

Desarrollo de un software libre para procesamiento de datos sísmicos 2D para
prospección de pozos de petróleo

Juan Carlos Galeano Estrella

Universidad Autónoma de Bucaramanga

Maestría en Software Libre

Bucaramanga

2018

Desarrollo de un software libre para procesamiento de datos sísmicos 2D para
prospección de pozos de petróleo

Presentado por

Juan Carlos Galeano Estrella

Trabajo de grado para optar por el título de Magíster en Software Libre

Director

Sebastián Roa Prada, PhD

Universidad Autónoma de Bucaramanga

Maestría en Software Libre

Bucaramanga

2018

RESUMEN

Hay momentos en que el precio del mercado de petróleo no cubre los costos de extracción, lo que hace necesario reducir o incluso detener la producción de crudo en una gran cantidad de campos petrolíferos en todo el mundo. Esta reducción en la producción resulta en una menor inversión en actividades de exploración. Por lo tanto, es necesario optimizar los costos, reducir los gastos y disminuir la incertidumbre en las actividades de exploración, especialmente cuando los recursos para la exploración son escasos debido a los bajos precios del petróleo. De acuerdo a estas necesidades se planteó el objetivo de crear una aplicación bajo licencia libre, que utilizara un algoritmo con la solución a la migración por medio de la ecuación de Kirchhoff, partiendo de la lectura de un archivo binario en formato estándar SEG-Y, mostrando resultados por medio de una gráfica antes y después de aplicar el proceso de migración, contando con una interfaz gráfica sencilla y un manual adecuado, y como aspecto final se realizó la evaluación de la aplicación por medio de un análisis a modo de benchmarking con otra aplicaciones de similares características.

La investigación realizada consistió en el desarrollo del algoritmo de migración, para lo cual se evaluaron las características más relevantes en el tema; se observaron todas las etapas del proceso realizado a los datos sísmicos desde su toma en el sitio hasta la entrega de imágenes sísmicas, en donde finalmente se desarrolló una aplicación compuesta por varias bibliotecas, cuyo principal objetivo es llevar a cabo la migración sísmica post-apilada por el método de Kirchhoff. Esta pieza de software presenta la funcionalidad para leer y exportar archivos escritos en el formato binario estándar SEG-Y, y para generar imágenes sísmicas. La aplicación tiene un manual de usuario e interfaz gráfica para facilitar su uso.

TABLA DE CONTENIDO

Glosario.....	1
Introducción.....	4
1. Revisión bibliográfica.....	6
1.1 Migración sísmica.....	6
1.2 Migración post apilado.....	6
1.3 Desarrollo paralelo.....	7
1.4 Dispositivos programables.....	7
1.5 Software utilizado en el modelado sísmico.....	8
1.6 Software libre para migración sísmica.....	8
2. Marco Teórico.....	10
2.1 Adquisición de datos sísmicos.....	10
2.2 Análisis de datos sísmicos.....	11
2.3 Migración por diferencias finitas.....	18
2.4 Migración omega-x.....	18
2.5 Migración cambio de fase.....	19
2.6 Migración de kirchhoff.....	20
2.7 Modelado.....	24
2.8 Software para migración sísmica.....	25
2.9 Herramientas libres para desarrollo.....	25
2.10 Archivos de datos sísmicos.....	26
3. Método de investigación.....	27
3.1 Desarrollo de una librería en java para lectura y escritura de archivos (seg y).....	27
3.2 Desarrollo e implementación del algoritmo para la migración con la ecuación de kirchhoff 2d.....	31
3.3 Desarrollo de la librería para generar imágenes.....	34
3.4 Desarrollo de la interfaz gráfica para el usuario.....	35
3.5 Prueba y corrección de fallos.....	35
3.6 Metodología para el desarrollo del software.....	36

3.7	Desarrollo del manual de usuario.....	36
3.8	Estudio de benchmarking con respecto a los otros software libres existentes en el mercado.....	37
4.	Resultados de la investigación.....	38
4.1	Software Desarrollado para migración sísmica.....	38
4.2	Documentación e interfaz gráfica.....	40
4.3	Análisis a modo de Benchmarking.....	42
4.4	Validación del software.....	47
5.	Conclusiones.....	51
6.	Recomendaciones y trabajos futuros.....	53
	Bibliografía.....	54
	Anexos.....	57

LISTA DE ILUSTRACIONES

Ilustración 1. Angulo y dirección de buzamiento.....	1
Ilustración 2. Comportamiento de las ondas sísmicas en una interface horizontal entre dos distintos medios litológicos.....	10
Ilustración 3. Modelos sísmicos, a) 2D, b) 3D.....	11
Ilustración 4. Deconvolución de una respuesta ideal.....	12
Ilustración 5. Figura de trazas sucesivas. (a) Reflectividad, (b) Reflectividad con límite de banda, (c) MED y (d) mínima entropía con constantes de frecuencia de dominio....	14
Ilustración 6. Diagrama de una traza.....	15
Ilustración 7. Apilado de trazas para un mismo CMP.....	16
Ilustración 8. Datos después del apilado, a) Antes de la migración b) Después de la migración.....	16
Ilustración 9. (a) Sección geológica, (b) Migración del segmento C'D' a su verdadera posición CD en la sección en tiempo.....	17
Ilustración 10. Ejemplo de las etapas iniciales de la migración por cambio de fase, a una velocidad constante y una velocidad $V(z)$	19
Ilustración 11. Velocidad media constante, Trayectoria Kirchhoff en tiempo de viaje....	22
Ilustración 12. Comparación de los resultados de la migración antes (izquierda) y después (derecha).....	23
Ilustración 13. Ejemplo de modelado 2D.....	24
Ilustración 14. Estructura de la secuencia de bytes de un archivo SEG-Y con N registros de encabezado de archivo de texto extendido y M registros de trazas.....	27
Ilustración 15. Formato <i>big-endian</i>	28
Ilustración 16. Algoritmo para la conversión de datos binarios a naturales.....	30
Ilustración 17. Comparativo de imágenes luego de la migración con diferentes velocidades.....	32
Ilustración 18. Algoritmo para la migración.....	33
Ilustración 19. A la izquierda la imagen antes y a la derecha después de la migración.	39
Ilustración 20. A la izquierda la imagen antes y a la derecha después de la migración.	40

Ilustración 21. Interfaz gráfica del software.....	41
Ilustración 22. Arriba los resultados antes y después con el algoritmo desarrollado, abajo los resultados antes y después de la migración con CWP/SU.....	48
Ilustración 23. Arriba los resultados antes y después con el algoritmo desarrollado, abajo los resultados antes y después de la migración con CWP/SU.....	49

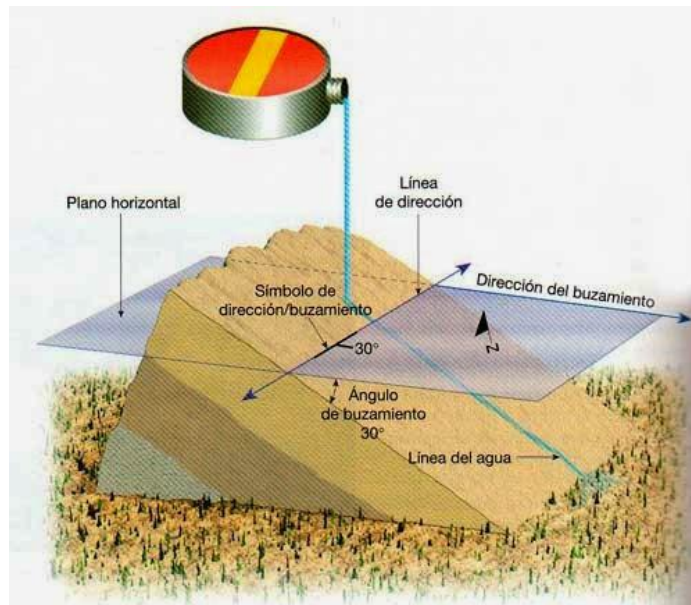
LISTA DE TABLAS

Tabla 1. Software libre para modelamiento sísmico.....	9
Tabla 2. Velocidad normalizada y salida b deseada.....	13
Tabla 3. Formato para conversión de valores 32bits.....	29
Tabla 4. Comparación de sistema operativo.....	42
Tabla 5. Comparación de la instalación del software.....	43
Tabla 6. Comparación de la lectura de archivos.....	44
Tabla 7. Comparación de la escritura de archivos.....	44
Tabla 8. Comparación del proceso de migración.....	45
Tabla 9. Comparación de la ilustración de sismogramas.....	45
Tabla 10. Comparación de interfaz gráfica.....	46
Tabla 11. Comparación del manual de usuario.....	46
Tabla 12. Comparación del tipo de licencia.....	47

GLOSARIO

ANGULO DE BUZAMIENTO: ángulo que forma la línea de máxima pendiente de una superficie (estrato, capa, filón o falla) con su proyección sobre el plano horizontal. (Real academia de ciencias exactas, físicas y naturales, 2016)

Ilustración 1. Angulo y dirección de buzamiento.



Fuente: (Olano & Suarez, 2014)

APILADO: consiste en la técnica de sumar varias trazas con aspectos en común con el fin de mejorar la señal y reducir el ruido.

AZIMUT: ángulo que forma el plano vertical que contiene una dirección dada con el meridiano local, contado en el plano del horizonte en sentido de las agujas del reloj. Como origen se toma, en unos casos, la dirección sur y en otros, la norte. (Real academia de ciencias exactas, físicas y naturales, 2016)

COMPUTACIÓN PARALELA: “es una forma de cómputo en la que muchas instrucciones se ejecutan simultáneamente, operando sobre el principio de que

problemas grandes, a menudo se pueden dividir en unos más pequeños, que luego son resueltos simultáneamente (en paralelo).” (Fundación Wikimedia, Inc., 2016)

DECONVOLUCIÓN: “un paso del procesamiento de las señales sísmicas para recuperar las altas frecuencias, atenuar las múltiples, ecualizar las amplitudes, producir una ondícula de fase cero, o para otros fines que por lo general afectan la forma de onda. La deconvolución, o filtrado inverso, puede mejorar los datos sísmicos afectados adversamente por el filtrado, o la convolución que se produce naturalmente a medida que la energía sísmica es filtrada por la Tierra. La deconvolución también puede efectuarse en otros tipos de datos, tales como datos gravimétricos, magneto-métricos o de registros de pozos.” (Schlumberger Limited, 2006).

PLIEGUE O FOLD: “estructura de una roca o conjunto pétreo que siendo una superficie de referencia definida como plana antes de la deformación se transforma en una superficie curvada o doblada. Salvo casos especiales, implica un acortamiento del espacio ocupado originalmente.” (Real academia de ciencias exactas, físicas y naturales, 2016).

MIGRACIÓN: “la migración consiste en reubicar elementos de datos para ajustar posiciones de los reflectores asociados o puntos donde las ondas sísmicas se dispersan o curvan a consecuencia de un obstáculo (puntos de difracción). Generalmente se basa en la premisa de que todos los elementos de datos representan ya sea reflexiones primarias o difracciones. La migración requiere el conocimiento de la distribución de la velocidad; los cambios laterales en la velocidad desvían las trayectorias de los rayos y por lo tanto afectan la migración, que da como resultado una sección o línea sísmica migrada de forma irreal.” (Vásquez, 2010).

OFFSET: es una pequeña distancia que se añade a los datos obtenidos en campo con el fin de corregir el desplazamiento de la fuente sísmica.

PROSPECCIÓN: “método de investigación del subsuelo mediante el estudio de la propagación de ondas elásticas provocadas artificialmente.” (Real academia de ciencias exactas, físicas y naturales, 2016)

TOMOGRAFÍA: es el procedimiento por medio del cual se procesan imágenes partiendo de secciones más pequeñas de la misma.

INTRODUCCIÓN

La variación de precio del petróleo trae momentos en los que el bajo costo del barril obliga a reducir la producción Colombiana del crudo debido a que no se justifican los costos de extracción. Esto como consecuencia de que existen pocos pozos y no son de gran tamaño. Dicha reducción también afecta la inversión realizada a la exploración, pero esto es algo que no debería ocurrir, pues es necesario optimizar los recursos proporcionados para la exploración de forma que se reduzcan los costos en la obtención de datos y se realicen nuevos e importantes hallazgos. También es importante tener en cuenta que una gran parte del territorio colombiano no ha sido explorado. Adicionalmente se puede mencionar que el análisis de los datos del territorio que ha sido explorado se realiza por medio de software comercial, que no sólo es relativamente costoso, sino que cuenta con limitantes en la manejabilidad. Según la Agencia Nacional de Hidrocarburos, ANH, los paquetes de software más comúnmente usados en Colombia son Omni, Mesa y DirectAid, los cuales corresponden a software de modelado sísmico y tienen en cuenta en menor medida la migración o la inversión sísmica, la cual es necesaria para obtener datos con buenos resultados, y aplicables a la prospección de hidrocarburos. Otro software utilizado es EcoElast2D, desarrollado por la empresa Numérica Ltda, en Bucaramanga. Este software permite el modelamiento de la propagación de ondas elásticas 2D en cuerpos sólidos y utiliza una aproximación al modelado de la propagación de ondas en medios sólidos por medio de diferencias finitas, sin embargo no han resuelto el problema inverso, lo que reduce su aplicabilidad en la exploración de hidrocarburos.

Actualmente el desarrollo de software para buscar soluciones tecnológicas se encuentra muy avanzado en diferentes campos de estudio, pero su aplicación a la exploración de petróleo se observa limitada a unas pocas alternativas y a un gran costo. Situación que puede mejorar por medio del desarrollo de aplicaciones basadas en software libre, que permitan reducir costos al mismo tiempo que facilitan el desarrollo

colaborativo, la multiplataforma, simplicidad en el proceso de ejecución y mayor aprovechamiento del hardware disponible.

De lo observado se desarrolló un software libre para prospección de petróleo, que realiza la migración sísmica, y que además puede ser mejorado por quien así lo vea necesario. Cabe mencionar que con este proyecto se pretende también dar un paso inicial y cerrar una brecha en busca de incentivar el desarrollo de software libre en la Universidad Autónoma de Bucaramanga, con base en la necesidad de software que realice análisis sísmico con fines de exploración de hidrocarburos, ubicación de fallas geológicas o cualquier otro fin.

En el contenido del presente documento se encuentran los procesos seguidos durante el desarrollo de la investigación, partiendo de la revisión bibliográfica sobre los últimos desarrollos en migración sísmica post-apilada y software utilizado actualmente en el análisis de datos sísmicos. Los conceptos necesarios para comprender el proceso y realizar un buen análisis sísmico se encuentran en el marco teórico, partiendo desde la adquisición de datos y el almacenamiento de los mismos, hasta llegar al modelado del sismograma. Con estos conceptos se desarrolló la metodología de investigación por medio de características especiales o requisitos específicos como lo son, el desarrollo de las librerías de lectura, escritura, migración y la interfaz gráfica, de igual forma se expone la metodología utilizada y el método en que se realizaron las pruebas. Luego del proceso investigativo se exponen los resultados obtenidos, por un lado se encuentra el software desarrollado, incluyendo su documentación y por el otro lado un análisis a modo de benchmarking con el fin de validar los resultados. Finalmente se cierra la investigación con las conclusiones y proponiendo su continuación al completar las funcionalidades requeridas para el análisis de datos sísmicos con recomendaciones y trabajos futuros hasta llegar a obtener un paquete de software muy completo.

1. REVISIÓN BIBLIOGRÁFICA

1.1 MIGRACIÓN SÍSMICA

Actualmente existe gran interés en el análisis de datos sísmicos y geológicos debido a su importancia en el campo de la exploración de petróleo, además existen diferentes condiciones de terreno para el procesamiento de datos hasta llegar a una imagen en condiciones óptimas que presenten poco o nada de ruido. Por tal motivo existen diferentes métodos y procesos para el tratamiento de los datos; el proceso de migración es muy necesario al momento de corregir la ubicación de los datos tomados en el terreno, lo que se analiza por medio de las condiciones de la imagen, vectores y ángulos de buzamientos, ya sea por los métodos de rayos o por los métodos de extrapolación de datos (Jones, 2014).

