

Imagen sísmica en profundidad: comparación de migración poststack por ecuación de onda en frecuencia y migración de Kirchhoff

Depth Seismic Imaging: a comparison between poststack wave equation migration and Kirchhoff migration

Lucía Páez Gayone¹, Juan I Sabbione^{2,3}, Tomás D'Biassi⁴

Resumen La migración sísmica es un proceso primordial para obtener una imagen del subsuelo fidedigna en áreas geológicamente complejas con estratos buzantes, fallas y/o pliegues estructurales. Dos de las técnicas más utilizadas en la industria para migrar datos sísmicos son la migración por ecuación de onda y el método de Kirchhoff. En este trabajo comparamos dos algoritmos de migración en profundidad aplicados sobre conjuntos de datos *poststack*. Uno de estos algoritmos es el método de *split-step*, basado en la continuación descendente del campo de propagaciones contemplando variaciones laterales de velocidad. El segundo algoritmo es la migración de Kirchhoff, que, aunque parte de la ecuación de onda, se enfoca en sumar las amplitudes observadas a lo largo de los arribos dados por las curvas de difracción. El método de *split-step* se implementa mediante un código propio en Julia, mientras que la migración de Kirchhoff se realiza a través del software comercial EPOS™. La motivación del estudio es analizar en qué escenarios resulta más conveniente utilizar un método u otro. Para ello, se procesan tres conjuntos de datos *poststack* mediante ambos métodos: (1) un conjunto de datos sintéticos generados a partir de un modelo de velocidad realista que incluye pliegues, zonas con buzamiento y una falla estructural; (2) un conjunto de datos reales de la Cuenca Austral, Tierra del Fuego; y (3) un conjunto de datos reales de la zona de Faja Plegada en la Cuenca Huallaga, Perú, caracterizada por grandes buzamientos. Los resultados muestran que, en general, ambos métodos generan una imagen correcta de las secciones sísmicas para los conjuntos de datos utilizados. No obstante, dada la naturaleza de los métodos, *split-step* reconstruye de forma más nítida los estratos con mayores buzamientos mientras que Kirchhoff resuelve mejor los problemas de borde y produce un resultado levemente superior en presencia de fallas.

Palabras clave Migración poststack, ecuación de onda, método de Kirchhoff, método de *split-step*.

Abstract Seismic migration is a key process for obtaining reliable subsurface images in geologically complex areas containing dipping layers, faults, and structural folds. Two of the most commonly used techniques in the industry for migrating seismic data are wave-equation migration and Kirchhoff migration. There is a large number of methods and variants for both techniques. In this study, we analyze and compare two depth migration algorithms applied to poststack data sets. The first algorithm is the *split-step* method, which is based on downward continuation of the wavefield while accounting for lateral velocity variations. The second algorithm is Kirchhoff migration, which, although also derived from the wave equation, focuses on summing observed amplitudes along arrivals defined by diffraction curves. The *split-step* method is implemented using a self-developed code in Julia, while Kirchhoff migration is implemented through the commercial software EPOS. This study aims to analyze in which scenarios it is more convenient to use one method over the other. For this purpose, three poststack data sets are processed using both methods: (1) a synthetic data set generated from a realistic velocity model that includes folds, areas with significant dips, and a structural fault; (2) a real data set from the Austral Basin acquired in Tierra del Fuego; and (3) a second real data set from the Fold and Thrust

¹Facultad de Ciencias Astronómicas y Geofísicas, Universidad Nacional de La Plata, Argentina. Email: lucia.paez17@gmail.com

²Centro de Investigaciones Geofísicas (CIGEOF), Universidad Nacional de La Plata, Argentina.

³Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina.

⁴YPF, Argentina.

Belt in the Huallaga Basin, Peru, characterized by large dips. The results show that, overall, both methods successfully generate reliable images of the seismic sections for the three poststack data sets. However, given the nature and approach of each method, the split-step method reconstructs steeper-dipping layers more sharply. Meanwhile, the Kirchhoff method better addresses edge effects and produces slightly superior results in areas with faults.

Keywords Poststack migration, wave equation, Kirchhoff migration, split-step migration.

INTRODUCCIÓN

En la exploración y desarrollo de hidrocarburos, la migración sísmica constituye el proceso que permite colapsar la energía difractada que fue medida en superficie a su verdadera posición en el subsuelo (Dobrin y otros, 1988). Su objetivo se basa en revertir los efectos de la propagación de las ondas que emergen de los reflectores en profundidad. En otras palabras, la migración permite estimar los coeficientes de reflexión en sus posiciones correctas en el subsuelo reubicando los reflectores desde su posición aparente a su posición verdadera (Bjørlykke, 2015). Esto resulta fundamental para poder realizar luego una adecuada interpretación estructural de la geología de la zona prospectada y una correcta caracterización sísmica del reservorio.

Existen diferentes métodos que pueden ser utilizados para migrar una sección sísmica. La elección del método a utilizar dependerá de varios factores, pero especialmente del objetivo de la prospección y de las características del subsuelo. Las técnicas de migración en tiempo son apropiadas en escenarios donde la variación lateral de la velocidad es suave. En aquellos casos donde las variaciones laterales de velocidad sean significativas, es conveniente emplear métodos de migración en profundidad (Yilmaz, 2001). A su vez, ante marcos geológicos sencillos la migración puede aplicarse luego del stack o apilado de los datos (migración *poststack*), en tanto que la migración *prestack* producirá mejores resultados para escenarios más complejos, a expensas de un mayor costo computacional.

Una gran variedad de técnicas de migración parten de la resolución de la ecuación de onda. Entre ellas, existe un conjunto de algoritmos basados en la extrapolación del campo de ondas y la reconstrucción del campo de desplazamiento. Estos métodos pueden ser resueltos en el dominio del tiempo, de la frecuencia, o en una combinación de ambos. Se los conoce como migración por ecuación de onda. Un corrimiento en tiempo o en profundidad se traduce en un cambio de fase en el dominio de la frecuencia. Entre los métodos tradicionales, este es el enfoque adoptado por la migración *phase shift*, donde las operaciones se llevan a cabo en el dominio de la frecuencia. Así, se puede calcular el campo de ondas en profundidad descendiendo las posiciones de fuente y receptor iterativamente a partir de un corrimiento de fase en frecuencia (Gazdag, 1978). No obstante, cuando la velocidad varía lateralmente de forma no despreciable, se deben introducir correcciones a los métodos de cambio de fase para que contemplen dichas variaciones. Una solución posible es extrapolar el campo de ondas a través de modelos de velocidad lateralmente uniformes utilizando dos velocidades diferentes e interpolar el resultado para los valores intermedios. Esta estrategia define al método denominado *phase shift plus interpolation (PSPI)* (Gazdag y otros, 1984). Alternativamente, se puede calcular la corrección que debe introducirse a partir de las diferencias existentes entre la lentitud promedio de una capa y el valor real que contempla su variación lateral. Este último enfoque motiva el método de *split-step* (SS) (Stoffa y otros, 1990). El método de *split-step* se aplica tanto en el dominio frecuencia–número de onda como en el dominio de frecuencia–espacio. De forma resumida, en el método de SS se define una lentitud de referencia (la media en el intervalo de migración) y un término de perturbación que varía espacialmente. Utilizando un número de onda vertical de referencia en el dominio frecuencia–número de onda, se realiza la extrapolación descendente en el intervalo, mientras que el término de perturbación se utiliza para hacer un segundo corrimiento de fase en el dominio frecuencia–espacio.

