de funciones desarrolladas por otros grupos de investigación que facilitarían su programación. La mayor parte de estas funciones (*GetSegyHeader.m; GetSegyHeaderBasics.m; GetSegyTraceData.m; GetSegyTraceHeader.m; isoctave.m; ReadSegy.m; SegymatVerbose.m*) son utilizadas exclusivamente para la lectura del formato binario en el que se encuentran originalmente los datos (.SEGY). Estas funciones fueron desplegadas como parte del proyecto SEGyMAT a cargo de Thomas Mejer Hansen. Una función adicional utilizada es la función de entrada gráfica (ginput2.m), que permite hacer la selección de los tiempos de llegada de forma interactiva (Carlos Vargas, 2009).

Al generar la interfaz humano-maquina (HMI), se planteó en forma de ventanas con botones y casillas (Fig 1). Estas tienen la ventaja de ocupar poca memoria en la computadora, mientras esperan la respuesta del usuario; además, liberan la memoria utilizada, una vez que han sido descartadas, y proveen una secuencia lógica para las decisiones que debe de tomar el usuario mientras utiliza el código.

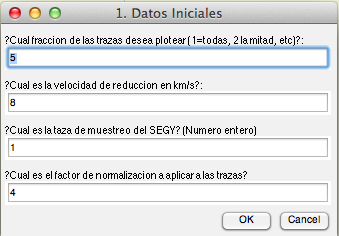


Fig 1. Ejemplos del sistema de ventanas pop-ups seleccionada la HMI.

El código requiere como entrada un archivo con la geometría de los receptores: sus seriales, coordenadas (las cuales pueden encontrarse en grados o en un sistema de proyección plana), las coordenadas del disparo y la elevación de los receptores. Adicionalmente, se requiere que los datos de cada receptor (en formato SEGY) se encuentren guardados en una carpeta (e.g. DATA.txt).

En primer lugar, el código solicita que se le especifique el tipo de coordenadas utilizadas en el archivo de geometría, la cantidad de canales a ser considerados (todos, la mitad, un tercio, etc), la velocidad de reducción a utilizar para la presentación de la sección, la tasa de muestreo sobre el archivo SEGY (puesto que trabajar con todos los datos puede ser pesado desde el punto de vista computacional), y el factor de normalización de amplitud de las trazas.

El segundo paso requiere el diseño de una filtro para el dato. Este filtro será aplicado a toda la sección y se puede seleccionar entre un filtro Butterworth, un filtro Gaussiano, o un filtro pasabanda de respuesta infinita (FIR). De estos filtros se pueden especificar todos los parámetros, de forma sencilla, a través de la HMI diseñada.

En segundo lugar, el código solicitará al usuario que seleccione un tipo de representación gráfica: Wiggle, Wiggle+Área variable, Área variable positiva o Área variable negativa. Las Fig. 2 y Fig 3. Presentan ejemplos de la diferencias entre una sección en Wiggle y otra en Wiggle+ Área variable, respectivamente.