1.2 MIGRACIÓN POST APILADO

Realizar un proceso de post apilado que aumente las condiciones necesarias para la interpretación de un análisis sísmico adecuado es necesario en la mayoría de casos cuando la geología es compleja, así se reubican los puntos reflectores y los ángulos de buzamiento a la posición adecuada por medio de algoritmos, y por lo tanto se generan imágenes correctas. Específicamente son analizados los algoritmos más comunes que se utilizan en la actualidad, los cuales son: diferencias finitas, Kirchhoff, diferencias finitas omega-x y cambio de fase, en cuya comparación de resultados se aprecia como de mejor calidad el método Kirchhoff (Vásquez, 2010) , que además de ser el más utilizado en la actualidad, permite ángulos de buzamientos hasta de 90 grados con resultados confiables.

1.3 DESARROLLO PARALELO

Al realizar la implementación de un algoritmo de cómputo para realizar el proceso de la migración sísmica, se pueden presentar exigencias altas al procesador, principalmente cuando en la zona se presentan algunos perfiles de velocidad con complicaciones o cuando hay muchas trazas, por lo que en los últimos años se han aprovechado los adelantos tecnológicos para agilizar y perfeccionar el proceso. Una opción que se ha analizado es la de ejecutar el algoritmo en paralelo, lo que indica que será ejecutado sobre un clúster de procesadores de alto rendimiento corriendo bajo Linux y comunicándose por medio de una memoria compartida (Teran Julio & Pita Escamilla, 2009) . Así mismo se ha propuesto el desarrollo paralelo PCAM (Particionamiento, Comunicación, Aglomeración y Mapeo) como una buena opción para el aprovechamiento de los avances en cuanto al hardware se refiere, porque el mapeo maximiza tanto el uso del procesador como la comunicación, aunque cabe resaltar que la exigencia del proceso de migrado depende en gran medida del tipo de migración utilizada (Linares Moreno, 2014).

1.4 DISPOSITIVOS PROGRAMABLES

Otra opción analizada recientemente es la del uso de dispositivos que superen las barreras que presentan las CPU tradicionales en la computación paralela como lo son los dispositivos FPGA (Field Programmable Gate Array), que mejoran los problemas de comunicación y de velocidad (acelerando el proceso entre 8 y 70 veces dependiendo del puerto de transmisión), sin embargo conlleva a que este tipo de dispositivos se esfuerce más durante el proceso del algoritmo debido a su significativamente baja frecuencia de operación (Fajardo, Castillo Villar, & Pedraza, 2013) . Adicionalmente cabe destacar el continuo desarrollo tecnológico de éste tipo de dispositivos, ya que se desarrollan versiones nuevas más frecuentemente que las de una CPU tradicional, y en donde las empresas que desarrollan este tipo de dispositivos no sólo piensan en el hardware sino también en herramientas de diseño de software, de tal modo que se

facilite la programación de los algoritmos necesarios y adicionalmente se brinde el soporte necesario para ello (Abreo Carrillo & Ramírez Silva, 2010).

1.5 SOFTWARE UTILIZADO EN EL MODELADO SÍSMICO

Debido a la complejidad de los cálculos y la cantidad de procesos a realizar, en un análisis de datos sísmicos se debe utilizar el software adecuado. Según la ANH (Agencia Nacional de Hidrocarburos), los paquetes de software usados comúnmente en Colombia son Omni, Mesa y DirectAid, los cuales corresponden a software de modelado sísmico y tienen en cuenta en menor medida la migración o inversión sísmica. Sin embargo empresas como Ecopetrol en la actualidad utilizan ProMAX, que aunque ofrece un paquete muy completo para el procesado de datos sísmicos, cuenta con una licencia que no solo es privativa sino que es muy costosa y poco flexible, lo que lo hace limitado en el campo de aplicación.

1.6 SOFTWARE LIBRE PARA MIGRACIÓN SÍSMICA

Una buena opción para realizar una tomografía adecuada es el software PGPLOT, cuya función está comprendida por varias librerías y aplicaciones de código libre según licencia GPL por medio de las cual es posible realizar el trazado de curvas por el método de tiempo de tránsito para ser exportadas por medio de la librería PGPLOT a varios formatos de imagen comerciales (Forbriger, 2015).

Dejando de lado la imagen, y teniendo en cuenta la migración por el método de rayo y el cálculo de tiempo de viaje, existe TAUP Toolkit. Software que cuenta con licencia GPL, se ocupa de varios tipos de modelos de velocidad, y puede calcular los tiempos eficientemente. Está desarrollado en Java, lo que permite que sea ejecutado en casi cualquier sistema operativo (University of South Carolina, 2013).

Sin embargo, en la actualidad se encuentran muchas más opciones, algunas libres, otras de código abierto, también con diferentes funciones en cuanto a análisis de datos sísmicos, objetivo principal y lenguaje de desarrollo, algunas de ellas son:

Tabla 1. Software libre para modelamiento sísmico

Nombre	Descripción	Desarrollador	Licencia	Plataforma	Lenguaje	Fuente
Madagascar	Analiza datos en varias dimensiones	Sergey Fomel and others	GPL	Multiplataforma	C, C++, Python	(Multiple free proposals, 2017)
FreeUSP	Análisis de datos a nivel educativo	Amoco, BP	Abierta	Unix, Linux	Fortran 77, C	(Amoco Production Company, 2011)
CWP/SU (Seismic Un*x)	Librerías para análisis de datos sísmicos	Stanford, Colorado School of Mines	Libre, estilo BSD	Unix, Linux, OS X, Cygwin32	Fortran 77, C	(Colorado School of Mines, 2007)
SEPlib	Análisis e imagen de datos sísmicos 3D y 4D.	Stanford University	Abierta	Multiplataforma	Fortran, C	(Stanford University, 2008)
Kogeo	Análisis e imagen de datos sísmicos 2D y 3D	University of Hamburg	GPL	Windows	C++	(University of Hamburg, 2005)
JRG	Análisis e imagen de datos sísmicos 3D	John Louie, University of Nevada	Abierta	Multiplataforma	Java	(University of Nevada, Reno, 2016)
OpendTect	Imagen y tomografía	dGB Earth Sciences	Creative Commons	Multiplataforma	C++	(OpendTect OSR team, 2018)

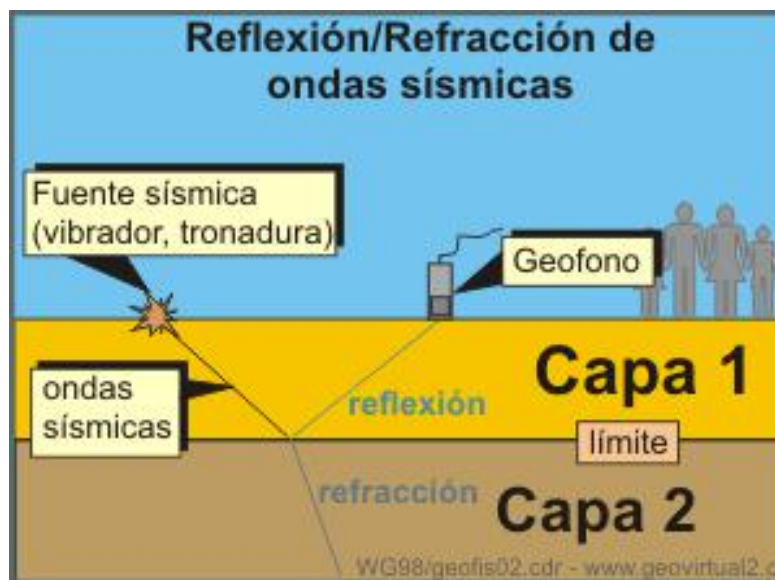
Fuente: Autor.

2. MARCO TEÓRICO

2.1 ADQUISICIÓN DE DATOS SÍSMICOS

El método geofísico más utilizado a nivel mundial en el campo de la prospección de hidrocarburos es el proceso de adquisición sísmica por reflexión. Este método consiste en la generación de ondas elásticas (por energía impulsiva o vibratoria) que se propaguen a través de las diferentes capas de suelo. A medida que la onda avanza desde la superficie terrestre se obliga a que se refleje una parte de la energía en cada cambio de estrato de suelo. La energía reflejada es recibida por geófonos, que son equipos ubicados en la superficie, cuya función es escuchar y almacenar en cintas magnéticas los datos de amplitudes de onda, longitudes de onda, fases y tiempos de viaje de onda.

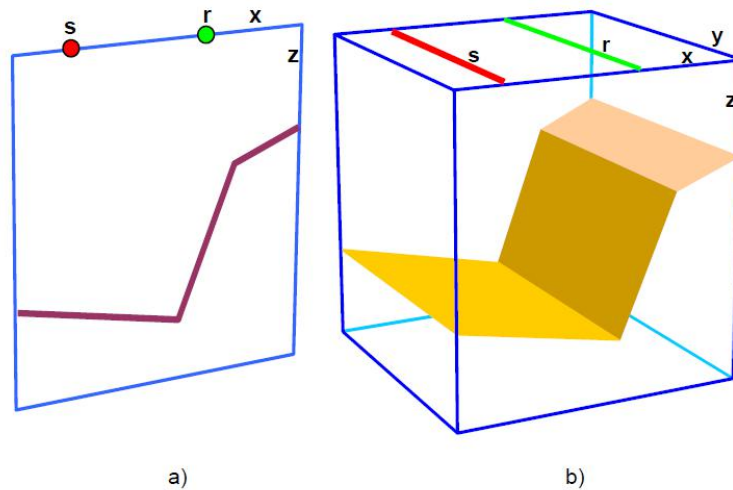
Ilustración 2. Comportamiento de las ondas sísmicas en una interface horizontal entre dos distintos medios litológicos.



Fuente: (Griem, 2015)

Dependiendo del tipo de imagen buscada (2D, 3D o 4D) se inicia el análisis de los datos, tanto los observados en campo como los obtenidos, de modo que se definan *offsets*, *azimut* y *fold* adecuados. Es en este punto en donde se inicia el proceso de análisis de datos sísmicos, que debido a la gran cantidad de datos obtenidos es necesario el uso de un software especializado que organice la información y aplique correcciones tanto de estática como de amplitud para proseguir con el análisis de datos sísmicos (Herrera & Cooper, 2010).

Ilustración 3. Modelos sísmicos, a) 2D, b) 3D



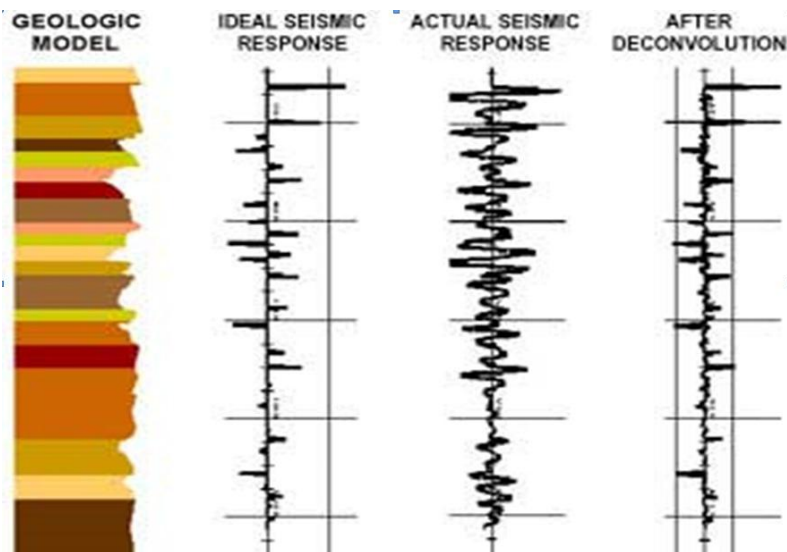
Fuente: (Bancroft, *A visual relationship between Kirchhoff migration and seismic*, 2002)

2.2 ANÁLISIS DE DATOS SÍSMICOS

El proceso de análisis puede dividirse en tres etapas, pre-apilado, apilado y post-apilado. La primera es el pre-apilado que consiste en aplicar varios algoritmos con funciones específicas en busca de mejorar la calidad de los datos iniciales. La última es el post-apilado que consiste en la corrección de los datos organizados, y a la vez puede incluir la migración sísmica.

El algoritmo inicial en el pre-apilado es el de reordenamiento de las trazas sísmicas en conjuntos de punto reflector común o CMP (*Common MidPoint*). Su objetivo es agrupar las trazas que por geometría pertenecen a un mismo punto medio entre una fuente y un receptor determinado; asumiendo que el espaciado entre CMP es la mitad del espaciado entre geófonos y que las reflexiones en estos conjuntos poseen también trayectorias hiperbólicas. Luego se debe aplicar un algoritmo de deconvolución, el cual es usado para aumentar la resolución temporal de las reflexiones, ya que comprime la señal al hacer convolucionar una onda invertida con cada traza como se muestra en la figura.

Ilustración 4. Deconvolución de una respuesta ideal.



Fuente: (Veritas Caspian LLP , 2010)

Existen diferentes métodos para realizar el proceso de deconvolución, sin embargo hay uno en especial con el que se reduce la cantidad de procesos realizados sin alterar significativamente los resultados, es la técnica MED (Deconvolución de Mínima Entropía) propuesto por Wiggins en 1978, que consiste en buscar la menor cantidad de datos con picos grandes que sean coherentes y por medio de un sistema matricial se invierten los datos en cada paso. La estimación de la entropía mínima en un plano de incidencia normal se puede expresar como la convolución entre dos componentes

básicos, la reflectividad (y_t) y las ondículas (w_t), que asumiendo una traza sin ruido en x_t entonces se expresaría por medio de la relación:

$$s_t = w_t * q_t \quad (1)$$

En donde * denota una convolución discreta, aunque es posible adoptar un esquema lineal como f_t , quedando.

$$q_t = s_t * f_t \quad (2)$$

La siguiente tabla resume las expresiones que están implicadas en el problema cuando las funciones $F(q_i) = q_i$ y $F(q_i) = \ln(q_i)$:

Tabla 2. Velocidad normalizada y salida b deseada. (a) Función de entropía Wiggins, (b) Función de entropía logarítmica.

$F(q_i)$	V	b_i
a) q_i	$\frac{1}{N} \sum_i q_i \cdot q_i$	$\frac{q_i \cdot y_i}{N \cdot V}$
b) $\ln(q_i)$	$\frac{1}{N \cdot \ln(N)} \sum_i q_i \cdot \ln(q_i)$	$\frac{(\ln(q_i) + 1)y_i}{N \cdot V + 1}$

Fuente: (Sacchi, Velis, & Cominguez, 1994)

Aunque dichos conceptos no son sencillos de definir y de aplicar, se presenta una solución de aplicación para éstas ecuaciones por medio del algoritmo de dominio de frecuencia descrito en (Sacchi, Velis, & Cominguez, 1994):

1. Se inicia asumiendo que $q_t = \hat{q}_t$
2. Se calcula b_t y B_k , donde B_k es la transformación discreta de Fourier

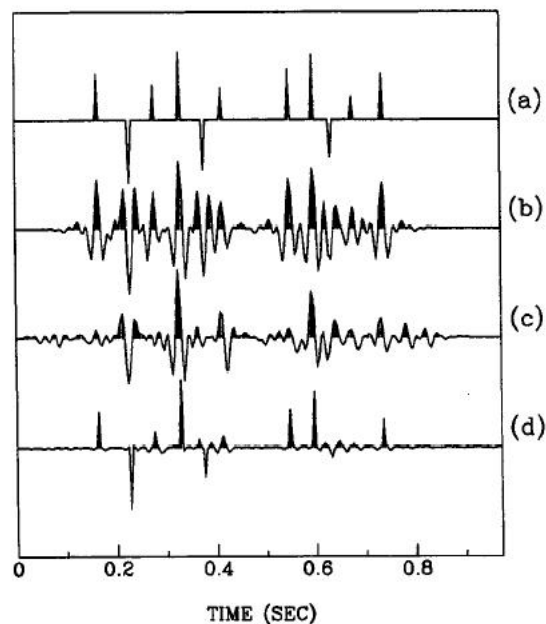
$$b_i = \frac{G(q'_i)q_i}{\frac{1}{N} \sum_j G(q'_j)q'_j} \quad (3)$$

$$G(q'_i) = F(q'_i) + q'_i \frac{\partial F(q'_i)}{\partial q'_i} \quad (4)$$

3. Se reemplazan las pérdidas de frecuencia baja y alta con B_k
4. A partir de la transformación inversa de Fourier, se calcula una estimación de la reflectividad. Se evalúa para comprobar la convergencia y se vuelve a iterar desde el paso 2.