Por otro lado, la migración por el método de Kirchhoff se destaca por ser una de las más utilizadas en

la industria ya que es un método que funciona muy bien en diferentes escenarios. La migración de Kirchhoff también parte de la ecuación de onda, pero su enfoque es a partir de su solución. Es un método integral debido a que emplea la integración para resolver la ecuación de onda (Schneider, 1978). A partir del principio de Huygens, se busca colapsar las difracciones y reposicionar los reflectores en sus posiciones correctas. La migración de Kirchhoff colapsa las difracciones sumando las amplitudes a lo largo de la curva de difracción y enfoca la energía en su ubicación de origen (Onajite, 2014). Previo a la suma, deben tenerse en cuenta diversos factores como la oblicuidad, la divergencia esférica y características de fase y amplitud (Yilmaz, 2001). Además, se reconstruyen la amplitud y la fase para que respondan de manera más exacta a la reflectividad.

La gran mayoría de los métodos de migración sísmica han sido rigurosamente analizados y explorados en la literatura (Gray y otros, 2001). Las compañías de servicios geofísicos y las compañías petroleras suelen contar con una suite de software comercial de procesamiento de datos sísmicos. Tales paquetes de procesamiento cuentan con manuales de uso, recomendaciones para la selección de parámetros, y demás versatilidades. Adicionalmente, los operadores especializados realizan diferentes chequeos y controles de calidad. No obstante, no es una práctica común recurrir a desarrollos académicos para realizar controles de calidad y/o comparaciones. Este último punto representa la motivación del trabajo. El objetivo es doble: por un lado, se quiere comparar el desempeño del método de *split-step* de desarrollo propio versus el software comercial de procesamiento que emplea el método de Kirchhoff más comúnmente utilizado en la industria para migrar imágenes sísmicas. Por otro lado, y partiendo de los buenos resultados obtenidos mediante ambos métodos, se analiza cuál de ellos debiese ser utilizado según las características de los datos a procesar.

A continuación se describen brevemente los fundamentos teóricos de cada uno de los dos métodos utilizados: migración *split-step* y migración de Kirchhoff. Dado que para el método de migración de *split-step* se implementó un código de desarrollo propio, se brindan más detalles del algoritmo en este primer caso. Luego, se presentan los tres conjuntos de datos utilizados: un conjunto de datos *poststack* sintéticos que resulta de utilidad para controlar y entender la selección de parámetros, y dos conjuntos de datos reales, también *poststack*, que permiten evaluar las potenciales virtudes y debilidades de cada estrategia en aplicaciones reales. Seguido de ello, se describen los parámetros fundamentales para el procesamiento utilizados para cada método, incluyendo un análisis respecto de cómo implementar la condición de borde en el método de *split-step*, y los valores de apertura y filtro antialias utilizados en la migración por el método de Kirchhoff. Luego se presentan y describen los resultados para cada método, y finalmente se enumeran las principales conclusiones obtenidas a partir de los resultados.

TEORÍA

Migración *split-step*

La migración de *split-step* parte del algoritmo de corrimiento de fase (*phase shift*) propuesto originalmente por Gazdag (1978) y se enfoca adicionalmente en corregir los efectos de cambios laterales de velocidad en cada intervalo de migración. Los cambios laterales de velocidad son tratados como una perturbación en el campo de desplazamientos, y en consecuencia demandan el cómputo de una transformada de Fourier espacial extra para cada extrapolación (Stoffa y otros, 1990).

En un dominio bidimensional (x, z), donde x denota el offset y z la profundidad, la ecuación de onda para las presiones p se puede escribir en un punto medio M entre fuente y receptor como:

$$\frac{\partial^2 p}{\partial x_M^2} + \frac{\partial^2 p}{\partial z_M^2} = \frac{1}{(c/2)^2} \frac{\partial^2 p}{\partial t^2}, \quad (1)$$

donde c denota la velocidad. Llamando u a la lentitud intervállica de ida y vuelta dada por el doble de

la inversa de la velocidad se tiene $u(x, z) = 2/c(x, z)$. Reemplazando en la ecuación anterior:

$$\frac{\partial^2 p}{\partial x_M^2} + \frac{\partial^2 p}{\partial z_M^2} = u^2 \frac{\partial^2 p}{\partial t^2}. \quad (2)$$

En el punto medio M podemos dividir la lentitud en dos términos:

$$u(x_M, z_M) = u_o(z_M) + \Delta u(x_M, z_M), \quad (3)$$

donde $u_o(z_M)$ representa la lentitud de referencia constante y $\Delta u(x_M, z_M)$ es el término de perturbación. Es decir, $u_o(z_M)$ es la media utilizada para migrar en el intervalo en el dominio frecuencia–número de onda (análogo al método de *phase shift* con velocidad constante), que no depende del offset; y $\Delta u(x_M, z_M)$ es la corrección aplicada en el dominio espacio–frecuencia como un segundo cambio de fase. Transformando ahora al dominio de la frecuencia angular ω a partir de la transformada de Fourier en tiempo y descomponiendo el campo de lentitud a partir de la [ecuación 3](#), la ecuación de onda puede escribirse como:

$$\begin{aligned} \frac{\partial^2 P}{\partial x_M^2} + \frac{\partial^2 P}{\partial z_M^2} + \omega^2 u_o^2 P &= -\omega^2 (2u_o \Delta u + \Delta u^2) \\ &= -S(x_M, z_M, \omega) \end{aligned} \quad (4)$$

Esta expresión representa una ecuación inhomogénea que incluye un término de fuente S debido a las variaciones en la lentitud. Para hallar la solución de la [ecuación 4](#) se ignora el término Δu^2 porque se considera despreciable frente al término que es lineal en Δu .

El algoritmo de migración de *split-step* puede sintetizarse como sigue. Primero se transforma el dato medido al dominio de la frecuencia angular ω a partir de la transformada de Fourier en tiempo. La profundidad se discretiza utilizando un intervalo Δz constante y se avanza iterativamente en profundidad con un paso Δz en cada iteración. Dado el campo migrado a profundidad z_n , $P(x_M, z_n, \omega)$, para hallar el campo migrado a profundidad z_{n+1} se deben realizar los siguientes pasos:

1. Transformar el campo ascendente previamente migrado a profundidad z_n del dominio del espacio x_M al dominio del número de onda lateral k_x .

$$P(x_M, z_n, \omega) \xrightarrow{\mathcal{F}_x} \tilde{P}(k_x, z_n, \omega),$$

donde \mathcal{F}_x representa la transformada de Fourier espacial (en x) al espacio k_x :

2. Aplicar un corrimiento de fase de Δz en el número de onda vertical k_{z_0} a partir de la lentitud de referencia media u_o en el intervalo $[z_n, z_{n+1}]$ para todas las frecuencias y números de onda:

$$\tilde{P}_1(k_x, z_n, \Delta z, \omega) = \tilde{P}(k_x, z_n, \omega) e^{i k_{z_0} \Delta z},$$

donde

$$k_{z_0} = \sqrt{\omega^2 u_o^2 - k_x^2} = \omega u_o \sqrt{1 - (k_x / \omega u_o)^2}.$$

3. Volver al dominio del espacio x_M a partir de la transformada de Fourier inversa sobre el campo al que se le aplicó el *phase shift*:

$$\tilde{P}_1(k_x, z_n, \Delta z, \omega) \xrightarrow{\mathcal{F}_x^{-1}} P_1(x_M, z_n, \Delta z, \omega), \quad (5)$$

donde \mathcal{F}_x^{-1} representa la transformada de Fourier espacial inversa.