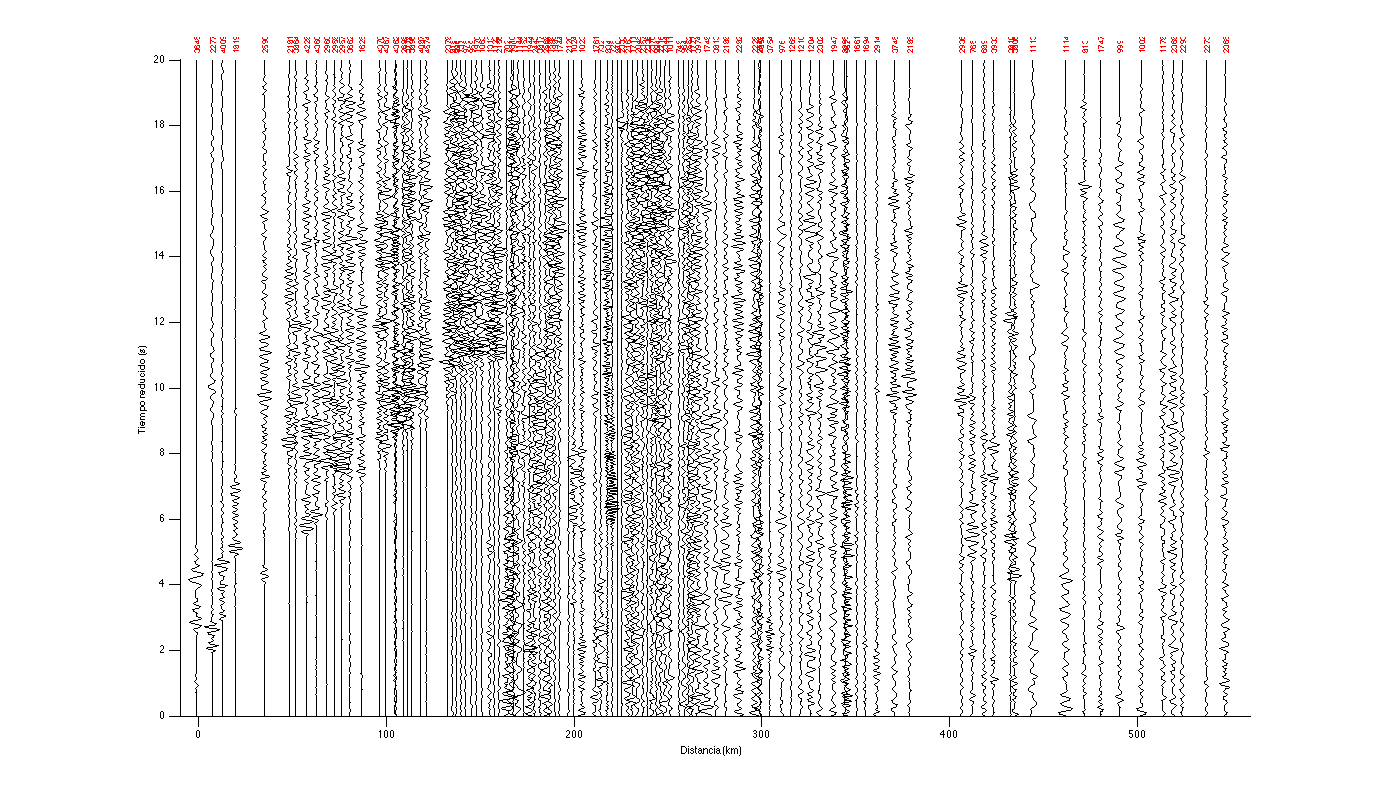


Fig 2. Ejemplo de una sección sísmica en presentación tipo Wiggle

En tercer lugar, el código iniciará el proceso de leer todos los archivos SEGY y mostrarlos en una figura (Plot) de Matlab utilizando un ciclo for. A medida que los archivos son leídos, el dato es preprocesado (se remueven la media, una tendencia lineal y un filtro tipo tapper se aplicado al inicio y final de la señal), la señal es filtrada según los parámetros previamente especificados, es normalizada y es colocada de acuerdo a su distancia con respecto al punto de disparo (el cuál siempre se localiza en cero y los receptores que se localizan hacia el norte son graficados en distancias negativas mientras que los receptores hacia el sur son graficados con distancias positivas), y la velocidad de reducción es aplicada. Cuando el ciclo ha concluido, se le presenta una sección al usuario y la posibilidad de eliminar canales ruidosos y canales sin información de forma interactiva.