Un ejemplo de prueba para el algoritmo en donde se aplica la tabla 2 y se comparan los resultados con otros métodos es el siguiente:

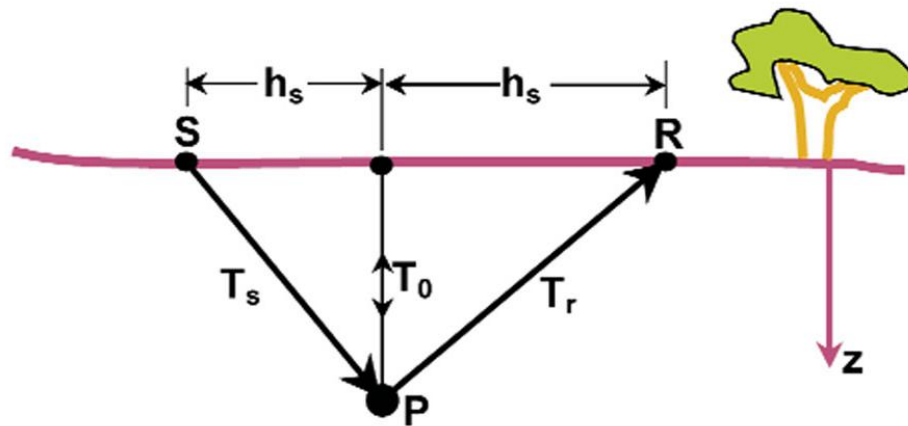
Ilustración 5. Figura de trazas sucesivas. (a) Reflectividad, (b) Reflectividad con límite de banda, (c) MED y (d) mínima entropía con constantes de frecuencia de dominio.



Fuente: (Sacchi, Velis, & Cominguez, 1994)

El algoritmo que continúa en el proceso es el de correcciones dinámicas (NMO), el cual transforma la trayectoria hiperbólica de los eventos de reflexión en el eje del tiempo, en otras palabras, convierte la trayectoria en una línea horizontal (que pasa a cero el offset) de manera que todas las trazas al ser sumadas estarán en fase. La forma de conseguir dicha alineación es mediante la asignación de la velocidad de la trayectoria de reflexión. Se procede con el algoritmo de análisis de velocidad, el cual se aplica sobre los conjuntos o grupos de conjuntos de CMP, buscando mejorar la relación señal/ruido y obteniendo finalmente un espectro de velocidades de mejor calidad para el proceso de apilamiento.

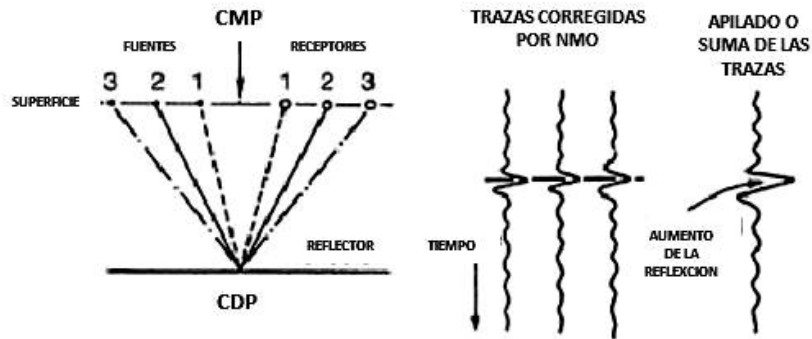
Ilustración 6. Diagrama de una traza.



Fuente: (Bancroft, Seismic Imaging: Prestack, 2002)

La segunda etapa del proceso es el apilado, y su propósito no es otro que el de sumar las trazas CMP, o en pocas palabras, sumar las trazas que tienen un mismo punto reflector, con el fin de obtener una sección sísmica que servirá para representar una imagen de los reflectores presentes en el subsuelo y con offset igual a cero. Sin embargo, dicha sección sísmica aún no es la definitiva, por lo que es necesario continuar con el proceso buscando siempre la mejora de los datos.

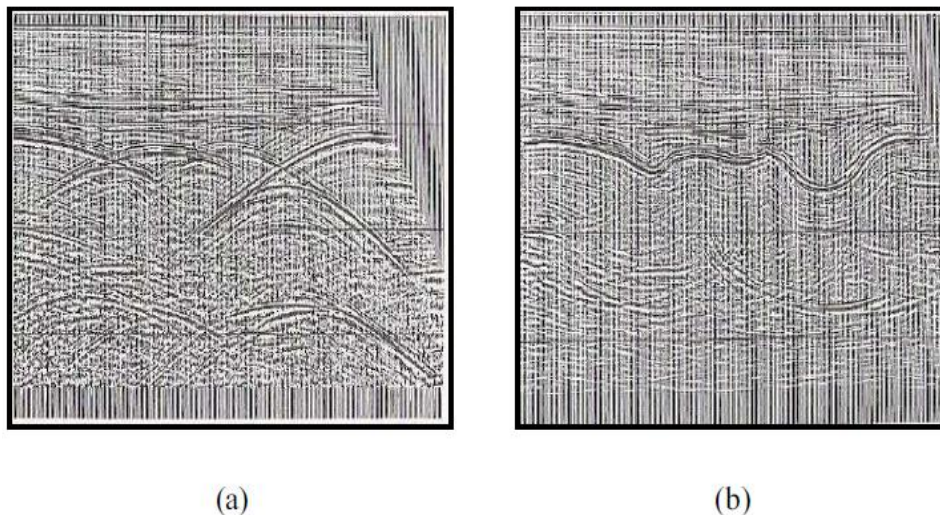
Ilustración 7. Apilado de trazas para un mismo CMP.



Fuente: (Benítez R., 2012)

La etapa final es el post-apilado, el cual inicia con un algoritmo que nuevamente convolucionna las trazas. Este proceso es necesario para que corrija las perdidas y elimine el ruido generado durante el apilado en frecuencias altas y bajas. El proceso continúa aplicando la migración sísmica, que se hace necesaria al momento de corregir la ubicación real de los puntos reflectores, lo que se analiza por medio de las condiciones de la imagen, los vectores y los ángulos de buzamientos. Los métodos de migración más utilizados actualmente son: diferencias finitas, Kirchhoff, diferencias finitas omega-x y cambio de fase.

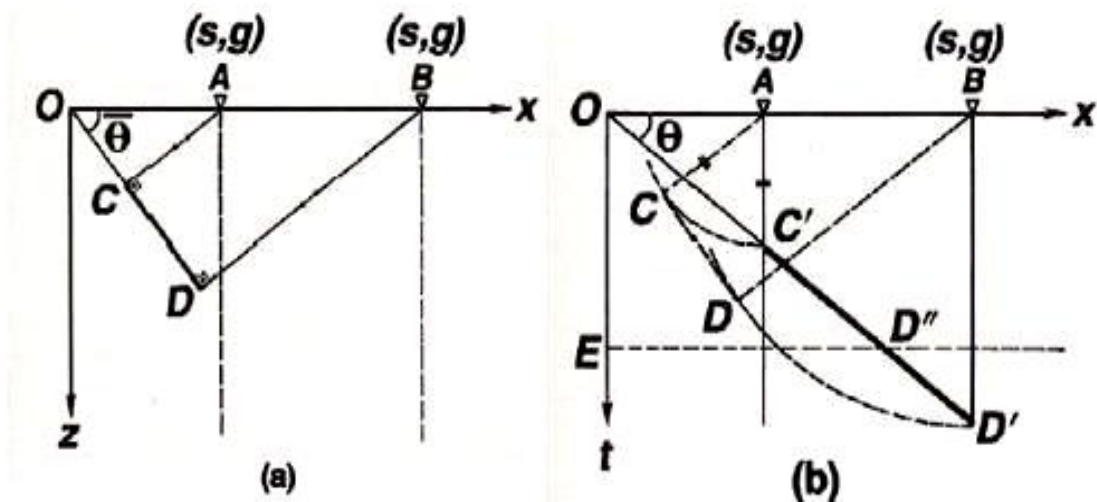
Ilustración 8. Datos después del apilado, a) Antes de la migración b) Después de la migración.



Fuente: (Vásquez, 2010)

Dichas etapas son utilizadas en todos los procesos de análisis sísmico por reflexión, sin embargo los algoritmos o procesos utilizados en cada una de ellas puede variar de acuerdo a las necesidades del proyecto o las preferencias de la persona que lo desarrolla. Como se aprecia en (Herrera Rivero, 2005), en donde se definió y analizó una secuencia de pasos para realizar el análisis de datos en dos trazas 2D. Proceso por el cual se tomaron datos sísmicos y se procesaron hasta lograr datos migrados que permitieran representar puntos de falla y estructuras presentes en el subsuelo. Se utilizó la migración antes y después del apilado. En una comparativa general de los resultados obtenidos en un proceso de migración sísmica que funcione en la mayoría de casos se observan muy buenos resultados en la opción del método Kirchhoff (Vásquez, 2010), que además de ser el más utilizado en la actualidad, permite ángulos de buzamiento hasta de 90 grados con resultados confiables.

Ilustración 9. (a) Sección geológica, (b) Migración del segmento C'D' a su verdadera posición CD en la sección en tiempo.



Fuente: (Benítez R., 2012)

También se aprecia una comparación entre migraciones en el documento de (Benítez R., 2012), en el que se busca optimizar los datos obtenidos de los estudios realizados en el proyecto Oro Negro, creado en Venezuela con el fin de reducir la incertidumbre en la prospección de hidrocarburos, conociendo que la relación señal-ruido (S/R) es baja en algunos puntos evaluados, y los resultados de las migraciones realizadas no

ofrecen una buena calidad de imagen, el autor realizó la evaluación de la migración antes de apilamiento en profundidad por los métodos de Kirchhoff y Ecuación de Onda (E.O.). Las conclusiones del documento presentan como mejor opción para este caso de análisis el método de migración por la ecuación de onda, sin embargo aclara que se debe realizar una ampliación de los datos para una nueva evaluación de resultados.

2.3 MIGRACIÓN POR DIFERENCIAS FINITAS

Esta opción utiliza el método de extrapolación, asumiendo $\tau = 0$, con el fin de poder reconstruir las señales propagadas en la superficie, “El método usa la extrapolación para deducir las posiciones reales de los cambios de impedancia acústica por medio de la derivada de la ecuación de onda $\left(\frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial z^2} = \frac{1}{v^2} \frac{\partial^2 P}{\partial t^2}\right)$, considerando un plano vertical (x, τ) , donde el eje de abscisas está en la superficie de la Tierra, y z es vertical y orientados hacia abajo x da a la distancia horizontal, y la z es la profundidad en términos de onda de doble dirección en tiempo de viaje de la superficie. Una onda de presión, que viaja en ese plano, es iniciada en (x, z) y es registrado en (x, z) después de la reflexión, como $P(x, z, t)$ ” (Vásquez, 2010).

2.4 MIGRACIÓN OMEGA-X

Según (Vásquez, 2010) esta migración emplea el método de diferencias finitas para dar solución a la ecuación de ondas, aunque difiere de este método en que es exacto hasta un máximo de 65 grados en el Angulo de buzamiento usando la continuación hacia abajo del campo. Sin embargo este máximo depende no solo de la calidad y cantidad de los datos del muestreo, sino también del contenido de frecuencia de los datos sísmicos. Evalúa principalmente los pares de tiempo y máxima frecuencia, también el incremento vertical y un multiplicador para el incremento vertical.

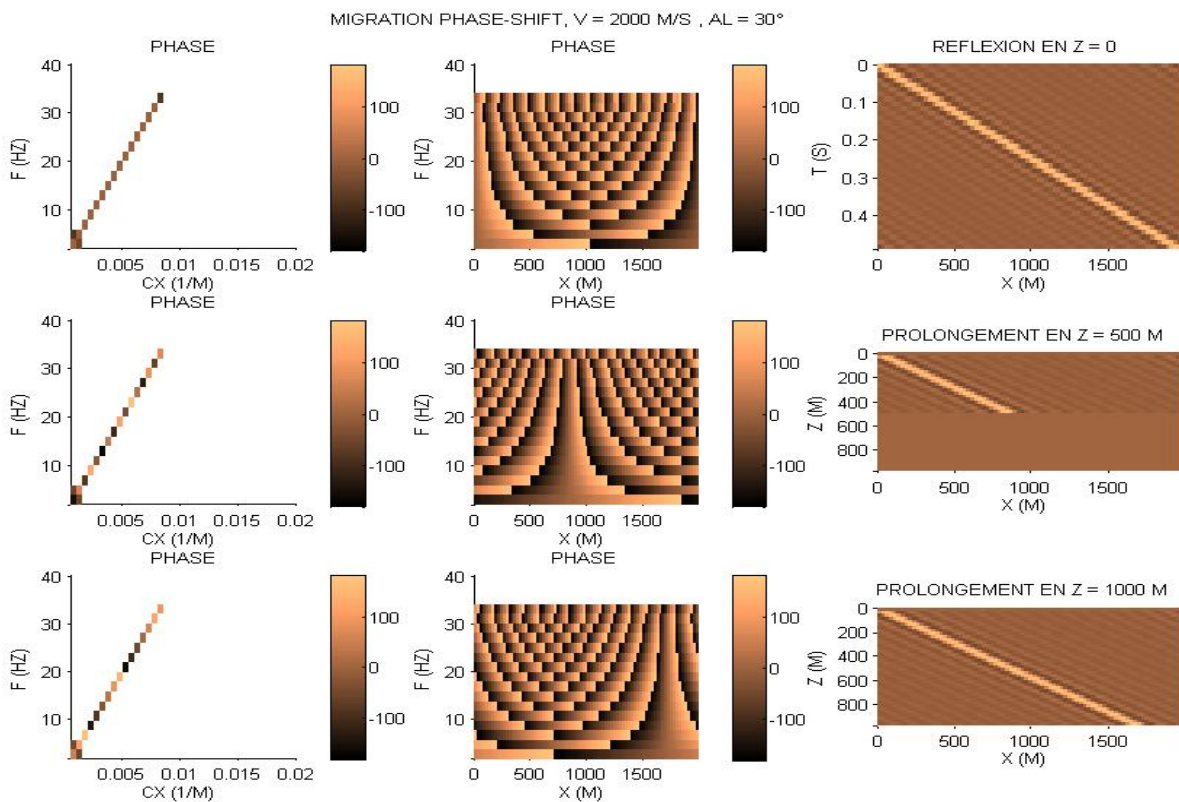
2.5 MIGRACIÓN CAMBIO DE FASE

Según (Bonomi & Cabitza, 1994) se asume que la velocidad de onda dividida en 2 es constante y $P(X,Z,t)$ está representado por una serie de Fourier doble:

$$P(X,Z,t) = \sum_{kz} \sum_w P(k_x, z, w) \exp [i(k_x X + wt)] \quad (5)$$

Siendo k_x el punto medio y w el número de onda en el tiempo.

Ilustración 10. Ejemplo de las etapas iniciales de la migración por cambio de fase, a una velocidad constante y una velocidad V (z).



Fuente: (Marthelot, 2015)

“Consiste en la solución al dominio de la frecuencia, de la ecuación de ondas unidimensional. Las condiciones iniciales de esta ecuación diferencial parcial son las de una sección con distancia fuente-receptor igual a cero. Considerando un plano vertical (x,τ) , donde el eje de abscisas está en la superficie de la Tierra, y los τ -ejes son verticales y orientados hacia abajo. x es la distancia horizontal, y la τ es la profundidad en términos de onda de doble dirección en tiempo de viaje de la superficie. Una onda de presión, que viaja en ese plano, es iniciada en (x,τ) y es registrado después de la reflexión, como $p(x,\tau,t)$; finalmente gobernado por la ecuación de onda.” (Vásquez, 2010).