4. Aplicar un segundo corrimiento de fase debido a la perturbación de la lentitud $\Delta u = u(x_M, z) - u_o(z)$

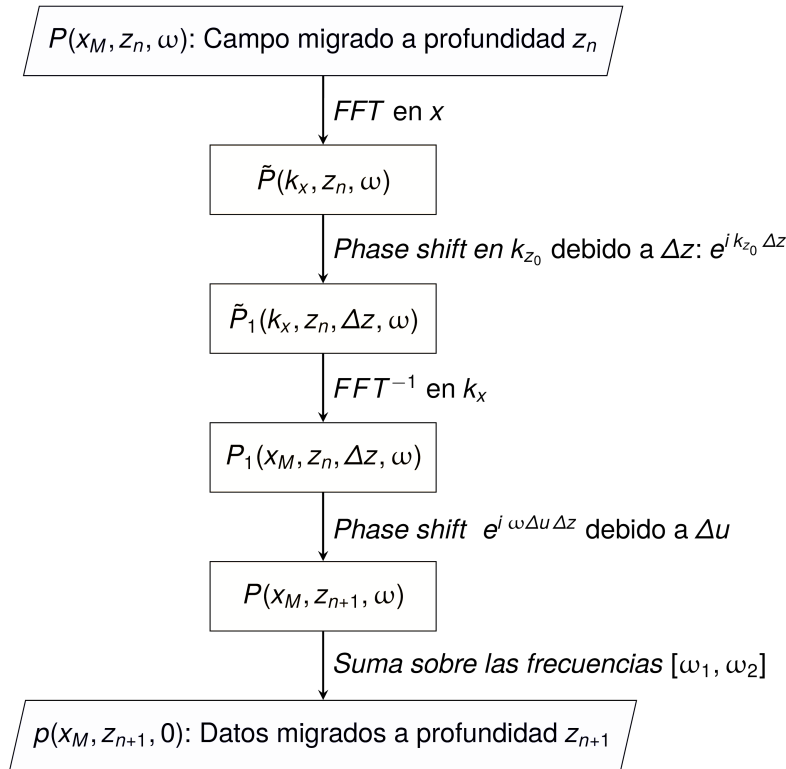


Figura 1. Diagrama de flujo esquemático del método de split-step.

en el intervalo $[z_n, z_{n+1}]$:

$$P(x_M, z_{n+1}, \omega) = P_1(x_M, z_n, \Delta z, \omega) e^{i \omega \Delta u \Delta z}. \quad (6)$$

5. Sumar $P(x_M, z_{n+1}, \omega)$ sobre todas las frecuencias de interés $[\omega_1, \omega_2]$ para finalmente obtener los datos migrados para la profundidad z_{n+1} :

$$p(x_M, z_{n+1}, 0) = \sum_{\omega=\omega_2}^{\omega=\omega_1} P(x_M, z_{n+1}, \omega). \quad (7)$$

Este proceso se repite iterativamente en Z de forma de avanzar en profundidad Δz en cada iteración. El diagrama de flujo de la [Figura 1](#) esquematiza el algoritmo utilizado.

Migración Kirchhoff *poststack*

La migración Kirchhoff se basa en sumar a lo largo de las curvas de difracción que describen las trayectorias de los rayos sísmicos. La migración *poststack* se realiza después de apilar las trazas en cada punto medio común (CMP). La imagen migrada en cualquier punto del subsuelo se construye sumando las contribuciones de todas las trazas sísmicas y utilizando los tiempos de viaje correctos definidos por las hipérbolas de difracción que representan el tiempo de viaje entre un punto medio común en la superficie y un punto en el subsuelo. Es decir, la ecuación de migración suma los datos sísmicos en los tiempos correspondientes al tiempo de viaje de ida y vuelta desde el punto de imagen (x, z) hasta el punto medio en la superficie $(x_M, 0)$.

Además, debe utilizarse una función de peso que contemple la dispersión geométrica, el factor de oblicuidad y cualquier otra corrección de amplitud necesaria para compensar los efectos de la propagación de ondas y la geometría de adquisición. Por lo tanto, la imagen migrada puede expresarse como:

$$I(x, z) = \sum_{x_M} d(x, t = t_{hyp}(x_M, x, z)) W(x_M, x, z), \quad (8)$$

con

$$t_{hyp}(x_M, x, z) = \frac{2}{v} \sqrt{(x_M - x)^2 + z^2}, \quad (9)$$

donde:

- $I(x, z)$ es la imagen migrada en la ubicación (x, z) ,
- $d(x, t)$ son los datos sísmicos *poststack*; es decir, los datos registrados en superficie en función del punto medio común $(x_M, 0)$ y el tiempo t ,
- $W(x_M, x, z)$ es una función de peso que tiene en cuenta factores como la dispersión geométrica, el factor de oblicuidad y cualquier otra corrección de amplitud necesaria para compensar los efectos de la propagación de ondas y la geometría de adquisición.
- $t_{hyp}(x_M, x, z)$ representa el tiempo de viaje de ida y vuelta entre el punto medio común $(x_M, 0)$ y el punto de imagen en el subsuelo (x, z) . El tiempo t_{hyp} se calcula utilizando la [ecuación 9](#), asumiendo una velocidad constante v .

La suma se realiza sobre todos los puntos medios comunes x_M . En algunas implementaciones simples, la función de peso puede ser uniforme, o puede determinarse mediante un análisis más avanzado del campo de ondas. La imagen migrada final se obtiene sumando las contribuciones de todas las trazas de datos *poststack* a lo largo de sus correspondientes curvas de difracción. El proceso de migración reconstruye los reflectores enfocando la energía de las hipérbolas de difracción. La suma sobre todos los puntos medios comunes asegura que la energía proveniente del verdadero punto de reflexión interfiera constructivamente, mientras que la energía de otros puntos se cancela.

La migración Kirchhoff se puede implementar tanto en los dominios de tiempo como de profundidad. Para la migración en profundidad, típicamente se requiere un modelo de velocidad para calcular los tiempos de viaje con precisión. La forma de la difracción puede asumirse como hiperbólica para la migración en tiempo. En el caso de la migración en profundidad, la difracción toma una forma más compleja y queda definida por el trazado de rayos (*ray-tracing*) o bien mediante el cómputo de tiempos de viaje en una grilla ([Bancroft, 2001](#)).

En la implementación para este trabajo, se utilizó el módulo de migración Kirchhoff del software comercial EPOS™. Para la migración en profundidad el software utiliza trazado de rayos. El método de reconstrucción de frentes de onda utiliza el trazado dinámico de rayos en una cuadrícula, enviando rayos desde la fuente hacia puntos del subsuelo dentro de una apertura definida. En cada paso temporal, los extremos de los rayos definen los frentes de onda y se interpolan los tiempos de viaje entre los puntos de la cuadrícula. Este método también calcula el factor de dispersión geométrica y la longitud de la trayectoria, guardando el evento con la mayor amplitud para su migración en profundidad ([Paradigm™ 19, 2019](#)). Para poder realizar una comparación fidedigna entre ambos métodos analizados, en cada ejemplo se usó el mismo modelo de velocidad que el utilizado para el método de *split-step*.

DATOS

Los métodos de migración elegidos para este trabajo fueron aplicados sobre tres conjuntos de datos: un set de datos sintéticos y dos sets de datos reales. Cada conjunto consiste en una sección *poststack* y su respectivo modelo de velocidades. Los datos utilizados fueron los siguientes: (1) Datos sintéticos utilizando el modelo de velocidad SAIG model ([Signal Analysis and Imaging Group, University of Alberta, 2015](#)); (2) Datos reales de Tierra del Fuego, Argentina, y (3) Datos reales de Faja Plegada, Cuenca de Huallaga, Perú. Se describen en detalle a continuación.