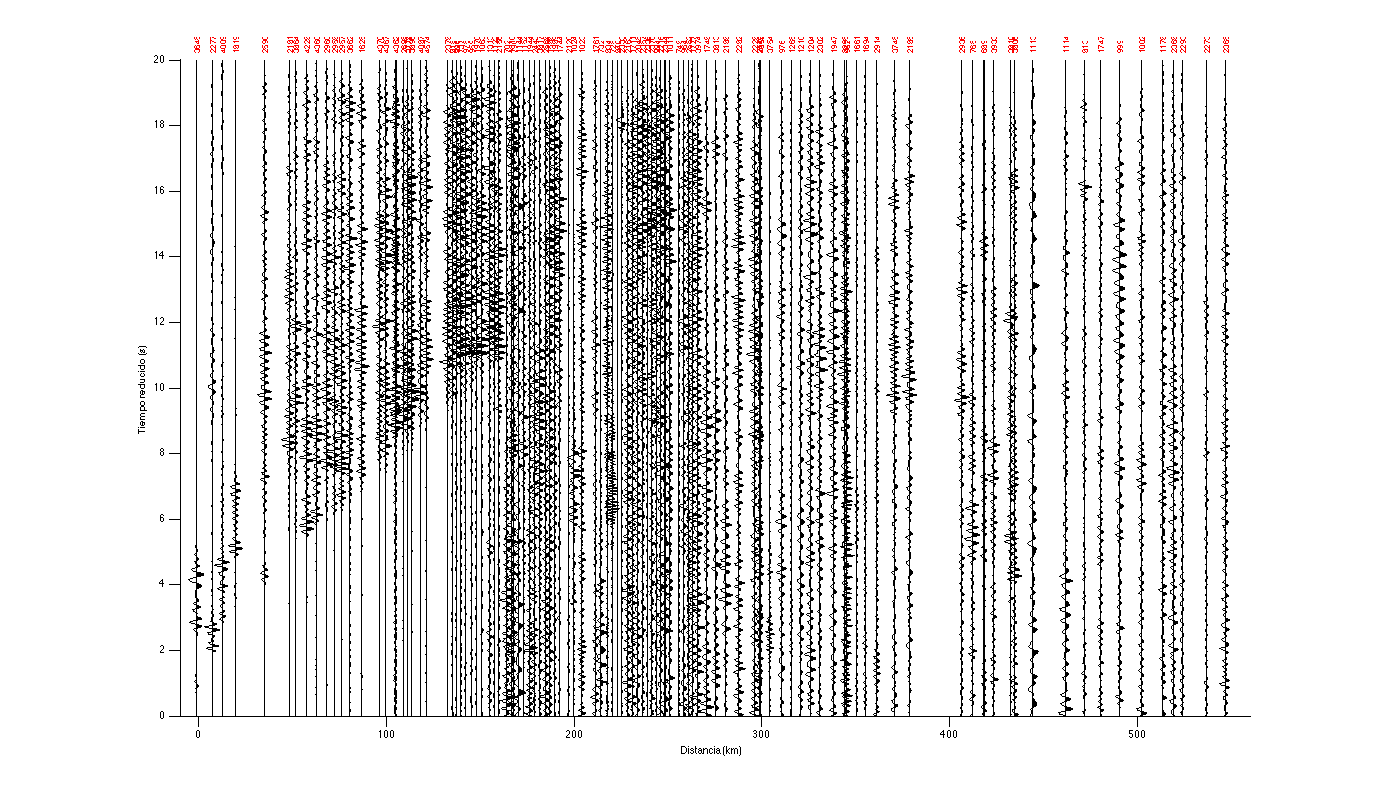


Fig 3. Ejemplo de una sección sísmica en presentación tipo Wiggle + Área variable

En cuarto lugar, el código iniciará la rutina de selección de fases, que permite la selección de las primeras llegadas para Pg, PmP, Pn, y Pi (a pesar de que el modulo puede ser utilizado de forma libre por el usuario). Utilizar esta rutina es simple: click izquiedo se utiliza para hacer zoom in, doble click para hacer zoom out, click derecho seleccionar el tiempo de llegada y marcarlo en la figura, Backspace o Delete se utilizan para borrar los puntos anteriores (todos los puntos de la fase actual pueden borrarse si es deseado), y presionar la tecla Enter indica que la selección actual ha terminado. Al salir de la rutina, el código automáticamente escribe un archivo de texto con el nombre de la fase seleccionada. Este archivo contiene la información de la distancia y los tiempos (absoluto y reducido) de las llegadas seleccionadas.