2.6 MIGRACIÓN DE KIRCHHOFF

Según (Gazdag & Sguazzero, 1984), el teorema de la integral Kirchhoff expresa el valor del campo de ondas en un punto arbitrario en términos del valor de campo de ondas y su derivada normal en todos los puntos en una superficie cerrada arbitraria que rodea el punto. En la práctica, las mediciones se realizan solamente en la superficie, por lo tanto, en lugar de una superficie cerrada, la integración debe limitarse a la superficie de la tierra. Además, en la práctica sísmica, sólo se registra el campo de ondas, y su derivada normal no está disponible.

Según (Vásquez, 2010) este método se basa en la suma de las difracciones, y consiste en generar y colapsar hiperboloides de difracción de máxima convexidad y curvatura, dependiente de la velocidad (que suele ser la misma del apilamiento) al que se le añade un factor de escala y cambio de fase antes de la sumatoria. Para lograr el objetivo, el algoritmo se basa en la solución integral de la ecuación escalar de ondas:

$$P_{salida}(x,\tau,t) = \frac{1}{2\pi} \int \left[\frac{\cos\theta}{r^2} P_{entrada}(x_{entrada}, \tau = 0, t - \frac{r}{V}) + \frac{\cos\theta}{Vr} \frac{\partial}{\partial r} P_{entrada}(x_{entrada}, \tau = 0, t - \frac{r}{V}) \right] dx \quad (6)$$

En donde $P_{salida}(x,\tau,t)$ se refiere al campo propagado para cualquier posición (x,τ) , donde x es la distancia horizontal en la superficie en unidades de longitud y τ es la

distancia vertical o profundidad. $P_{entrada}(x_{entrada}, \tau = 0, t)$ se refiere a la sección de incidencia normal, medida en superficie.

Entonces:

θ : Se refiere al ángulo de incidencia de la onda.

V : es la velocidad RMS en un punto (x, τ)

$$r = [(x_{entrada} - x)^2 + \tau^2]^{\frac{1}{2}} \quad (7)$$

t : es el tiempo de muestras por segundos, representado por los siguientes dos casos:

- Si la sección esta apilada con distancia fuente-receptor igual a cero, entonces:

$$t = \left[t_0^2 + \left(\frac{x}{2v} \right)^2 \right]^{\frac{1}{2}} \quad (8)$$

- Si los datos están en CMP entonces:

$$t = \left[\left(\frac{t_0^2}{4} + \frac{x_s^2}{v^2} \right) + \left(\frac{t_0^2}{4} + \frac{x_r^2}{v^2} \right) \right]^{\frac{1}{2}} \quad (9)$$

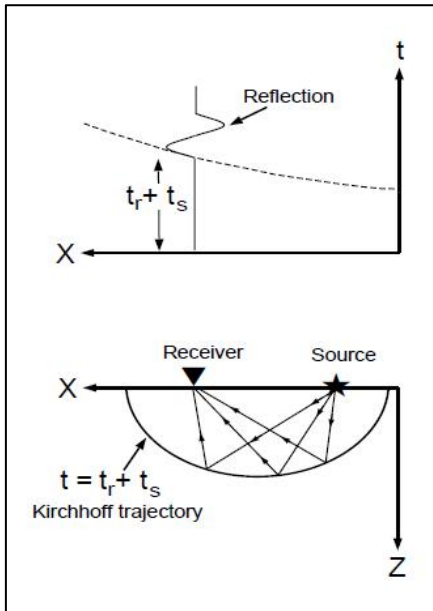
t_0 : Es el tiempo donde se tomó la muestra en segundos en la sección distancia fuente-receptor igual a cero.

v : Velocidad de la muestra en la posición de distancia fuente-receptor igual a cero.

x_s : Distancia de la fuente a su posición en distancia fuente-receptor igual a cero.

x_r : Distancia del receptor a su posición en distancia fuente-receptor igual a cero.

Ilustración 11. Velocidad media constante, Trayectoria Kirchhoff en tiempo de viaje.



Fuente: (Chávez Pérez, 1997)

Una forma simple de aplicar la teoría de migración Kirchhoff en post-apilado se observa en el algoritmo de Jon F. Clarebout, publicado en 1997 (Stanford Exploration Project, 1997), el algoritmo se basa en la ecuación de la relación circulo-hipérbola (10), de tal forma que al reemplazar la variable τ , por z de la ecuación (11) que sería su equivalente a la profundidad en una gráfica en 2D y luego de despejar el tiempo se forma la ecuación (12), que equivale al dato de profundidad convertido en un dato de tiempo y por lo tanto al aplicarlo en la ecuación (13) se puede obtener el número de traza que se debe corregir.

$$t^2 = \tau^2 + \frac{x^2}{v^2} \quad (10)$$

$$\tau = z_0 + dz \cdot nt \quad (11)$$

$$t = \sqrt{(z_0 + dz \cdot nt)^2 + \left(\frac{x-y}{v}\right)^2} \quad (12)$$

$$ntc = 0.5 + \frac{t - t_0}{dt} \quad (13)$$

Dónde:

z_0 : Profundidad inicial (m).

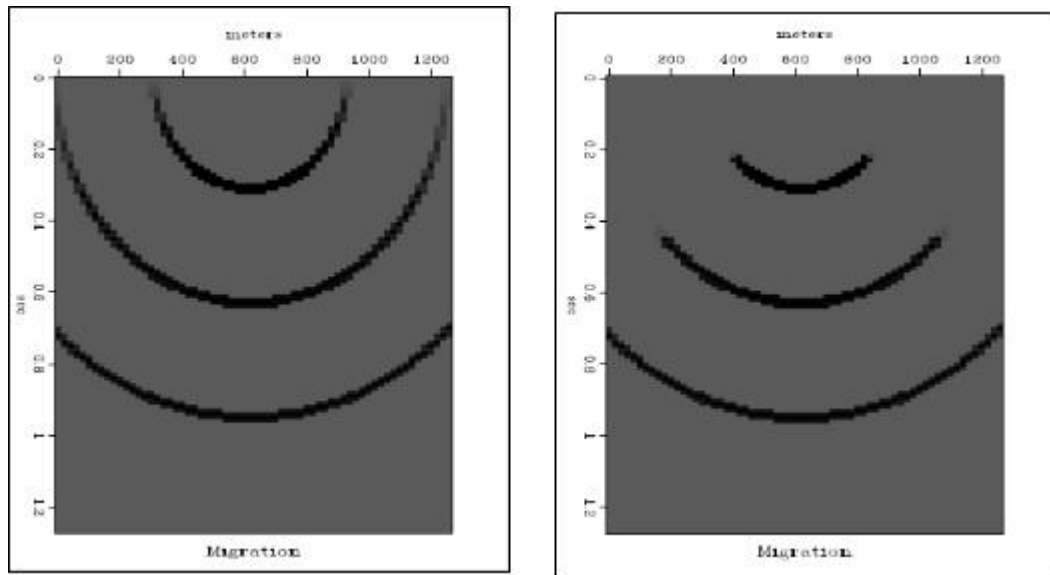
dz : Distancia vertical (m).

nt : Número de traza que se desea corregir.

ntc : Número de traza que se debe corregir.

El siguiente es un ejemplo de aplicación, en donde por medio de una subrutina se aplica un algoritmo en el que se presenta la solución a la migración por la ecuación de Kirchhoff. Está desarrollado en lenguaje Fortran y realiza el proceso por medio de la suma de los datos de entrada a lo largo de una trayectoria hiperbólica (Código incluido en el anexo A).

Ilustración 12. Comparación de los resultados de la migración antes (izquierda) y después (derecha).



Fuente: (Wong & Terwilliger, 2012)

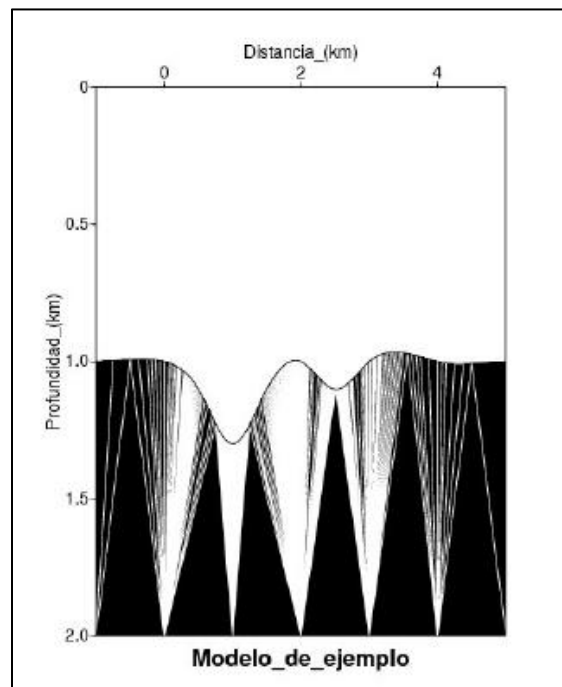
Aplicar la migración por el método de Kirchhoff durante el pre-apilado presenta altas exigencias de recursos computacionales, y por lo tanto los tiempos de cómputo

requeridos al realizar dicha migración son altos. Al realizar una evaluación al algoritmo de migración de Kirchhoff en 2D, con el fin de medir el rendimiento en tiempo de procesado de forma analítica, puede observarse un significativo ahorro en tiempo, a tal punto que en algunos casos superó el 80% y con errores casi imperceptibles; esto al comparar los resultados con otros algoritmos de migración y adicionalmente apoyándose en un clúster de procesadores (Teran Julio & Pita Escamilla, 2009).

2.7 MODELADO

Finalmente se establece que para poder interpretar los datos y ver el resultado de todo el proceso es necesario realizar el modelado, generando así una gráfica del modelo del subsuelo; por medio de una grilla 2d ya sea en escala de grises o con colores, el proceso consiste en ubicar números distribuidos en un área específica, de igual forma que se hace en topografía.

Ilustración 13. Ejemplo de modelado 2D.



Fuente: (Martínez Macedo, 2008)

2.8 SOFTWARE PARA MIGRACIÓN SÍSMICA

Dentro del software existente para realizar el proceso de migración sísmica se encuentran varias opciones que en su mayoría no sólo cuentan con una licencia de software privativo sino que su precio es demasiado elevado. Dentro de las opciones existentes también se encuentran algunas cuyo código es abierto, en especial se encuentran las librerías del software Seismic Unix (SU), que es la alternativa más común en la academia, sin embargo la orden de ejecución para realizar el proceso requerido debe realizarse por medio de línea de comandos, ya que dicho conjunto de librerías no posee ninguna interfaz gráfica (Colorado School of Mines, 2007).

También existe Tomo2D, software desarrollado en C++, distribuido como aplicación de código abierto; comprendido también por un conjunto de librerías, pero en esta ocasión se cuenta con una librería cuyo objetivo es proporcionar una implementación tomográfica por tiempo de tránsito, por medio tanto de refracción como de reflexión, lo cual implica un proceso bastante útil para lograr una comparación de resultados en la inversión sísmica (Korenaga, 2003).

2.9 HERRAMIENTAS LIBRES PARA DESARROLLO

Uno de los lenguajes más utilizados para el desarrollo de aplicaciones en la actualidad es JAVA. Este es un lenguaje que ofrece muchas ventajas en el mercado del software libre, y algunas de estas podrían beneficiar el desarrollo de la aplicación, entre otras están:

- Consiste en un lenguaje multiplataforma cuya ejecución puede darse en diferentes tipos de hardware y sistemas operativos.
- Cuenta con licencia libre (GNU GPL).
- Es muy estable
- Puede ser desarrollado desde diferentes entornos de desarrollo.

- Actualmente se encuentra mucha información de soporte tanto en manuales como en foros.

Uno de los entornos de desarrollo más populares para JAVA es Eclipse, porque cuenta diferentes complementos que pueden irse añadiendo y adaptando según las necesidades que surjan en el proyecto. Es compatible con el IDE de Java y su compilador, también tiene como ventaja su licencia de código abierto, lo que finalmente lo convierte en una muy buena alternativa para el desarrollo de la aplicación.

2.10 ARCHIVOS DE DATOS SÍSMICOS

Algunos de los datos tomados en campo o datos procesados pueden ser encontrados en línea por medio de bancos de datos, generalmente estos datos se encuentran en formato SEG-Y que es el formato estandarizado para almacenamiento de datos sísmicos, aunque también se encuentran en formato tipo *SHAPE* que es el formato utilizado por el programa ArcGIS. La obtención de los archivos con estos datos puede encontrarse dependiendo del caso de uso, es decir, si se necesitan datos sísmicos de forma educativa pueden descargarse desde la página de varias universidades como es el caso de la universidad de Utah, (<http://utam.gg.utah.edu/SeismicData/SeismicData.html>). En un caso real de estudio pueden encontrarse según el país de origen, en el caso de Colombia algunos se pueden descargar desde la página oficial de la Agencia Nacional de Hidrocarburos, en la sección de Selección de datos, otros se pueden obtener por medio de una solicitud, (<http://www.anh.gov.co/Banco%20de%20informacion%20petrolera/EPIS/Paginas/Seleccion-de-datos.aspx>); también se pueden encontrar una biblioteca muy completa en la página del gobierno de Canadá (<http://open.canada.ca/data/en/dataset/84b5f70e-d407-54c5-949f-689f1df888f0>).

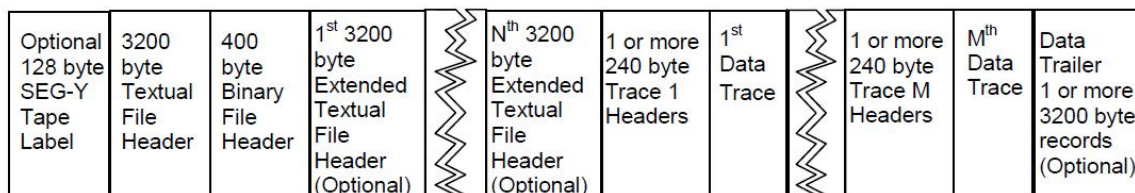
3. MÉTODO DE INVESTIGACIÓN

Debido a las características del proyecto se realizó la investigación por medio de la metodología exploratoria, de tal forma que se ampliaron los datos que se tienen sobre la migración sísmica, sus algoritmos y resultados; por esto se desarrolló la investigación mediante la consecución de pasos que finalmente llevaron a cumplir con los objetivos planteados:

3.1 DESARROLLO DE UNA LIBRERÍA EN JAVA PARA LECTURA Y ESCRITURA DE ARCHIVOS (SEG Y)

El formato generalizado para el almacenamiento de datos sísmicos que utilizan las empresas dedicadas a la toma de datos y/o análisis sísmico, y el más aceptado en la industria de la exploración sísmica es SEG-Y, un formato binario organizado que brinda el acceso a los datos de forma ordenada, segura y ágil; por lo que se tuvo en cuenta como requisito que el software desarrollado contara con una librería que permitiera leer y escribir los datos sísmicos en dicho formato. Para dar solución a este requerimiento se desarrolló una librería en Java.

Ilustración 14. Estructura de la secuencia de bytes de un archivo SEG-Y con N registros de encabezado de archivo de texto extendido y M registros de trazas.



Fuente (SEG Technical Standards Committee, 2017).

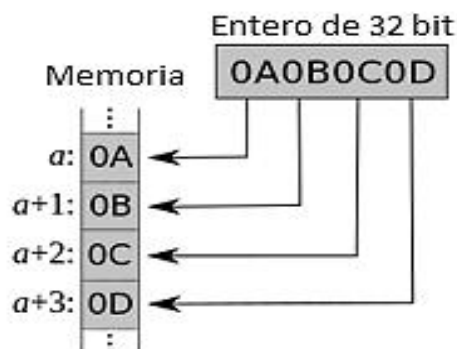
El sistema binario en el que se encuentran almacenados los datos del formato SEG-Y son: el tamaño de datos que posee el encabezado del documento, el encabezado de

cada traza y sus datos correspondientes; la información necesaria para entender dicho formato se encuentra en el manual Seg-Y (SEG Technical Standards Committee, 2017), en especial se analiza la **Ilustración 14** que corresponde al número de bytes de cada uno de los contenidos del archivo.

Para el desarrollado de la librería se utilizaron únicamente algunos de estos campos, pues el proceso realizado no requiere todo el contenido del encabezado en detalle; los campos utilizados corresponden a los campos (3221-3222) número de datos por traza, (3513-3520) número de trazas y (3601-...) contenido de cada traza omitiendo el encabezado específico de cada una.