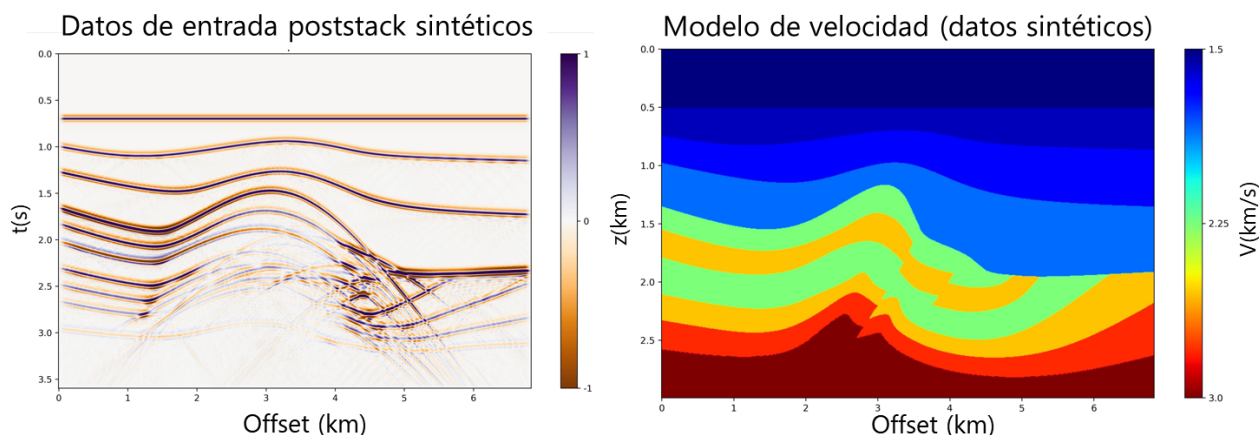


Figura 2. Sección poststack de entrada (panel izquierdo) y modelo de velocidades (panel derecho) correspondiente a los datos sintéticos generados con el SAIG model.

Datos sintéticos

Este conjunto de datos consiste en una sección sintética *poststack* y el correspondiente modelo de velocidad con el que fue generado. El modelo de velocidades está basado en el modelo Marmousi ideado por el *Institut français du pétrole (IFP)*. Los datos sintéticos fueron generados mediante un *script* de Seismic Unix (Cohen y otros, 2007) por el grupo SAIG (Signal Analysis and Imaging Group, University of Alberta, 2015). Se utilizó la rutina *sufdmod2.pml* (Hale y otros, 1998), que es un modelador de diferencias finitas de segundo orden para la ecuación de onda acústica e implementa las condiciones de borde PML (*perfectly matched layer*) (Hastings y otros, 1996).

El modelo de velocidad tiene una longitud cercana a los 7 km en el dominio del offset y 3 km de profundidad, dado por 684 trazas espaciadas $\Delta x = 10$ m en x y 600 muestras con una separación de $\Delta z = 5$ m en z . El dato *poststack* se modeló utilizando una ondícula de Ricker con frecuencia pico de 15 Hz con un intervalo de muestreo $\Delta t = 4$ ms. Cada traza tiene 900 muestras y la sección *stack* en tiempo llega hasta los 3.6 s. El contenido de frecuencias se extiende hasta los 45 Hz aproximadamente. El dato *poststack* de entrada en tiempo y el correspondiente modelo de velocidad utilizado se pueden observar en la Figura 2.

Datos sísmicos de Tierra del Fuego

Este conjunto de datos *poststack* y su respectivo modelo de velocidad corresponden a la Cuenca Austral, en Tierra del Fuego. El set de datos consta de 1401 trazas, espaciadas lateralmente en $\Delta x = 12.5$ m, por lo que la extensión lateral de la sección estudiada es cercana a los 17.5 km. El número de muestras en la sección *stack* es de 2001 con un intervalo de muestreo $\Delta t = 2$ ms y el modelo de velocidad tiene 1661 muestras en profundidad separadas $\Delta z = 5$ m. El espectro de amplitud muestra un contenido de frecuencias que alcanza los 60 Hz. El dato de entrada y el modelo de velocidad para Tierra del Fuego (TdF) se muestran en la Figura 3.

Datos sísmicos de Faja Plegada

El segundo conjunto de datos reales analizado corresponde a una faja plegada y corrida situada en la Cuenca de Huallaga en Perú. El número de trazas es de 3701. Los receptores están lateralmente espaciados en $\Delta x = 12.5$ m, por lo que la extensión en offset de la sección es de aproximadamente 46 km. El número de muestras en la sección *stack* es de 4001 con intervalo de muestreo $\Delta t = 2$ ms. El modelo de velocidad contiene 3901 muestras en profundidad con una separación de $\Delta z = 5$ m. En este caso, las frecuencias altas del espectro de amplitud fueron filtradas utilizando un filtro pasabanda

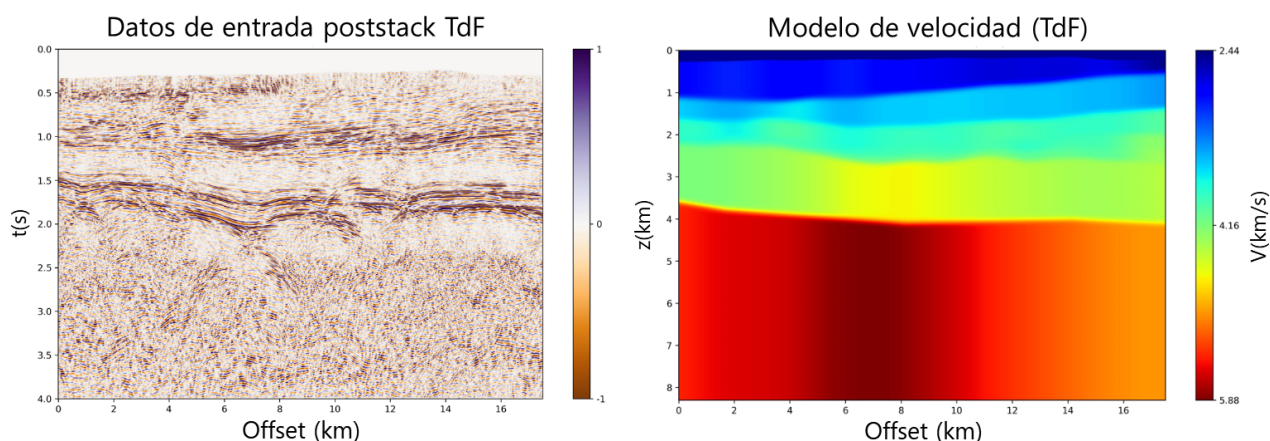


Figura 3. Sección poststack de entrada (panel izquierdo) y modelo de velocidades (panel derecho) correspondiente a los datos de Tierra del Fuego.