La rutina consta de dos secciones, la primera es para la graficación de líneas guías (líneas que se utilizan exclusivamente para guiar al intérprete durante la interpretación) y la selección de los tiempos de llegada. La rutina corre un número infinito de veces y da la opción de cargar los puntos hechos con anterioridad. Cuando todas las fases han sido elegidas, el código muestra la figura con las selecciones de manera de que pueda ser editada una última vez, de forma interactiva, antes de ser exportar como un archivo PostScript (.ps). Un ejemplo de una sección interpretada preliminarmente se presenta en la Fig 4.

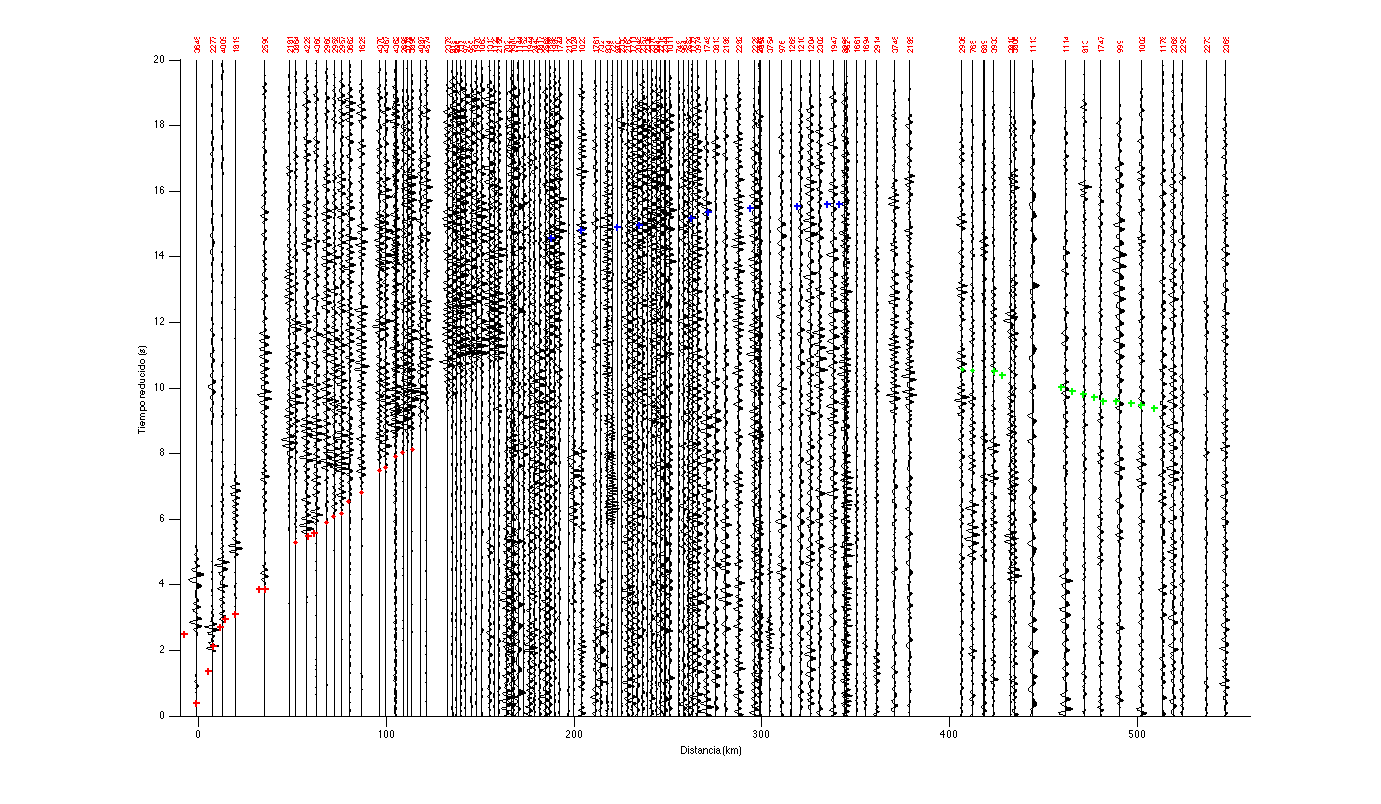


Fig 3. Ejemplo de una sección sísmica preprocesada, filtrada e interpretada en el ambiente Seisplot 4.1.1