La lectura del archivo inicia con un proceso para tomar cada dato de forma binaria y a medida que se avanza se forman grupos de 2 o de 4 bytes para convertirlos en datos numéricos según el valor almacenado. Del encabezado se omiten los valores no necesarios para agilizar el proceso, pero es importante resaltar que el tipo de conversión de los datos numéricos de cada una de las trazas está basado en el sistema *big-endian*, que consiste en escribir los valores en el mismo orden de lectura, caso que se aplica a grupos de 4 bytes.

Ilustración 15. Formato *big-endian*



Fuente: Autor.

La parte final de la lectura consiste en un algoritmo para la conversión de datos, que consiste en un proceso que se realiza a cada grupo de 4 bytes con el fin de crear un valor con un número natural. Se inicia asumiendo que si cada uno de los 8 bits que contiene cada byte se obtendrían 32 bits de tal forma que se toman en orden el signo, el exponente y la mantisa de la siguiente forma:

Tabla 3. Formato para conversión de valores 32bits.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	1	0	0	0	1	1	0	0	0	1	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Signo	Exponente							Mantisa																							

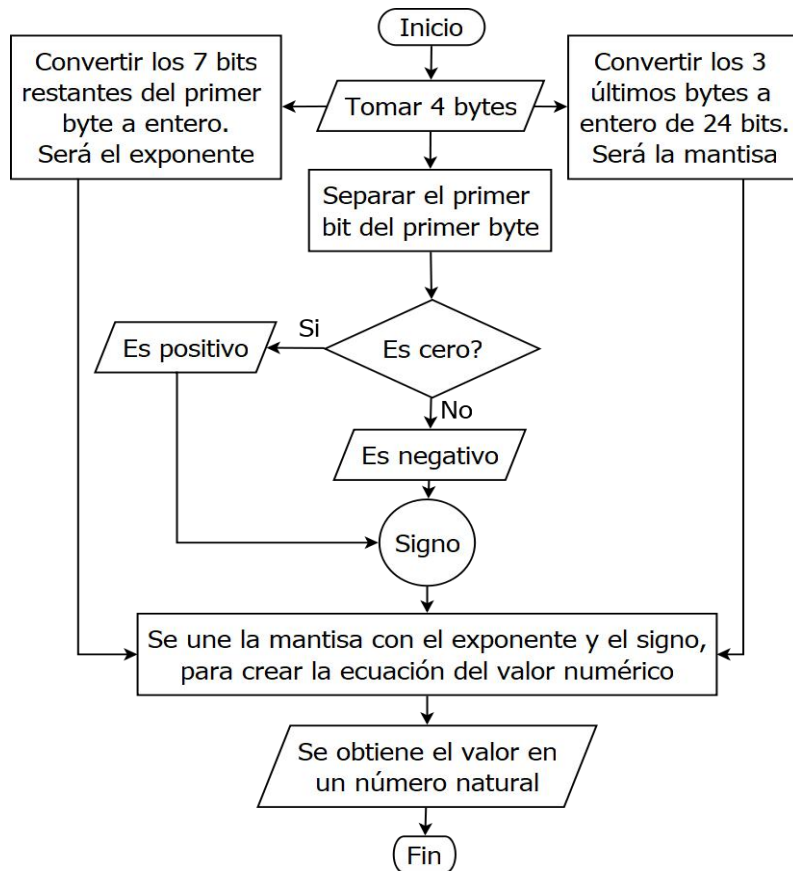
Fuente: Autor.

- Se puede entender entonces que el primer bit corresponde a 0 si el valor es positivo y a 1 cuando es negativo.
- Los 7 bits siguientes corresponden al exponente.
- Los 24 bits restantes que forman la mantisa.

Con estos tres datos se procede a resolver la ecuación (14) que dará como resultado el valor numérico del dato.

$$valor\ numérico = signo \cdot \sqrt[exp-64]{16} \cdot mantisa \quad (14)$$

Ilustración 16. Algoritmo para la conversión de datos binarios a naturales.



Fuente: Autor.

Luego de formado el valor en números naturales, se escriben los datos uno a uno en un arreglo de enteros en dos dimensiones, de tal forma que se pueda tener acceso a los datos fácilmente ubicando la primera variable del arreglo en el número de traza y la segunda en la posición del dato dentro de la traza.

La escritura del archivo binario SEG-Y se realiza invirtiendo el proceso de la lectura. En este caso se escriben los valores de los datos de cada traza y se conservan los encabezados tal cual fueron leídos, con el fin de no afectar la integridad del archivo o las trazas, y también porque no hay la necesidad de modificar ningún campo.

3.2 DESARROLLO E IMPLEMENTACIÓN DEL ALGORITMO PARA LA MIGRACIÓN CON LA ECUACIÓN DE KIRCHHOFF 2D

De acuerdo al análisis realizado en el marco teórico sobre los métodos de migración existentes, se definió que el desarrollo del algoritmo se realizaría resolviendo el problema de migración sísmica por medio de la ecuación de Kirchhoff, siendo éste el método de mejor desempeño, con mayor calidad en los resultados y con mayor amplitud de ángulos de buzamiento. Cabe resaltar que el método se aplica a los datos como proceso post apilado, de tal forma que se reduzca el tiempo al procesar los datos sin reducir en mayor medida la calidad de la migración.

Durante el desarrollo del algoritmo se encontraron factores como la tabla de velocidades, distancias iniciales, y factores dx y dy que son muy relevantes al momento de aplicar el proceso. Para facilitar el desarrollo se utilizó como base el algoritmo de Jon F. Clarebout, publicado en 1997 (Stanford Exploration Project, 1997), cuya finalidad es la misma, pero está desarrollado para lenguaje C y Fortran. Dicho algoritmo se basa en la relación circulo-hipérbola de la ecuación (15), y por medio de múltiples iteraciones realiza la sumatoria de Kirchhoff, añadiendo datos a lo largo de la hipérbola en el punto específico de cada traza, como puede observarse en el algoritmo de migración en la **Ilustración 18**.

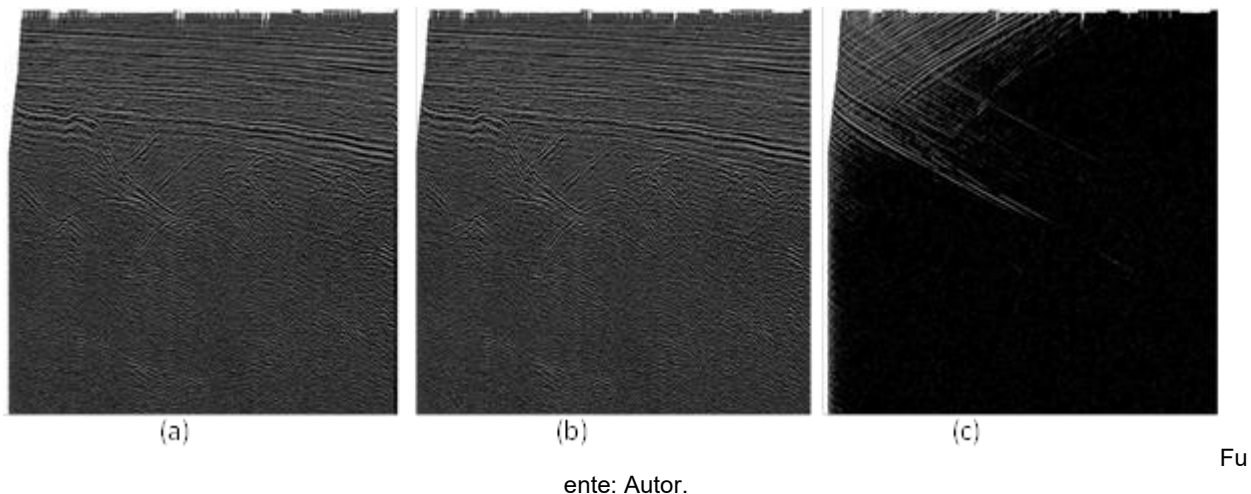
$$t^2 = \tau^2 + \frac{x^2}{v^2} \quad (15)$$

Para la ejecución del algoritmo se deben ingresar algunas variables adicionales a las contenidas en el archivo binario. Por lo tanto se lee el archivo de propiedades y cargan los valores de dx, dt, to, y de velocidad media. Esto quiere decir que se debe contar con estos valores dependiendo de cada proyecto. En caso de no contar con estos datos, existe la posibilidad de utilizar los valores predeterminados. Los valores predeterminados se obtuvieron por medio de pruebas con varios archivos de datos en

forma de ensayo y error. Estos valores funcionaron en la mayoría de casos y se establecieron observando las imágenes obtenidas en cada caso, estos valores son:

- desplazamiento horizontal (dx) = "0.04 metros".
- desplazamiento en el tiempo (dt) = "0.04 metros".
- tiempo inicial (t_0) = "1 segundo".
- velocidad media = "0.1 m/s".

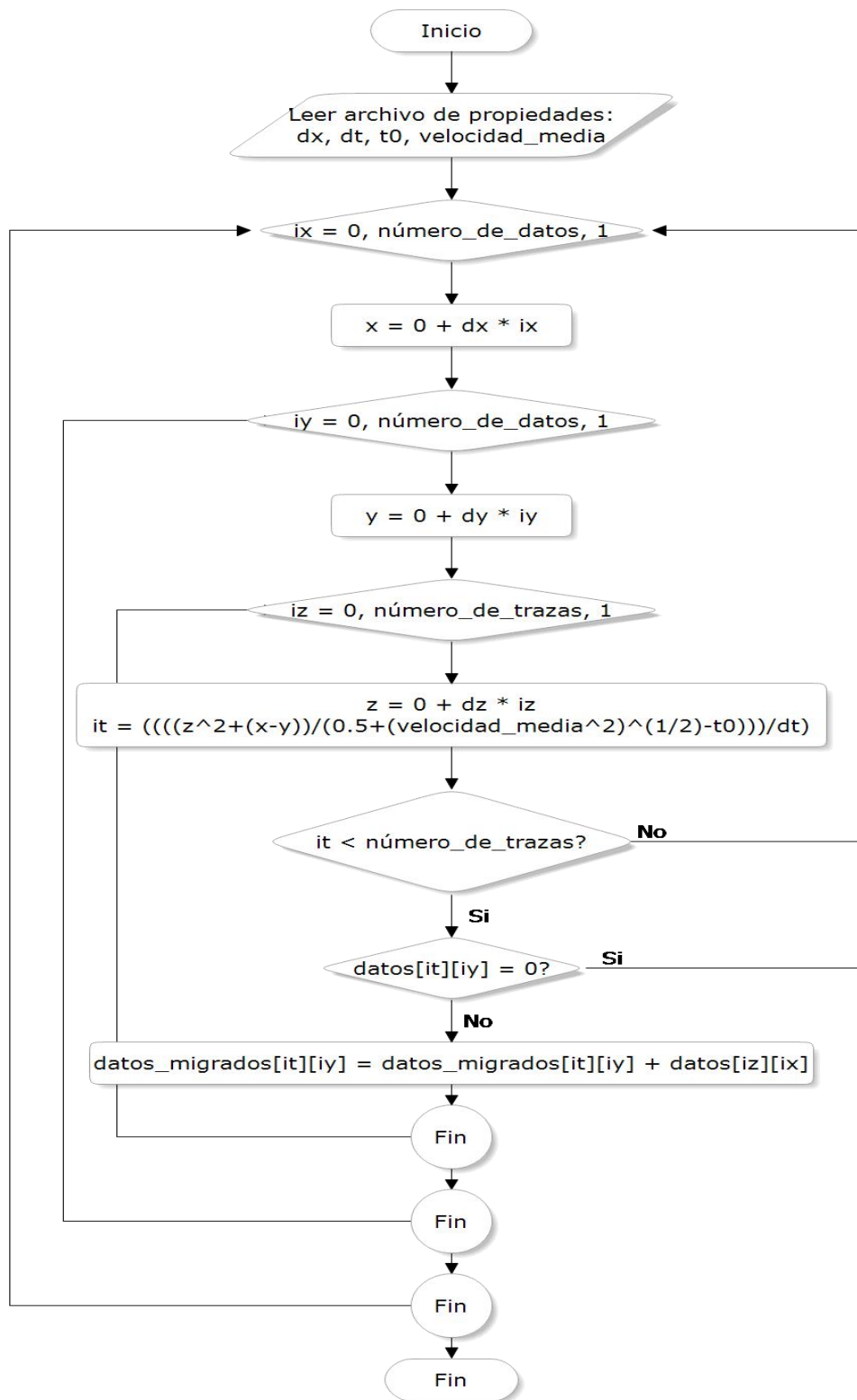
Ilustración 17. Comparativo de imágenes luego de la migración con diferentes velocidades.



En la **Ilustración 17** se observan 3 casos como ejemplo de la comparación:

- En la imagen (b) se observa la gráfica de los datos originales.
- En la imagen (a) se observa la migración con una velocidad de 0.01m/s y puede observarse que no existen cambios.
- En la imagen (c) se observa la migración con una velocidad de 1m/s, en donde puede observarse la pérdida total de los ángulos de buzamiento originales y la distorsión de los puntos reflectores.

Ilustración 18. Algoritmo para la migración.



Fuente: Autor.

El proceso que se realiza en el algoritmo consiste en hacer iteraciones en tres niveles que “barren” cada traza y cada dato de cada traza, con el fin de hallar los valores de las posiciones que se van a utilizar en la sumatoria de Kirchhoff; creando finalmente un nuevo arreglo de trazas en donde cada valor ha sido sometido al problema de la inversión sísmica.

Como puede observarse en el algoritmo se utiliza una velocidad media en lugar de un archivo externo con el análisis de velocidades para cada traza, ya que es poco necesario para la migración post apilado debido a que los datos apilados en CMP cuentan con la corrección de velocidades y es suficiente con una velocidad media para todas las trazas. Se resalta éste aspecto ya que es muy importante porque en él se basa la diferencia de realizar una migración pre apilado y una post apilado.

3.3 DESARROLLO DE LA LIBRERÍA PARA GENERAR IMÁGENES

Para la generación de imágenes ya sean antes o después de aplicar el proceso de migración se debe tener en cuenta el orden en que se almacenaron los datos leídos, de esa forma se puede conservar la integridad de la información y obtener una imagen de calidad. Una característica muy importante que se tuvo en cuenta es que el signo de cada valor separa el dato leído en tiempo, es decir, los valores negativos implican el tiempo que toma el disparo en llegar al punto de refracción y los positivos el tiempo del punto de refracción a la superficie.

La librería consiste simplemente en un proceso para leer y ubicar el dato según su posición en un plano cartesiano, de tal forma que no se aplicaron mayores ecuaciones o algoritmos especiales en esta función; sin embargo, se aplicó una relación entre los datos para poder obtener una buena relación de aspecto y así poder obtener una imagen clara y bien definida. La relación consiste en crear una variable de escala que lleva la amplitud máxima de los datos, lo que conserva un valor que sea aplicable a cada valor de la traza.

$$escala = \frac{ValorMáximo - ValorMínimo}{2} \quad (16)$$

Finalmente se ejecuta el trazado de líneas horizontales desde el eje de la traza hasta el valor del dato de la traza.

3.4 DESARROLLO DE LA INTERFAZ GRÁFICA PARA EL USUARIO

Con el fin de facilitar el manejo del software se realizó una interfaz gráfica para el usuario, por medio de la cual se seleccione el archivo SEG-Y con los datos, luego de la lectura se ilustra una imagen con los datos sísmicos, sin embargo, luego de realizar el proceso de migración por medio del menú se cambiará la imagen inicial por la imagen migrada.

Al finalizar el proceso y a gusto del usuario, será posible seleccionar la exportación de los datos a un nuevo archivo SEG-Y, en donde también se almacenarán las dos imágenes obtenidas (antes y después de aplicar el proceso de migración).

Para realizar el proceso de migración satisfactoriamente se requiere dar valores a las variables básicas, rellorando los campos en el menú antes de realizar el proceso, estos campos son:

- desplazamiento horizontal (dx).
- desplazamiento en el tiempo (dt).
- tiempo inicial (t0).
- velocidad media.