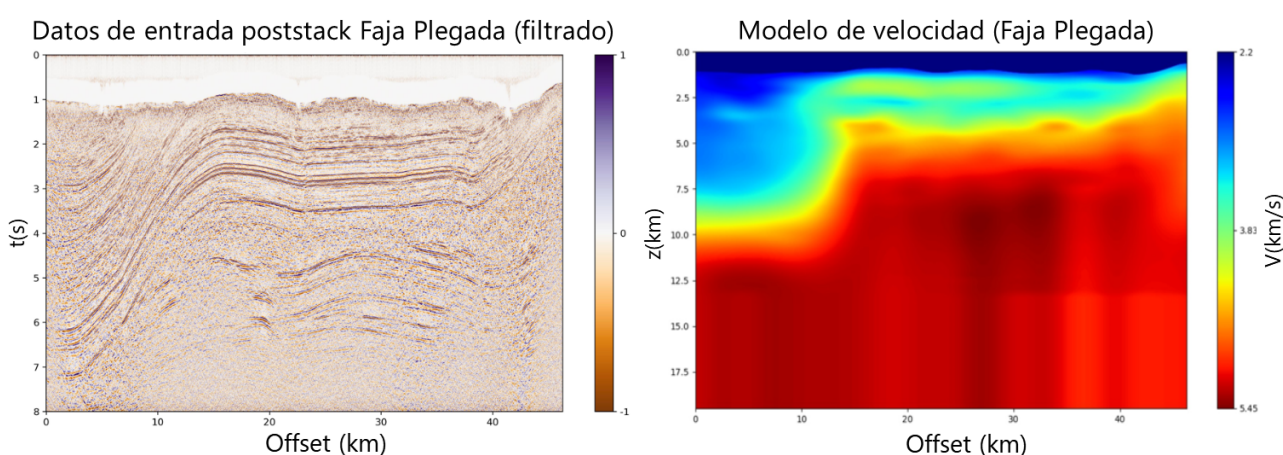


Figura 4. Sección poststack de entrada (panel izquierdo) y modelo de velocidades (panel derecho) correspondiente a los datos de Faja Plegada, Cuenca de Huallaga, Perú.

con frecuencias de corte bajas y altas para el filtrado de 0 – 0 Hz y 50 – 60 Hz, respectivamente. El dato de entrada y el modelo de velocidad para Faja Plegada se muestran en la [Figura 4](#).

PROCESAMIENTO

Parámetros de procesamiento

Los métodos de *split-step* y de migración en profundidad por Kirchhoff dependen de diferentes parámetros para su implementación. Algunos de ellos fueron descritos en las secciones previas en la que se detalló cada método. Se recuerda que el método de *split-step* se implementó mediante un código en lenguaje Julia. Por otro lado, para el método de Kirchhoff se utilizó el software comercial EPOS™. Los parámetros de adquisición de cada set de datos se resumen en la [Tabla 1](#). A su vez, los parámetros necesarios para correr los algoritmos de migración inherentes a cada método utilizado se muestran en la [Tabla 2](#) a continuación.

Algoritmo de migración de *split-step*

La aplicación de la migración *split-step* se realizó a través de un código en lenguaje Julia ([Sabbione y otros, 2016](#)) que fue adaptado para los objetivos del presente trabajo. Muchas de las funciones empleadas en el mismo son parte de los paquetes SeisMain, SeisPlot y SeisProcessing, del módulo

Parámetro	Datos sintéticos	Tierra del Fuego	Faja Plegada
Δt	4 ms	2 ms	2 ms
Δx	5 m	5 m	5 m
Δz	10 m	12.5 m	12.5 m
Frecuencia	40 Hz	60 Hz	60 Hz
SRD (Datum)	0 m s.n.m.	300 m s.n.m.	1500 m s.n.m.

Tabla 1. Parámetros de adquisición para cada set de datos.

Parámetro	Datos sintéticos	Tierra del Fuego	Faja Plegada
Floating Datum	–	Versión suavizada de la topografía	
Vel. de reemplazo	–	2500 m/s	2200 m/s
Apertura	Máx. según N° de CDPs	8000 m	12500 m
Filtro antialias	Mínimo	Mínimo	Mínimo

Tabla 2. Parámetros de procesamiento para cada set de datos.

de procesamiento Seismic Julia ([Stanton, 2015](#)).

El programa principal `poststack.jl` realiza la carga de datos de entrada y modelo de velocidad, declaración de variables, parámetros a utilizar y visualización de resultados. El código utiliza las siguientes funciones o subrutinas para implementar las distintas etapas del algoritmo:

- `zossm.jl`: Implementa la migración a offset cero. Para ello previamente se calculan las lentitudes, se aplica la condición de borde NBC y una ventana de Hamming (mediante funciones *ad-hoc* descritas a continuación). La sección migrada se obtiene de sumar para todas las frecuencias ω los campos previamente propagados.
- `slowness.jl`: Calcula los valores de lentitud y de lentitud de referencia que utiliza el algoritmo, tal cual se explica en el artículo de [Stoffa y otros \(1990\)](#).
- `nbc.jl`: Implementa las *Non-reflecting Boundary Conditions* (condiciones de borde no reflectivas) a partir de un vector en x que decae exponencialmente hacia los límites espaciales del dominio según la función que implementa la correspondiente condición de borde. Esta función se describe con profundidad más adelante y está basada en el artículo de [Cerjan y otros \(1985\)](#).
- `ssprop.jl`: Computa los operadores de propagación de ida y vuelta basados en el método de *split-step*. Esta función es el núcleo del algoritmo de migración de *split-step*. Se calcula la propagación del campo de ondas según el método propuesto por [Stoffa y otros \(1990\)](#). La salida arroja un campo propagado para todas las profundidades para la frecuencia ω . Luego se suma sobre todas las frecuencias digitales.

Condición de borde

Cuando se buscan las soluciones discretas de la ecuación de onda se generan reflexiones espurias en los límites de la grilla utilizada producto de la conservación de energía en los límites del dominio utilizado. Este efecto se debe mitigar para evitar que las señales sísmicas de interés queden enmascaradas por dichas reflexiones. Una posible solución es aumentar el tamaño de la grilla de forma tal que la zona de interés no se vea afectada por el efecto de borde, pero esto conlleva un mayor costo computacional. Otro posible enfoque tiene como objetivo proponer condiciones de borde que eviten las reflexiones (*non-reflecting boundary conditions*, NBC). Esta estrategia se basa en reemplazar la ecuación de onda en el borde por ecuaciones de onda hacia un solo sentido y que así la energía no pueda propagarse desde los bordes hacia el centro del dominio de la grilla. Sin embargo, la efectividad de este método disminuye para ángulos de incidencia bajos. Alternativamente, se puede