|  |
| --- |
| RESULTADOS Y DISCUSIÓN |

En general, el código diseñado es eficiente, rápido y flexible. Esperamos poderlo mejorar y añadirle otras características que ayuden al usuario a hacer la selección de las llegadas de manera rápida, amigable e interactiva en un ambiente que no se encuentra limitado por el tipo de sistema operativo, ni que requiera ser compilado. El código ha demostrado operar correctamente en distintos entornos y sistemas operativos (Linux Mint 17, Ubuntu 14.04, Mac OSX 10.8 y 10.9.4, Windows Xp, Windows 7 y Windows 8) y bajo todas las versiones de Matlab superiores a las 2012a.

Con respecto a la velocidad de computo, una computadora con un procesador Intel 1.7 Gz core i7 y 8GB de memoria RAM (1600 MHz DDR3) es capaz de manejar sin problemas una gran cantidad de datos, correspondientes a 534 canales sísmicos con un total de 46 mb de tamaño, procesando cada archivo de 84 kb en menos de un segundo. Resultados similares se pueden obtener con una computadora con un procesador Intel 1.3 Gz core i3 y 4 GB de memoria RAM (1600 MHz DDR3). Computadoras, con procesadores de dos núcleos de generaciones anteriores y memorias de 1 y 2 GB, pueden procesar los datos sin ninguna dificultad, pero carecen de suficiente capacidad para manejar las secciones que el código genera, y no son recomendadas su utilización. Igualmente, el código ha sido probado en paralelo con otras aplicaciones y, en general, no presenta conflictos con aplicaciones de este tipo.

|  |
| --- |
| **CONCLUSIONES** |

En conclusión, hemos sido capaces de desarrollar un código eficiente, rápido y flexible que permite preprocesar, procesar, representar gráficamente e interpretar los datos de refracción profunda. El código desarrollado, en su versión actual (Seisplo 4.1.1), no presenta limitantes con respecto al sistema operativo seleccionado para correrlo. Es eficiente para manejar grandes cantidades de datos con ordenadores de nueva generación, aunque puede ser utilizado en computadoras de generaciones previas sin más limitantes que las impuestas por la versión de Matlab utilizada.

|  |
| --- |
| REFERENCIAS |

Castejón, B., Marquez, C., Urbáez, M. (1986). *Modelo de corteza en la Costa Oriental del Lago de Maracaibo*. Trabajo especial de grado. Inédito. Universidad Central de Venezuela, Caracas, 208 pp.

Hansen, T.M. (2011). SEGyMAT Project. <http://segymat.sourceforge.net> (Consulta, Septiembre 2014)

Schmitz, M., Bezada, M., Avila, J., Vieira, E., Yánez, M., Levander, A., Zelt, C.A., Magnani, M.B., Jácome, M.I. and the BOLIVAR active seismic working group (2008). Crustal thickness variations in Venezuela from deep seismic observations. *Tectonophysics*, doi: 10.1016/j.tecto.2007.11.072.

Schmitz, M., Chalbaud, D., Castillo, J., Izarra, C. (2002). The Crustal Structure of the Guayana Shield, Venezuela, from seismic refraction and gravity data. *Tectonophysics,* Vol. 345 (1-4), 103-118.

Vargas, C., (2009). Ginput 2. <http://www.mathworks.com/matlabcentral/fileexchange/20645-ginput2-m-v3-1--nov-2009-> (Consulta, Septiembre 2014)