3.5 PRUEBA Y CORRECCIÓN DE FALLOS

Se prestó especial atención al formato utilizado en la lectura de los datos para no obtener datos errados, pero el proceso más complejo fue la verificación de los

resultados obtenidos al aplicar el algoritmo de migración sísmica. Por lo tanto, se realizaron pruebas con diferentes archivos SEG-Y y con diferentes datos iniciales, hasta obtener valores generalizados que se ajusten como predeterminados. Así mismo, se realizaron las pruebas para obtener las imágenes sísmicas antes y después de la migración.

Al final de cada etapa se realizaron pruebas de funcionamiento con el fin de encontrar fallas o puntos de mejora. Las pruebas se realizaron en 2 equipos, uno Windows con procesador Intel i5 y otro Linux Fedora con procesador AMD X2.

3.6 METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE

La metodología que se utilizó para el desarrollo del software es el modelo en espiral, en donde se inició determinando objetivos para poder hacer un análisis de riesgos, es decir un análisis de eventos probables no deseados, luego se pasó a desarrollar las soluciones y a realizar las pruebas, tanto de funcionamiento como de resultados. Con todo esto se realizó una nueva planificación de los requerimientos, para que finalmente se vuelva a iniciar el proceso determinando objetivos, analizando riesgos, etc., de forma cíclica hasta lograr cada una de las metas propuestas.

Se aplicó este modelo ya que se centra en la evaluación y la reducción de riesgos, y parte desde un segmento del objetivo buscado, específicamente planificando el desarrollo de cada librería por separado y verificando su funcionamiento de forma comparativa con algunas aplicaciones similares.

3.7 DESARROLLO DEL MANUAL DE USUARIO

Para el manejo del software se requieren conocimientos básicos sobre sísmica, tales como el manejo de algunos conceptos fundamentales en análisis de datos sísmicos. Sin embargo para la ejecución del software no se requieren conocimientos de

programación, por lo que es necesario contar con un manual adecuado que permita el uso óptimo del software y así se pueda aprovechar adecuadamente.

3.8 ESTUDIO DE BENCHMARKING CON RESPECTO A LOS OTROS SOFTWARE LIBRES EXISTENTES EN EL MERCADO

La comparación entre software libres existentes en el mercado es necesaria para asegurar la calidad del software, por lo tanto se tuvieron en cuenta aspectos como la implementación de una interfaz gráfica, las opciones de migración, la exportación de resultados, la licencia y el manual. En este caso se realizó el análisis con aplicaciones que no solo cumplen con la condición de ser libres, sino que aportan solución a alguna de las características relacionadas con el tema de investigación, ya sea la lectura de archivos SEG-Y, la migración sísmica o la generación de imágenes sísmicas. Dichas aplicaciones se encuentran registradas en la Tabla 1.

4. RESULTADOS DE LA INVESTIGACIÓN

Como producto del resultado de la investigación se desarrolló un software para procesamiento y visualización de datos sísmicos 2D utilizando un algoritmo de técnicas de migración sísmica, con documentación suficiente para facilitar su uso y la comparación tipo benchmarking con otro software comparable para su validación.

4.1 SOFTWARE DESARROLLADO PARA MIGRACIÓN SÍSMICA

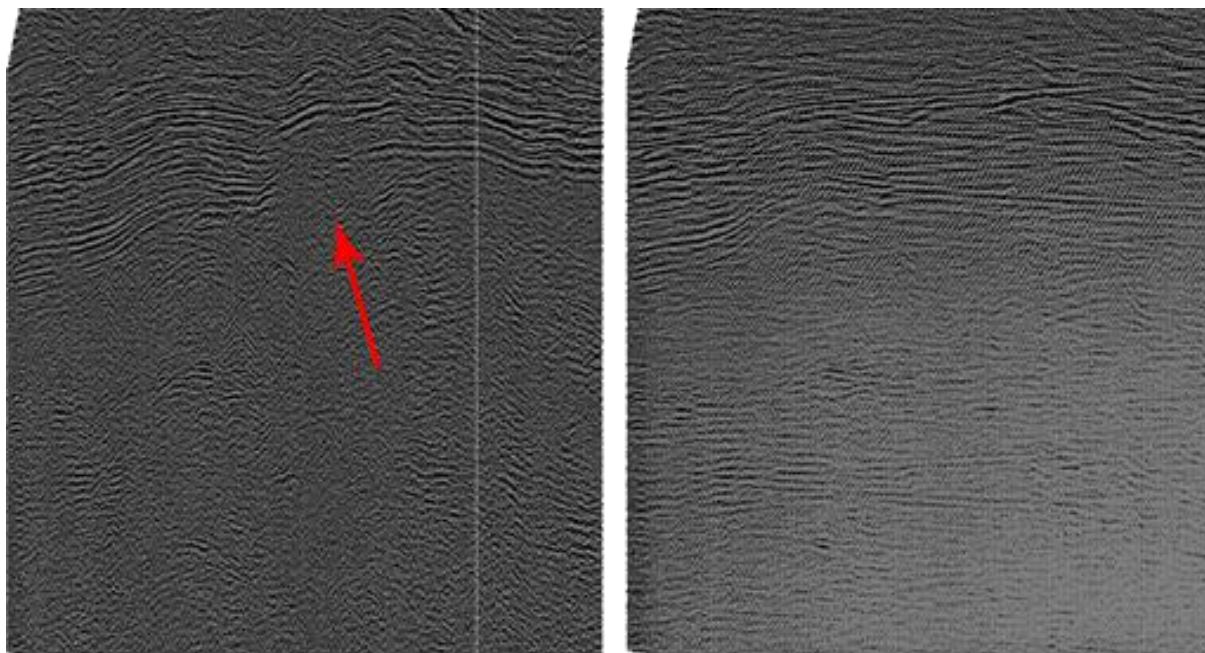
El software resultado de la investigación cuenta con una licencia de software libre GPL v3 que asegura la distribución de su código fuente para que pueda ser estudiado, modificado y redistribuido por quien así lo vea necesario.

El software consta de varias librerías, cuya función se detalla en las siguientes características:

- La lectura de los datos sísmicos que van a ser procesados es tomada desde un archivo binario SEG-Y, de tal forma que durante la lectura se transforman los datos en valores enteros que puedan ser procesados y entendidos fácilmente. Finalmente en forma inversa se realiza la exportación por medio de la escritura de un nuevo archivo SEG-Y con los datos procesados.
- El procesamiento de datos sísmicos 2D consiste en aplicar un algoritmo de migración sísmica por método de Kirchhoff a los datos apilados, por lo se clasificaría como un tratamiento post-apilado de los datos, realizado con el fin de mejorar la calidad de los mismos y resaltar las características relevantes para el análisis de los sismogramas en el campo de los hidrocarburos.
- La creación de las imágenes sísmicas se realiza para los datos originalmente leídos y para los datos que han sido migrados, con el fin de corroborar la calidad de

los resultados de la migración. La presentación de las imágenes se realiza por medio de la pantalla principal de la aplicación, sin embargo serán almacenadas en formato JPG en la misma carpeta del archivo inicial, luego de exportar el proyecto desde el menú de archivo.

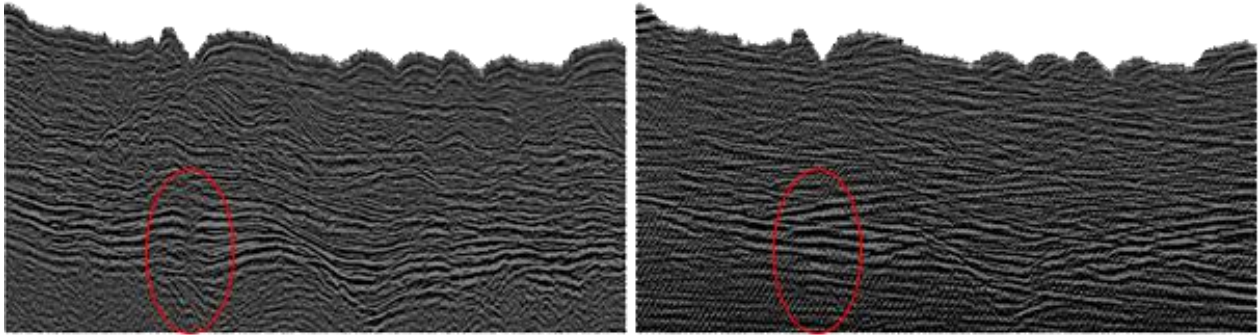
Ilustración 19. A la izquierda la imagen antes y a la derecha después de la migración.



Fuente: Autor.

- Como resultado del proceso de migración se presenta la **Ilustración 19**, en donde se observan al lado izquierdo los datos leídos del archivo SEG-Y sin ningún tipo de mejora y al lado derecho los datos después de aplicar el proceso de migración sísmica; pueden observarse algunos cambios en la mejora de zonas oscuras como se indica con la flecha roja y de igual forma la desaparición de zonas curvas, lo que implica mejoras en la ubicación de puntos reflectores y corrección en los ángulos de buzamiento.

Ilustración 20. A la izquierda la imagen antes y a la derecha después de la migración.



Fuente: Autor.

- Otro resultado obtenido se presenta en la **Ilustración 20**, al igual que el anterior se aprecian a la izquierda los datos sin migrar y a la derecha los datos luego de la migración, pueden apreciarse varios cambios en la reducción de los ángulos de buzamiento, haciendo ver zonas con mayor reflectividad (resaltada en rojo) y también pueden observarse aumentos en las amplitudes de las trazas.

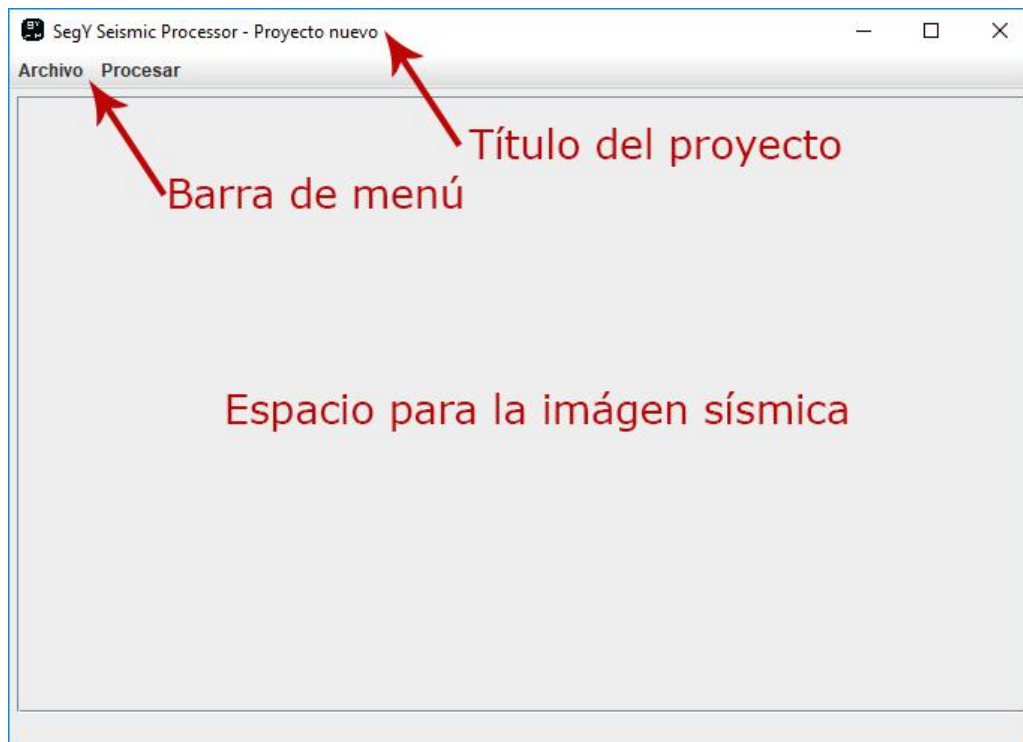
4.2 DOCUMENTACIÓN E INTERFAZ GRÁFICA

La utilización del software no exige conocimientos en programación, de modo que puede ser utilizado por cualquier persona con conocimientos básicos de sismica, por lo que cuenta con una interfaz gráfica sencilla y un manual de usuario suficientemente específico.

En el manual de usuario se encuentra una breve descripción del software, así como la descripción de las opciones, menús y ventanas gráficas utilizadas, cabe resaltar que en todo momento se utilizan imágenes para facilitar su comprensión. El manual se encuentra en formato pdf para facilitar su apertura desde cualquier dispositivo.

La interfaz gráfica desarrollada utiliza menús sobre la ventana principal observada en la **Ilustración 21** que facilitan su comprensión.

Ilustración 21. Interfaz gráfica del software.



Fuente: Autor.

Dentro de las características relevantes en la interfaz grafica existen tres componentes a tener en cuenta:

- El primero consiste en el título del proyecto, con el fin de saber en todo momento el nombre del archivo que se está trabajando se cambia el nombre de la aplicación incluyendo el nombre del archivo importado.
- El segundo es la barra de menú y las opciones que presenta, debido a que las opciones específicas de la migración se encuentran en el menú procesar únicamente.
- La tercera es el espacio reservado para la presentación de la imagen sísmica lograda durante la importación del archivo, que a su vez cambiará luego de la migración por la imagen migrada.

4.3 ANÁLISIS A MODO DE BENCHMARKING

Un aspecto importante de la investigación es el análisis a modo de benchmarking con otro software libre encontrado para el procesamiento de datos sísmicos, comparando aspectos diferenciadores, y con el fin de validar los resultados del software desarrollado, para lo cual se analizaron las siguientes características diferenciadoras:

Sistema Operativo, a pesar de ser un software especializado, puede mejorar su aplicabilidad si permite la instalación multiplataforma, es decir, desde diferentes sistemas operativos.

Tabla 4. Comparación de sistema operativo.

Software	Característica
Madagascar	<ul style="list-style-type: none">• Linux (Fedora, RedHat, Ubuntu, etc.)• Solaris• MacOS X• Windows under the Cygwin environment.
CWP/SU (Seismic Un*x)	<ul style="list-style-type: none">• Linux (Fedora, RedHat, Ubuntu, etc.)
FreeUSP	<ul style="list-style-type: none">• Linux (Fedora)
SEPlib	<ul style="list-style-type: none">• Linux (Fedora, RedHat, Ubuntu, etc.)• Solaris• MacOS X• Windows under the Cygwin environment.
JRG	<ul style="list-style-type: none">• Multiplataforma
Kogeo	<ul style="list-style-type: none">• Windows

Fuente: Autor.

Instalación del software; una instalación adecuada que permita utilizar el software sin complicar el proceso y que evite la instalación de compiladores o paquetes adicionales simplifica no sólo la instalación sino la comprensión del software.

Tabla 5. Comparación de la instalación del software.

Software	Característica
Madagascar	<ul style="list-style-type: none"> • Requiere la instalación de librerías X11, phyton, gcc y git o subversión. • Instalación muy demorada ya que compila cada librería una por una. • Se apoya básicamente en librerías CWP/SU.
CWP/SU (Seismic Un*x)	<ul style="list-style-type: none"> • Requiere la instalación de librerías X11, phyton, gcc y gfortran. • Se debe instalar manualmente cada paquete complementario.
FreeUSP	<ul style="list-style-type: none"> • Requiere la instalación de librerías X11, phyton, gcc y gfortran. • Se debe instalar manualmente cada paquete complementario. • No se actualiza desde 2011 y presenta muchas incompatibilidades.
SEPlib	<ul style="list-style-type: none"> • Requiere la instalación de librerías X11, phyton, gcc y gfortran. • Se debe instalar manualmente cada paquete complementario. • No se actualiza desde 2009 y presenta muchas incompatibilidades.
JRG	<ul style="list-style-type: none"> • No requiere instalación, únicamente requiere tener instalado java JRE. • Tiene un archivo ejecutable por cada función que realiza la aplicación.
Kogeo	<ul style="list-style-type: none"> • La instalación es sencilla, por medio de un instalador de interfaz gráfica.

Fuente: Autor.

Lectura de archivos binarios SEG-Y; es importante que el software se adapte a los estándares existentes para el almacenamiento de datos sísmicos:

Tabla 6. Comparación de la lectura de archivos.