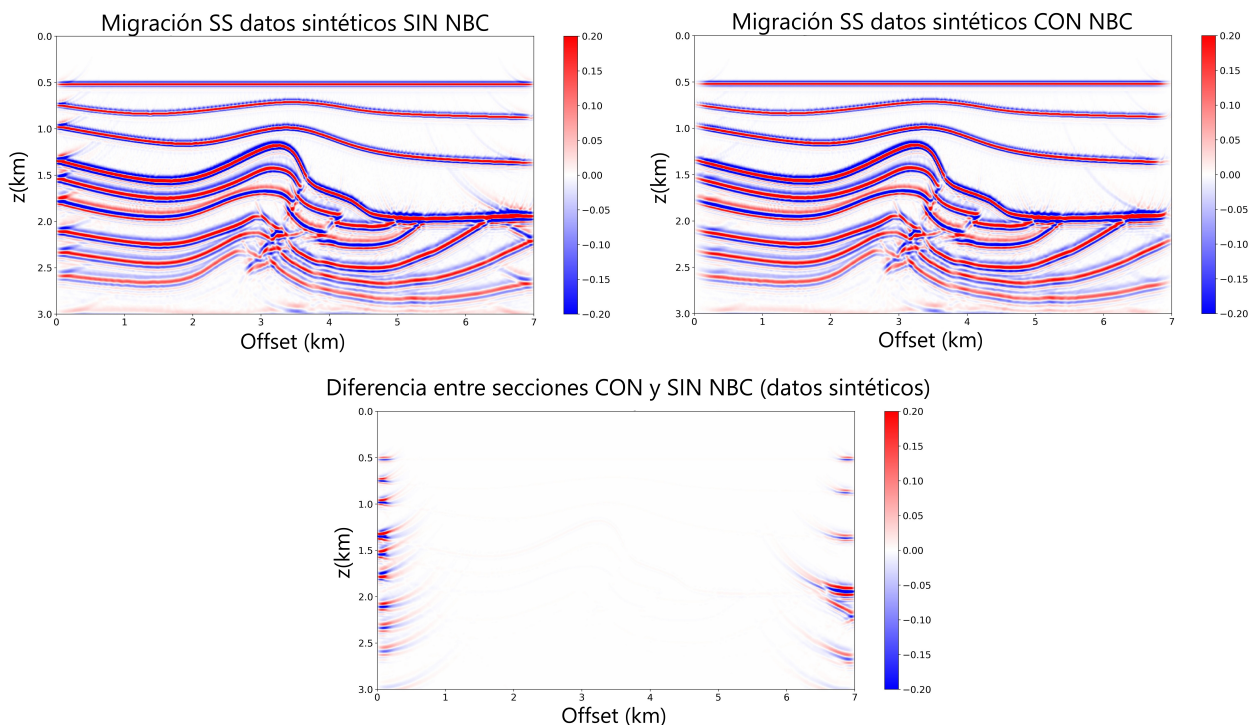


Figura 5. Condición de borde. Panel superior izquierdo: sección migrada mediante *split-step* de los datos sintéticos sin aplicar la condición de borde. Panel superior derecho: ídem anterior con la condición de borde aplicada. Panel inferior: diferencia entre los dos paneles superiores.

aplicar una reducción gradual de las amplitudes en una franja de nodos cercanos al contorno de la grilla. La efectividad en este caso no disminuye según los ángulos de incidencia. Este es el método implementado para este trabajo, que está basado en el artículo de [Cerjan y otros \(1985\)](#).

Para ejemplificar la aplicación de este método se realiza el siguiente proceso. En cada iteración de tiempo, se disminuyen los valores del campo dentro de una determinada franja en los límites del dominio espacial. La reducción de la amplitud en cada franja es aplicada gradualmente desde el borde interior. Para ello se calcula el siguiente factor de atenuación:

$$G = e^{-\alpha(N-i)^2}, \quad i = 1, \dots, N. \quad (10)$$

En el algoritmo de *split-step* se utilizó una franja de $N = 30$ con un valor de $\alpha = 0.005$. Así, G toma un valor igual a 1 para $i = 30$, y es igual a 0.97919 para $i = 1$. Para ilustrar el impacto e importancia de la inclusión de las condiciones de borde, en la [Figura 5](#) se muestran los resultados de la migración del ejemplo sintético mediante el método de *split-step* (SS) sin aplicar la condición de borde NBC y habiendo aplicado la condición de borde. Adicionalmente, se presenta la diferencia entre ambas secciones, donde se evidencia claramente el efecto de la reflexión en el contorno de la grilla que la condición de borde logra mitigar.

Migración Kirchhoff

El método de migración de Kirchhoff se aplica sumando la energía distribuida a lo largo de cada curva de difracción, que es así colapsada hasta enfocarse sobre la ubicación correspondiente a cada punto difractor. Este procedimiento se realiza para todos los puntos de la sección sísmica de entrada. La amplitud y el coeficiente de reflexión relativo se obtienen calculando una suma pesada de las muestras ([Veeken y otros, 2013](#)). Como se mencionó previamente, la migración Kirchhoff tiene en cuenta la oblicuidad, la divergencia esférica y las características de la ondícula. La implementación del algoritmo Kirchhoff de migración en profundidad se realizó mediante el software EPOS™ (**epos**).

Para el trazado de rayos se utilizaron los modelos de velocidad intervállica descritos en el capítulo de descripción de los Datos.

Para ejecutar el método de Kirchhoff se deben definir ciertos parámetros. Entre ellos, se destacan la apertura de migración y el filtro antialias utilizado. Estos dos parámetros y su efecto sobre los resultados se describen brevemente a continuación.

Apertura

La apertura de migración es la extensión lateral del operador de migración. En la práctica, este parámetro está determinado —y limitado— por la amplitud lateral de los datos disponibles. El tamaño de la apertura determina además el buzamiento que puede ser resuelto en el proceso de migración. Para seleccionar un valor óptimo se probaron distintos valores y se evaluó el desempeño del algoritmo por inspección visual. Se optó por el mayor valor de apertura a partir del cual ya no se percibía una mejora en la sección. En el caso particular de los datos sintéticos, se consideró la máxima apertura en función de la cantidad de CDPs. Los valores que se utilizaron para migrar los datos reales fueron de 8.0 Km para Tierra del Fuego y de 12.5 Km para Faja Plegada. Estos valores se indican en la [Tabla 2](#). En el caso de los datos reales estos valores se alcanzan a una profundidad de 1000 m.

Filtro antialias

El filtro antialias se utiliza para eliminar frecuencias indeseadas en el espectro de los datos sísmicos. Tiene como fin remover aquellas frecuencias que puedan haber causado *aliasing* en el proceso de muestreo de la señal analógica durante la adquisición o cuando el intervalo de muestreo de los datos digitales disminuye debido a alguna instancia del procesamiento sísmico ([SLB, 2023](#)). El *aliasing* en el operador de migración se presenta cuando la inclinación de los estratos es elevada y ocurre primero para las altas frecuencias. El *aliasing* espacial en datos sísmicos ocurre cuando el espaciado de los receptores es demasiado amplio, especialmente cuando las trayectorias de los rayos que arriban a los receptores presentan alta inclinación. A su vez, el operador de migración de Kirchhoff también sufre de *aliasing*, lo que afecta la imagen obtenida especialmente en áreas someras. Los filtros antialias ayudan a reducir estos efectos aplicando un filtrado de frecuencias altas, pero a velocidades bajas pueden reducir la resolución en las zonas someras ([Paradigm™ 19, 2019](#)).

En el software EPOS™, el filtro se define mediante un parámetro que permite ajustar el nivel de filtrado, donde el mínimo actúa de manera más conservadora, mientras que el máximo elimina más frecuencias del espectro. En la implementación del algoritmo de migración Kirchhoff, se eligió el nivel mínimo de filtrado para los tres ejemplos analizados. De esta manera, si bien se produce una imagen algo más ruidosa, se reduce el riesgo de sobre-filtrar los datos (ver [Tabla 2](#)).

RESULTADOS

Tanto en la implementación del algoritmo de migración Kirchhoff como en el método *split-step* se utilizaron los mismos modelos de velocidad y datos de entrada. De esta manera, los resultados finales reflejan exclusivamente las propiedades y diferencias de cada método. Luego de haber aplicado ambos algoritmos de migración en profundidad sobre los tres conjuntos de datos *poststack* se obtuvieron las secciones migradas que se describen a continuación.

Sintéticos

En el caso de los datos sintéticos se puede observar que ambos algoritmos de migración en profundidad resuelven adecuadamente la reflectividad del modelo. La migración colapsa la energía de las hipérbolas de difracción que aparecen en la zona del fallamiento inverso permitiendo distinguir de forma nítida la estructura en el centro de la sección ([Figura 6](#)). A su vez, se puede observar que

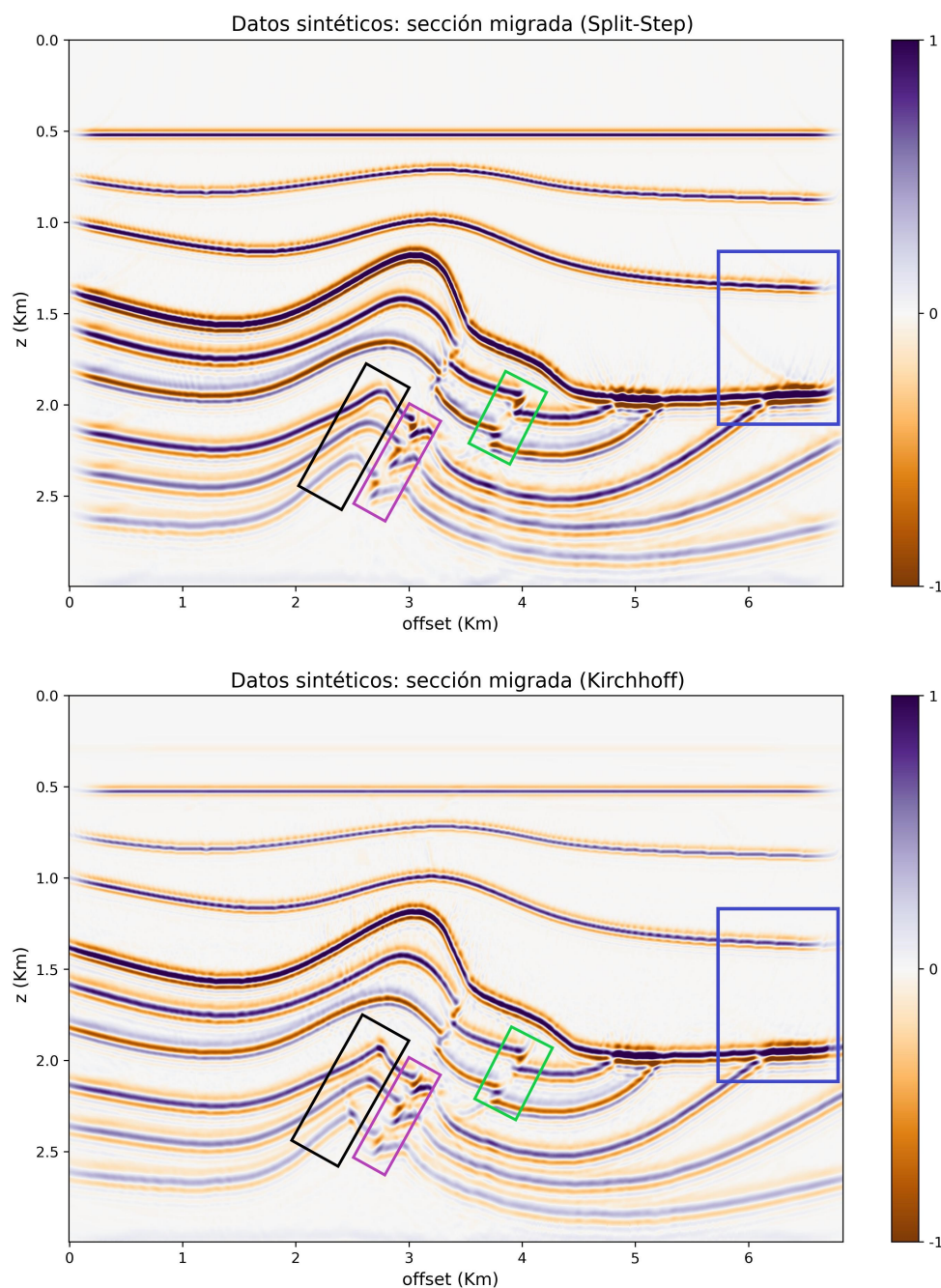


Figura 6. Datos sintéticos migrados con el algoritmo *split-step* (panel superior) y con Kirchhoff (panel inferior). Las zonas donde se observan diferencias se encuentran remarcadas con rectángulos de diferentes colores.

el anticlinal en el centro de la sección es resuelto como una estructura más angosta en la sección migrada.

En la [Figura 6](#) se destacan con rectángulos de diferentes colores las zonas de las secciones en las que se perciben diferencias entre los resultados de los algoritmos de migración. Particularmente, en los estratos plegados en el centro del anticlinal, como se muestra en el cuadro a) de la [Figura 7](#), se observa una suave diferencia en la curvatura de los pliegues. Además, se puede notar una leve diferencia en la delimitación de las fallas inversas. Esto se observa con más detalle en la [Figura 7](#), cuadros b) y c). En el caso de la migración Kirchhoff la curvatura es más angulosa. En el cuadro d) queda de manifiesto el hecho de que el problema de borde es mejor mitigado mediante el algoritmo Kirchhoff que mediante el del *split-step*.

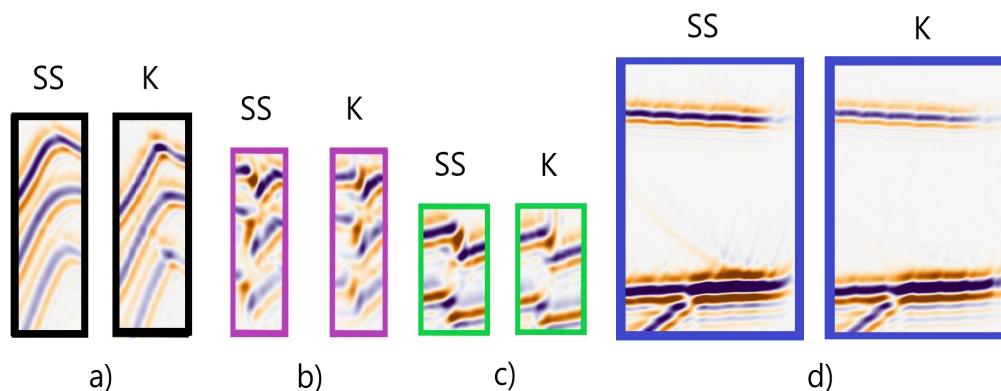


Figura 7. Recortes de las zonas delimitadas en la [Figura 6](#) con diferencias entre las secciones migradas en el caso de datos sintéticos. En cada cuadro el panel izquierdo corresponde al algoritmo split-step (SS) y el derecho a Kirchhoff (K).

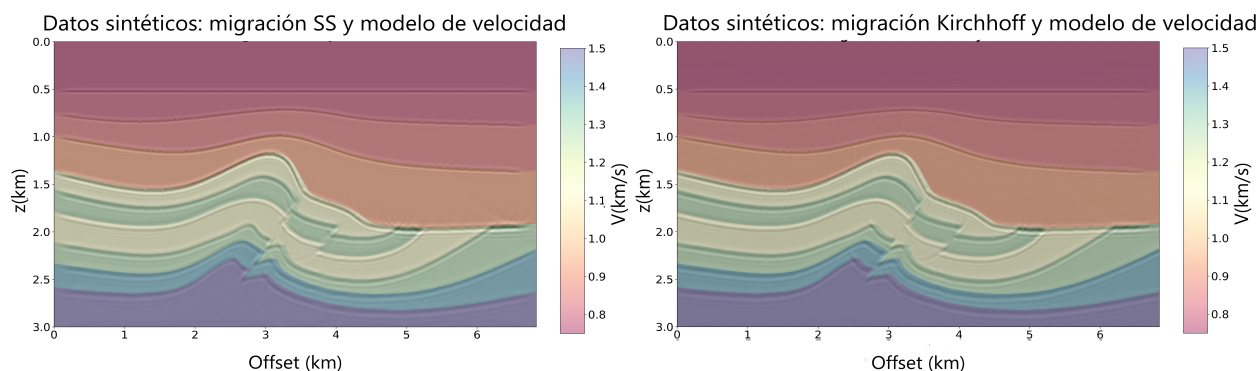


Figura 8. Secciones migradas mediante split-step (panel izquierdo) y Kirchhoff (panel derecho) superpuestas con el modelo de velocidad utilizado para generar los datos sintéticos.

Finalmente, en la [Figura 8](#) se superponen las secciones finales con los modelos de velocidad utilizados para generar los datos (y para correr las migraciones). Se aprecia claramente el acuerdo entre el modelo de velocidad de entrada y las secciones migradas mediante cada método.