Software	Característica
Madagascar	<ul style="list-style-type: none"> • Permite convertir archivos SEG-Y y SU a su formato RSF apoyado con la librería de conversión que utiliza CWP/SU. • Requiere contar con el encabezado en un archivo diferente para la conversión, lo que complica el posterior uso de los datos.
CWP/SU (Seismic Un*x)	<ul style="list-style-type: none"> • Lee archivos SEG-Y por medio de la conversión a formato SU, sin embargo presenta incompatibilidades con algunos archivos.
FreeUSP	<ul style="list-style-type: none"> • Lee archivos SEG-Y por medio de la conversión a formato USP.
SEPlib	<ul style="list-style-type: none"> • Lee archivos SEG-Y por medio de la conversión.
JRG	<ul style="list-style-type: none"> • Lee archivos SEG-Y de forma nativa.
Kogeo	<ul style="list-style-type: none"> • Permite importar archivos SEG-Y y SU.

Fuente: Autor.

Escritura de archivos binarios SEG-Y; es importante que el software se adapte a los estándares existentes para el almacenamiento de datos sísmicos:

Tabla 7. Comparación de la escritura de archivos.

Software	Característica
Madagascar	<ul style="list-style-type: none"> • Permite convertir archivos RSF a SEG-Y o SU apoyado con la librería de conversión que utiliza CWP/SU.
CWP/SU (Seismic Un*x)	<ul style="list-style-type: none"> • Exporta archivos SU a SEG-Y, sin embargo presenta incompatibilidades al crear el encabezado de algunos archivos.
FreeUSP	<ul style="list-style-type: none"> • Exporta archivos SEG-Y por medio de la conversión.
SEPlib	<ul style="list-style-type: none"> • Exporta archivos SEG-Y por medio de la conversión.
JRG	<ul style="list-style-type: none"> • Exporta archivos binarios de forma nativa.
Kogeo	<ul style="list-style-type: none"> • Permite exportar archivos a formato SEG-Y y SU.

Fuente: Autor.

Migración sísmica; el proceso objeto del proyecto consiste en el mejoramiento de la calidad de las imágenes por medio de la migración sísmica, por lo que se debe tener en cuenta la capacidad de realizar el proceso:

Tabla 8. Comparación del proceso de migración.

Software	Característica
Madagascar	Utiliza varios algoritmos de migración muy funcionales, sin embargo es necesario contar con el análisis de velocidades y el archivo de encabezados para su ejecución.
CWP/SU (Seismic Un*x)	Utiliza varios algoritmos de migración muy funcionales, sin embargo es necesario contar con el análisis de velocidades para su ejecución.
FreeUSP	Aplica las librerías de migración de CWP/SU.
SEPlib	Aplica el proceso de migración pre aplilado y únicamente para datos 3D
JRG	No realiza el proceso de migración sísmica.
Kogeo	No realiza el proceso de migración sísmica.

Fuente: Autor.

Ilustración de sismogramas; el principal elemento para que una persona pueda entender y analizar un proceso sísmico es la imagen sísmica, lo que hace indispensable que el software cuente con dicha capacidad:

Tabla 9. Comparación de la ilustración de sismogramas.

Software	Característica
Madagascar	Permite realizar sismogramas de buena calidad.
CWP/SU (Seismic Un*x)	Permite realizar sismogramas de calidad modificando la escala y los colores.
FreeUSP	Permite realizar sismogramas de buena calidad.
SEPlib	Permite realizar sismogramas de buena calidad.
JRG	Permite realizar sismogramas, sin embargo la calidad no es buena y no conserva la relación de aspecto.
Kogeo	Permite realizar sismogramas de buena calidad, con colores y adicionalmente permite usar la escala.

Fuente: Autor.

Interfaz gráfica; una interfaz amigable o sencilla puede llegar a facilitar el uso del software y de igual manera puede simplificar el proceso de aprendizaje cuando no se tienen muchos conocimientos sobre el tema:

Tabla 10. Comparación de interfaz gráfica.

Software	Característica
Madagascar	No cuenta con una interfaz gráfica.
CWP/SU (Seismic Un*x)	No cuenta con una interfaz gráfica.
FreeUSP	No cuenta con una interfaz gráfica.
SEPLib	No cuenta con una interfaz gráfica.
JRG	Cuenta con una interfaz gráfica sencilla y entendible.
Kogeo	Cuenta con una interfaz gráfica sencilla y entendible.

Fuente: Autor.

Manual de usuario; es necesario contar con un manual lo suficientemente explicado para que el proceso de aprendizaje no pierda su objetivo y el software cumpla su objetivo:

Tabla 11. Comparación del manual de usuario.

Software	Característica
Madagascar	No cuenta con un manual específico, sin embargo cuenta con guías en pdf sobre algunas de sus funciones.
CWP/SU (Seismic Un*x)	Cuenta con un manual que nombra todas las funciones de la aplicación pero no explica todas las propiedades, algo muy necesario ya que no cuenta con interfaz gráfica.
FreeUSP	No cuenta con un manual adecuado, y la documentación de sus funciones se encuentra fuera de servicio.
SEPLib	Cuenta con un manual muy básico, y la mayoría de su documentación en línea se encuentra fuera de servicio.
JRG	No cuenta con un manual adecuado y hay poco soporte de sus funciones en la web.
Kogeo	Cuenta con un manual básico pero muy funcional desde su sitio web.

Fuente: Autor.

Tipo de licencia; la licencia puede limitar o fomentar el desarrollo de aplicaciones, por lo tanto es ideal que se cuente con una licencia libre, que permite la distribución del código y su uso sin restricciones, sobre todo se busca que continúe el desarrollo del software de forma colaborativa:

Tabla 12. Comparación del tipo de licencia.

Software	Característica
Madagascar	GPL
CWP/SU (Seismic Un*x)	Libre, estilo BSD y Abierta
FreeUSP	Abierta
SEPlib	Abierta
JRG	Abierta
Kogeo	GPL

Fuente: Autor.

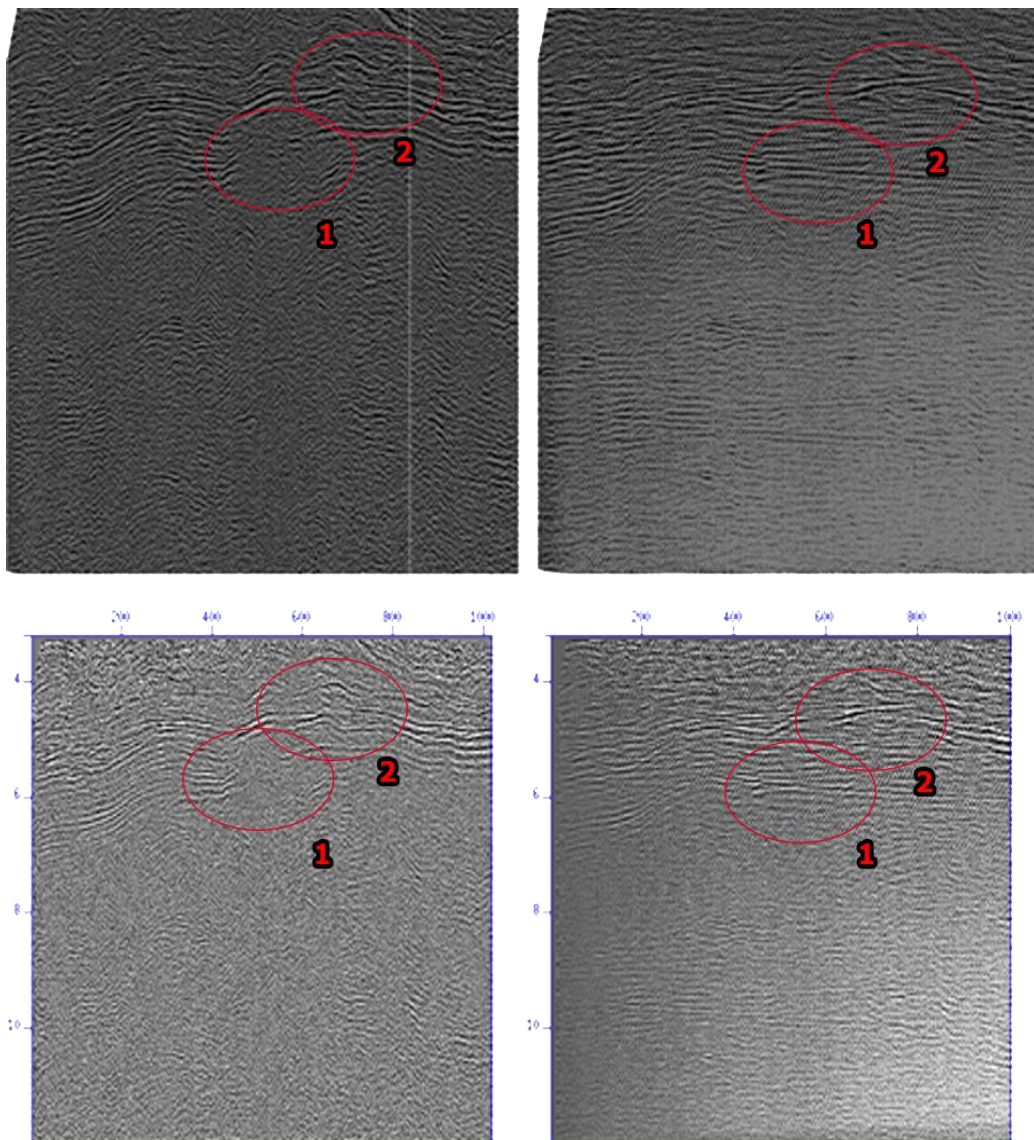
4.4 VALIDACIÓN DEL SOFTWARE

Resultados y tiempo de ejecución; es necesario validar los tiempos de ejecución y los resultados presentados por las aplicaciones que realizan procesos similares, en este caso se realiza la comparación de los resultados de la migración sísmica post apilado, y se observa que únicamente dos aplicaciones realizan el proceso, Madagascar y CWP/SU, sin embargo Madagascar se basa en las librerías de migración que tiene CWP/SU por lo que únicamente se compara con los resultados de éste último.

Para realizar el proceso de migración por medio de CWP/SU, se debe seguir una serie de pasos o líneas de comando; se inicia con la selección de los comandos necesarios para la conversión del archivo SEG-Y, la migración por Kirchhoff y la generación de la gráfica; sin embargo se encuentra que a pesar de contar con muchas opciones de personalización no se cuenta con una compatibilidad completamente eficaz con el formato SEG-Y. Con respecto a la migración se observa que exige el ingreso de datos específicos en la mayoría de casos, ya sea un archivo binario con las líneas de tiempo o uno con el análisis de velocidad, datos que en la mayoría de casos no se encuentran

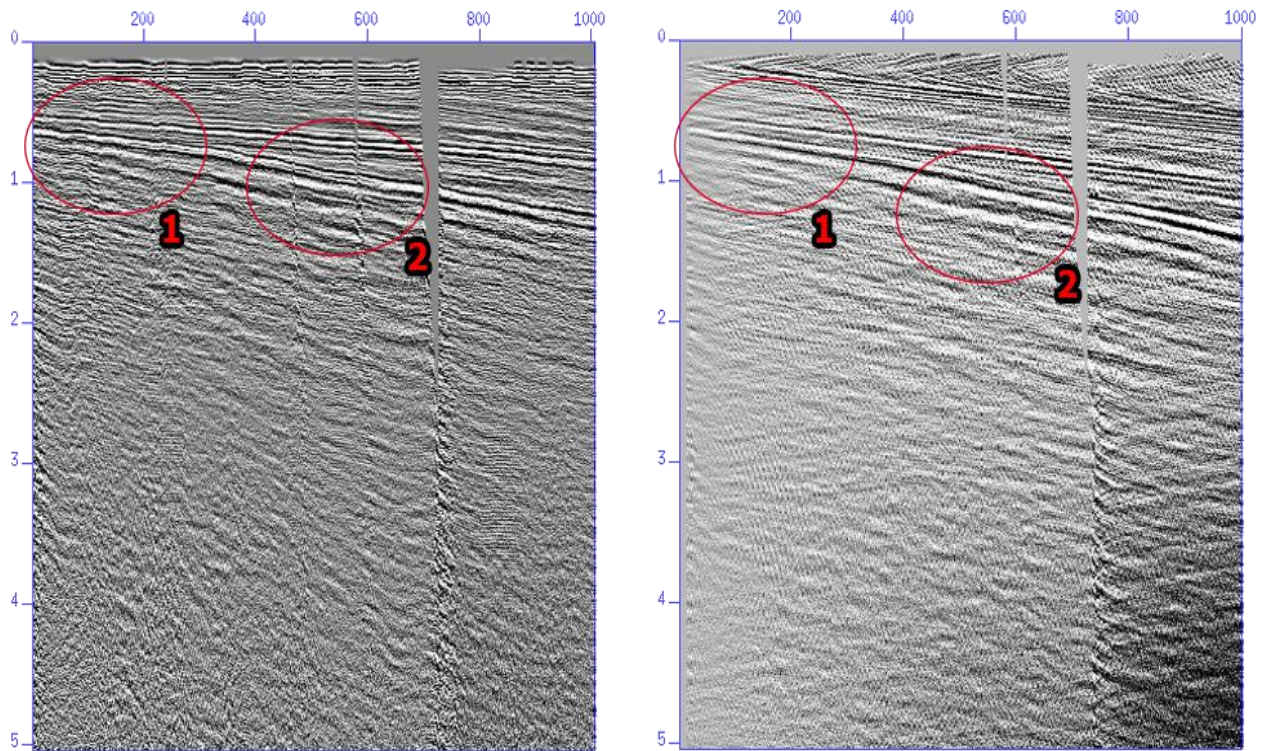
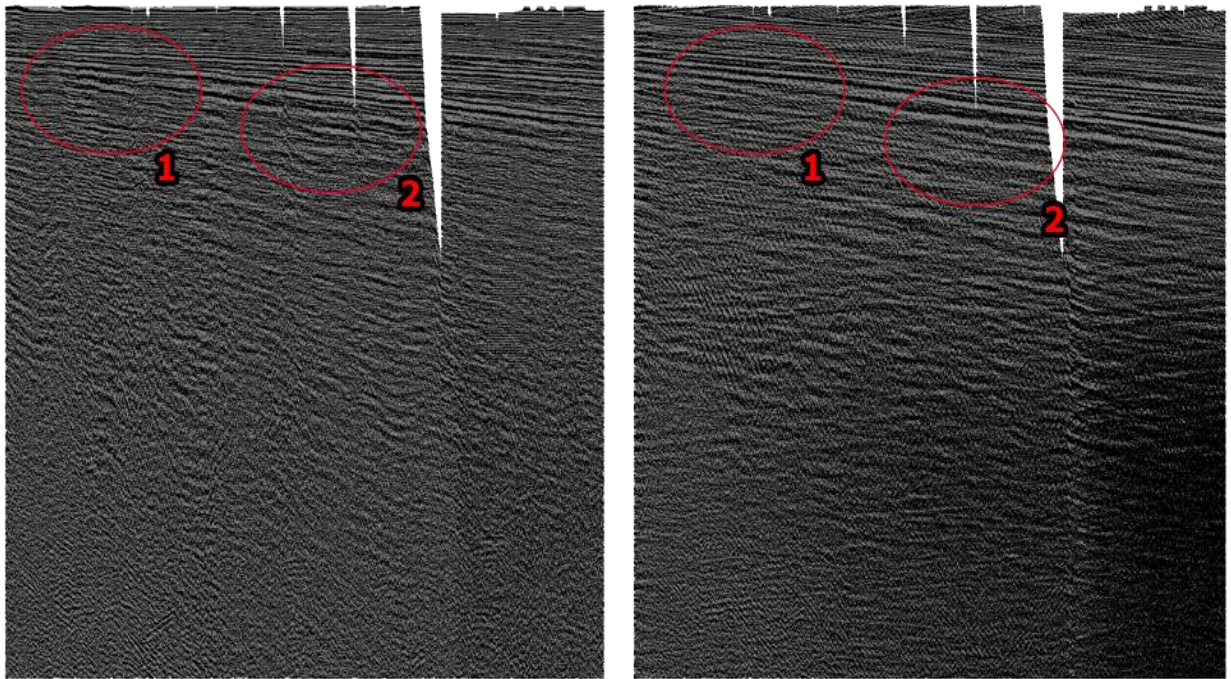
disponibles porque únicamente se genera o sólo se comparte el archivo binario SEG-Y; sin embargo se continúa con el proceso hasta obtener las imágenes, pero se aprecia que los resultados no tienen calidad suficiente para la comparación de resultados, lo que indica que son necesarios en todos los casos los archivos faltantes (cabe resaltar que para obtener resultados de la mejor calidad se debe contar con las tablas de velocidad).

Ilustración 22. Arriba los resultados antes y después con el algoritmo desarrollado, abajo los resultados antes y después de la migración con CWP/SU



Fuente: Autor.

Ilustración 23. Arriba los resultados antes y después con el algoritmo desarrollado, abajo los resultados antes y después de la migración con CWP/SU



Fuente: Autor.

Arriba, en la Ilustración 22 y la Ilustración 23 se presentan las imágenes obtenidas por medio del algoritmo desarrollado, y abajo las imágenes obtenidas del software CWP/SU, a la izquierda luego de su conversión a formato SU y a la derecha luego de la migración.

Pueden apreciarse coincidencias entre las imágenes antes y después de la migración, puesto que las amplitudes de las trazas son similares, así como la ubicación de los puntos reflectores. También se observa en los sectores resaltados 1 y 2, un aumento parecido en los ángulos de buzamiento y zonas con mayor reflectividad.

Para poder aplicar el proceso de migración en CWP/SU se utilizaron los siguientes comandos:

- Convertir el archivo binario SEG-Y a SU por medio de “segypread”.
(Necesario para leer el archivo y exportarlo como SU)
- Crear la tabla de velocidades por medio de “makevel”.
(Necesario para crear y establecer un archivo con las velocidades)
- Crear las tablas de tiempo de viaje por medio de “rayt2d”.
(Basado en la tabla de velocidades y necesario para crear un archivo con datos los tiempos de viaje)
- Aplicar la migración por medio de “sukdmig2d”.
(Requiere del archivo SU, las tablas de tiempos de viaje y velocidades para aplicar el proceso de migración por Kirchhoff post apilado)
- Crear la imagen por medio de “suximage”.
(Presenta la gráfica de los archivos SU que se desea observar)

Finalmente se comparó el tiempo transcurrido en la ejecución del proceso de migración con el algoritmo desarrollado contra el de Seismic Unix. Se observó que el tiempo que duró el proceso de migración tomó de 5 a 7 minutos por cada 1000 trazas en ambos casos.

5. CONCLUSIONES

El algoritmo de migración post apilado es un proceso que favorece al análisis de datos sísmicos cuando se pretende enfocar hacia la prospección de pozos de petróleo. Es posible observar que los resultados obtenidos varían de acuerdo al tipo de datos ingresado, al ángulo de incidencia y a la velocidad media entre otros aspectos generales, y de acuerdo a esto puede que su efecto sea reducido en algunos casos.

Es muy importante que el software tenga la capacidad de leer y escribir archivos SEG-Y para que su aplicación sea efectiva y sea aprovechable; esto porque en la actualidad existen muchas aplicaciones libres que pretenden favorecer el análisis sísmico, sin embargo dichas aplicaciones se encuentran enfocadas a funcionalidades específicas o se enfocan en tomar la alimentación de datos por medio de un formato propio, no uno generalizado o estandarizado.

Según el análisis realizado a las aplicaciones existentes de forma comparativa se pueden resaltar muchas características diferenciadoras que lograrán el éxito del software desarrollado, dentro de ellas se pueden resaltar:

- No requiere instalación, el único requerimiento que tiene es una máquina virtual JAVA, sistema con el que cuentan todos los dispositivos y sistemas operativos.
- Cuenta con una interfaz gráfica que facilita el acceso a todas las funcionalidades del software.
- Permite leer y escribir archivos SEG-Y, un estándar generalizado para el manejo de datos sísmicos.
- Grafica los datos sísmicos antes y después de aplicar el proceso de migración sísmica.

Una característica muy importante a tener en cuenta cuando se utiliza un software para migración sísmica, es asegurarse de que los datos ingresados corresponden a los requeridos en el proceso. Esto quiere decir que existen datos sin procesar, pre apilados con migración pre apilado, apilados, post apilados y con migración post apilados. En este caso se aplica la migración post apilado, lo que implica que los datos ingresados deben estar apilados y no deben haber sido migrados durante el pre apilado. No tener en cuenta esta condición implica que se aplique dos veces el proceso de migración, alterando de forma errónea los resultados del proceso.

Se observa una alta correlación entre los resultados obtenidos con el software desarrollado y los obtenidos con otro software equivalente, lo cual introduce confiabilidad al uso de este software en aplicaciones industriales reales.

Finalmente se aprecia una relación similar en el tiempo que dura el proceso en ambos software, el cual depende directamente del número de trazas a procesar. Sin embargo se observa una gran ventaja del software desarrollado en cuanto a los pasos requeridos por otro software y la complejidad para aplicar el proceso de migración.

6. RECOMENDACIONES Y TRABAJOS FUTUROS

Según el proceso realizado se recomienda continuar el desarrollo de la aplicación con el fin de añadir algunos procesos que complementen y faciliten el análisis de los datos, proceso tales como el apilado, convolución, deconvolución y corrección de velocidad.

De acuerdo al análisis realizado al software existente en forma de benchmarking es posible encontrar varias alternativas para el análisis sísmico, sin embargo todas las alternativas presentan algún tipo de desventaja o limitante; por otro lado la mayoría presenta la ventaja del tipo de licencia utilizado, lo que podría implicar que se retomen las mejores librerías de las aplicaciones para colocarlas dentro de una aplicación para que cumpla con las expectativas y calidad esperada.

BIBLIOGRAFÍA

- Abreo Carrillo, S. A., & Ramírez Silva, A. B. (2010). Viabilidad de acelerar la migración sísmica 2D usando un procesador específico implementado sobre un FPGA. *INGENIERÍA E INVESTIGACIÓN*, 64-70.
- Amoco Production Company. (2011). *FreeUSP (Versión 1.05-2){Software}*. Obtenido de: <http://www.freeusp.org/FreeUSP.html>.
- Bancroft, J. C. (2002). *A visual relationship between Kirchhoff migration and seismic*. Calgary, Canada: CREWES Research Report - Volume 14.
- Bancroft, J. C. (2002). Seismic Imaging: Prestack. *Recorder(Official publication of the Canadian Society of Exploration Geophysicists)*, 09-33.
- Benítez R., H. S. (2012). *Evaluación de métodos de migración sísmica antes del apilamiento en profundidad basados tanto en la integral de Kirchhoff como en la ecuación de onda en datos reales (Proyecto Oro Negro 07G 3D) (tesis de pregrado)*. Caracas, Venezuela: Universidad Central de Venezuela.
- Bonomi, E., & Cabitza, G. (1994). Migration of seismic data. *Journal of Statistical Physics*, 703-723.
- Chávez Pérez, S. (1997). *Enhanced Imaging of Fault Zones in Southern California From Seismic Reflection Studies*. Reno, Nevada: University of Nevada.
- Colorado School of Mines. (2007). *CWP/SU: Seismic Unix (Version 44) {Software}*. Obtenido de: <http://www.cwp.mines.edu/cwpcodes/index.html>.
- Fajardo, C., Castillo Villar, J., & Pedraza, C. (2013). Reducción de los tiempos de cómputo de la Migración Sísmica usando FPGAs y GPGPUs: Un artículo de revisión. *Ingeniería y Ciencia*, 261–293.
- Forbriger, T. (2015). *Seitosh (Versión 1.0) {Software}*. Obtenido de: <https://git.scc.kit.edu/Seitosh/Seitosh>.
- Fundación Wikimedia, Inc. (07 de 04 de 2016). *wikipedia*. Recuperado el 09 de 05 de 2016, de https://es.wikipedia.org/wiki/Computaci%C3%B3n_paralela#cite_note-1
- Gazdag, I., & Sguazzero, P. (1984). Migration of Seismic Data. *Proceedings of the IEEE*, 1302-1315.

- Griem, W. (15 de 07 de 2015). *Museo Virtual, Geología*. Recuperado el 28 de 05 de 2016, de <http://www.geovirtual2.cl/geologiageneral/ggcap01c.htm>
- Herrera Rivero, C. M. (2005). *Secuencia básica de procesamiento sísmico de datos complejos 2D, línea ME-90B-27, Maturín, EDO. Monagas (Tesis pregrado)*. Valle de Sartenejas. Venezuela: Universidad Simón Bolívar.
- Herrera, Y., & Cooper, N. (2010). *Manual para la adquisición y procesamiento de sísmica terrestre y su aplicación en Colombia*. Bogotá, Colombia: Universidad Nacional de Colombia.
- Jones, I. (2014). Tutorial: migration imaging conditions. *First break*, 45-55.
- Korenaga, J. (2003). *Tomo2D (Versión 1.4.3) {Software}*. Obtenido de: <http://people.earth.yale.edu/software/jun-korenaga>.
- Linares Moreno, C. A. (2014). *Paralelización de algoritmos en plataformas distribuidas: caso de estudio en procesamiento sísmico en la industria petrolera. (Tesis de maestría)*. Bogotá D.C.: Pontificia Universidad Javeriana.
- Marthelot, J.-M. (2015). *La geofísica internas y sus aplicaciones*. Recuperado el 27 de 05 de 2016, de <http://saxifra.u-strasbg.fr/Sismique/isis.html#PS>
- Martínez Macedo, R. M. (2008). *Aplicaciones de Exploración y Producción de Petróleo y Gas en Plataformas Cluster Heterogéneas y Grid (Tesis pregrado)*. Sartenejas, Venezuela: Universidad Simón Bolívar.
- Multiple free proposals. (2017). *Madagascar (Versión 2.0) {Software}*. Obtenido de: <http://www.ahay.org>.
- Olano, K. K., & Suarez, R. (23 de 11 de 2014). *Geología*. Recuperado el 26 de 05 de 2016, de <http://keygeolo.blogspot.com.co/2014/11/semana-12.html>
- OpendTect OSR team. (2018). *Opendtect (Versión 6.2.3){Software}*. Obtenido de: <https://opendtect.org/osr/>.
- Real academia de ciencias exactas, físicas y naturales. (26 de 01 de 2016). *RACEFN Glosario de Geología*. Recuperado el 08 de 05 de 2016, de http://www.ugr.es/~agcasco/personal/rac_geologia/rac.htm
- Sacchi, M. D., Velis, D. R., & Cominguez, A. H. (1994). Minimum entropy deconvolution with frequency-domain constraints. *Geophysics*, 938-945.
- Schlumberger Limited. (12 de 06 de 2006). *Oilfield Glossary en Español*. Recuperado el 08 de 05 de 2016, de <http://www.glossary.oilfield.slb.com/>

- SEG Technical Standards Committee. (2017). *SEG-Y_r2.0: SEG-Y revision 2.0 Data Exchange format*. Houston, Texas: Society of Exploration Geophysicists.
- Stanford Exploration Project. (18 de 11 de 1997). *SEP*. Recuperado el 18 de 12 de 2017, de http://sepwww.stanford.edu/public/docs/sep73/jon1.kirch/paper_html/
- Stanford University. (2008). *Seplib (Versión 6.5.3){Software}*. Obtenido de: <http://sepwww.stanford.edu/doku.php?id=sep:software:seplib>.
- Teran Julio, G. E., & Pita Escamilla, J. R. (2009). *Evaluación del algoritmo de migración de Kirchhoff en 2D en un clúster de procesadores*. Bucaramanga: Universidad Industrial de Santander.
- University of Hamburg. (2005). *Kogeo (Versión 2.7){Software}*. Obtenido de: <http://www.kogeo.de/index.htm>.
- University of Nevada, Reno. (2016). *JRG (Versión 4.3.3){Software}*. Obtenido de: <http://crack.seismo.unr.edu/jrg/>.
- University of South Carolina. (2013). *Taup (Versión 2.1.2) {Software}*. Obtenido de: <http://www.seis.sc.edu/TauP/>.
- Vásquez, J. A. (2010). *Análisis de diferentes algoritmos de migración post-apilamiento sobre datos sísmicos bidimensionales al oeste del estado barinas (Tesis pregrado)*. Caracas, Venezuela: Universidad Central de Venezuela.
- Veritas Caspian LLP . (2010). *Veritas Caspian Web*. Recuperado el 27 de 05 de 2016, de <http://www.veritas-caspian.kz/pages/70/>
- Wong, M., & Terwilliger, R. U. (2012). *Lab 4: Kirchhoff migration (Fortran version)*. Stanford, California: Stanford University.

ANEXOS

Anexo A. Código de subrutina en la que se aplica la solución a la migración por la ecuación de Kirchhoff, desarrollado en lenguaje Fortran, Fuente: (Wong & Terwilliger, 2012).

```
module Kirchhoff
```

```
implicit none
integer, private                :: nt, nx
real, private                   :: amax, apermax, t0, dt, dx
logical, private                :: isamax
real, dimension(:), allocatable, private :: velhalf
```

```
contains
```

```
subroutine kirchfast_init(velhalf_in, t0_in, dt_in, dx_in, nt_in, nx_in, isamax_in)
integer                :: nt_in, nx_in
real                   :: t0_in, dt_in, dx_in
logical                :: isamax_in
real, dimension(:), allocatable :: velhalf_in
nt = nt_in
nx = nx_in
t0 = t0_in
dt = dt_in
dx = dx_in
isamax = isamax_in
allocate(velhalf(nt))
velhalf = 0.
velhalf = velhalf_in
end subroutine
```

```
subroutine update_limit(amax_in, apermax_in)
real                :: amax_in, apermax_in
amax = amax_in
apermax = apermax_in
end subroutine
```

```
subroutine kirchfast(adj, add, modl, data)
logical, intent(in)                :: adj, add
real, dimension(:,:)               :: modl, data
integer                             :: ix, iz, it, ih, xstart, xend
real                                :: amp, t, z, h
```

```

if(.not. add) then
  if(adj) then
    modl = 0.
  else
    data = 0.
  endif
endif

do ih=-nx,nx
  h = dx * ih ! h = offset
  do iz= 2, nt
    z = t0 + dt * (iz-1) ! z = travel-time depth
    t = sqrt( z**2 + (h/velhalf(iz))**2 )
    it = 1.5 + (t - t0) / dt
    if(it > nt) then
      exit
    endif
    amp = (z / t) * sqrt( nt*dt / t )
    xstart = 1-ih
    xstart = max0( 1, xstart)
    xend = nx-ih
    xend = min0( nx, xend)
    if(.not. adj) then
      do ix= xstart, xend
        data(it,ix+ih) = data(it,ix+ih) + modl(iz,ix)*amp
      enddo
    else
      do ix= xstart, xend
        modl(iz,ix) = modl(iz,ix) + data(it,ix+ih)*amp
      enddo
    endif
  enddo
enddo

end subroutine
end module

```

Anexo B. Código del algoritmo de migración desarrollado, se basa en el algoritmo de Jon F. Clarebout, publicado en 1997 (Stanford Exploration Project, 1997):

```

public void kirchhoffMigration () {

    // Crea y asigna valores a las variables
    double x0=0.0, y0=0.0;
    double dx=dx, z0=t0, dz=dt;
    int nz=nt;
    double x, y, z, hs, t;
    int it;
    data = new int[nt][nx];

    //Inicia con las iteraciones según el número de trazas y datos por traza
    for (int ix=0; ix < nx; ix++) {

        //Asigna el valor de x para el primer nivel de iteraciones.
        x = x0 + dx * ix;
        for (int iy=0; iy < nx; iy++) {

            // Asigna el valor de y para el segundo nivel de iteraciones.
            y = y0 + dy * iy;
            for (int iz=0; iz < nz; iz++) {

                // Asigna el valor de z para calcular y evaluar it.
                z = z0 + dz * iz;
                hs = (x-y) / velhalf;
                t = Math.sqrt(z*z + hs*hs);
                it = (int)((0.5+(t-t0)/ dt);

                //Verifica si el valor de it es válido
                if (it < nt) {

                    //Verifica si el valor de la posición del dato no es nulo
                    if (oldData[it][iy]!=0){

                        //Asigna el Nuevo valor del dato con el fin de que funcione la sumatoria de Kirchhoff
                        data[it][iy] = (int) (data[it][iy] + oldData[iz][ix]);
                    }
                }
            }
        }
    }
}

```