Tierra del Fuego

Analizamos ahora los resultados de la sección que corresponde a una línea sísmica de la Cuenca Austral, en Tierra del Fuego. En los datos *poststack* sin migrar se observa una geometría en la que los estratos están mayoritariamente paralelos ([Figura 3](#)). No obstante, en los datos de entrada se distingue la presencia de hipérbolas de difracción especialmente en la parte central de la sección y cerca del segundo paquete de estratos. Los datos migrados mediante los dos métodos utilizados se muestran en la [Figura 9](#). En las secciones migradas por ambos métodos vemos que las hipérbolas se colapsan apropiadamente. Esto deja de manifiesto una mejor delimitación de los estratos y permite observar la presencia de algunos pliegues. En términos generales, ambas secciones migradas logran un resultado similar.

Las zonas que presentan diferencias en los resultados se delimitan con rectángulos en la [Figura 9](#). Como se remarca allí, en la zona central de la sección (a los 4 km de profundidad) la estructura de estratos plegados se ve más amplia en el caso de la implementación del algoritmo Kirchhoff. Esto se destaca en la [Figura 10](#). También se observa que los estratos quedan ubicados a mayor profundidad en el caso de la migración *split-step*.

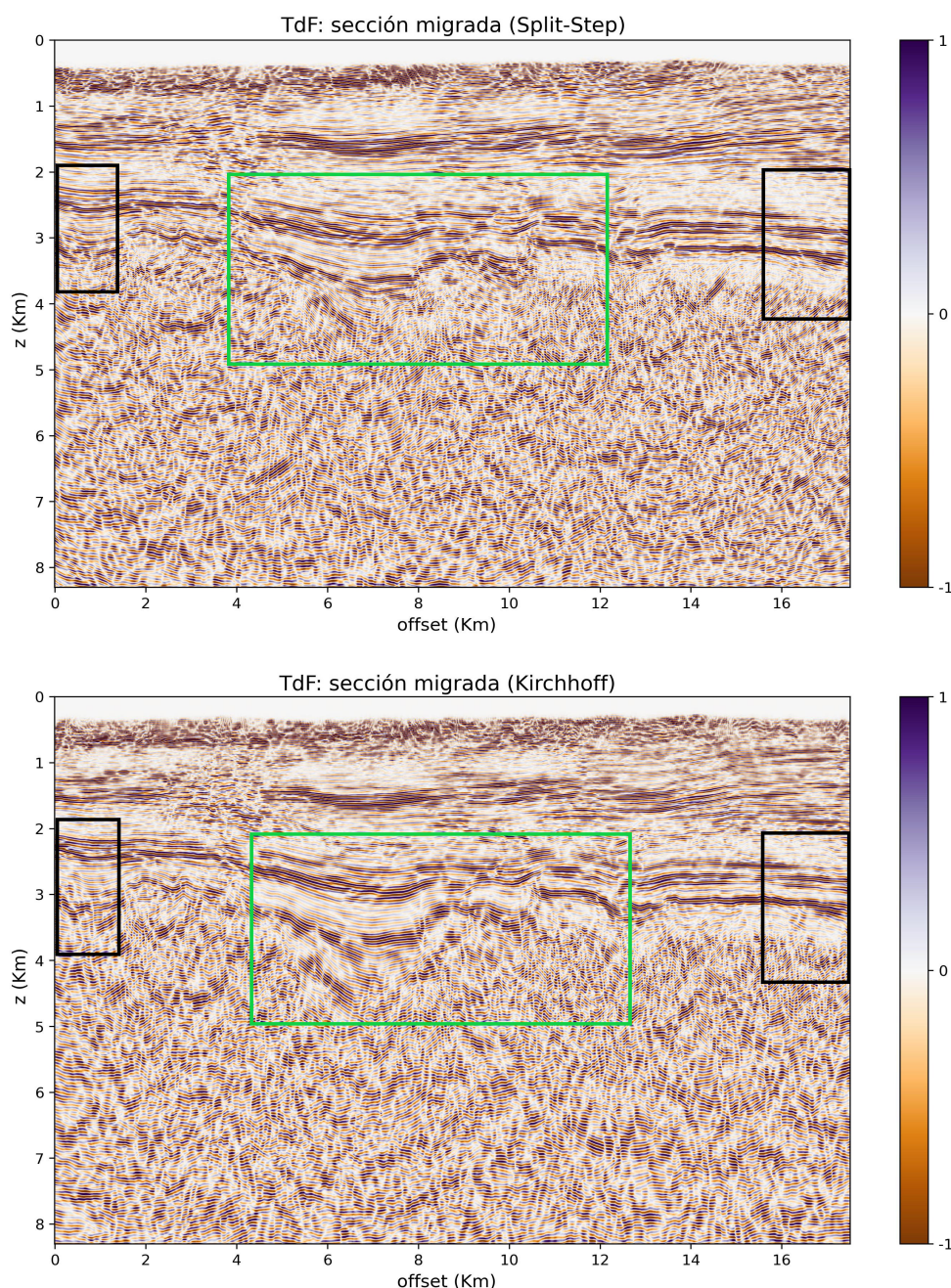


Figura 9. Datos de Tierra del Fuego migrados con el algoritmo *split-step* (panel superior) y con Kirchhoff (panel inferior). Las zonas donde se observan diferencias se encuentran remarcadas con rectángulos de diferentes colores.

Por último, en la [Figura 11](#) se muestra la superposición de los resultados de los algoritmos con el modelo de velocidad utilizado. La figura demuestra que las profundidades y estructuras delimitadas luego de migrar por ambos métodos concuerdan muy bien con el modelo de velocidad utilizado.

Faja Plegada

La sección estudiada de la Cuenca Huallaga, Perú, corresponde a un área de faja plegada y corrida que presenta una serie de estructuras con importante buzamiento ([Figura 4](#)). En la [Figura 12](#) se presentan los resultados de la migración por el método de *split-step* y por el método de Kirchhoff. El flanco presente en la sección suroeste (correspondiente al sinclinal Biabo) es acortado en las secciones resultantes luego de la migración. En los estratos plegados profundos al migrar los datos

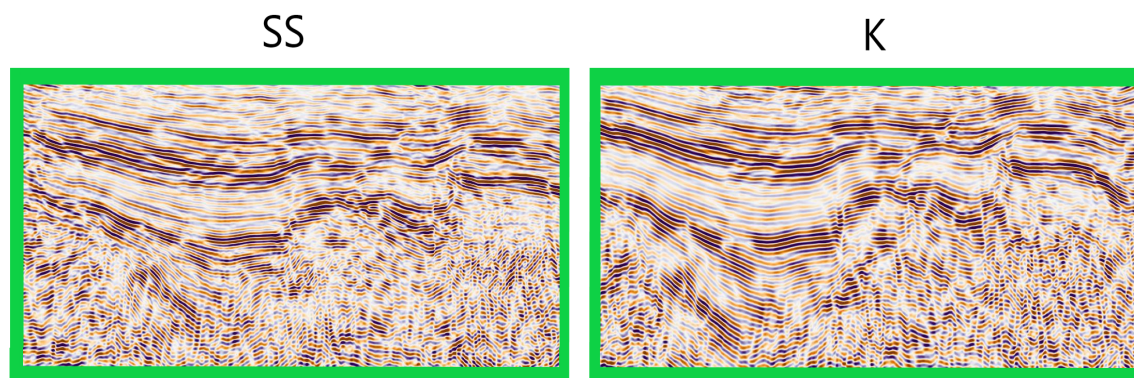


Figura 10. Recortes de las diferencias entre las secciones migradas en el caso de datos de Tierra del Fuego. El panel izquierdo corresponde al algoritmo *split-step* (SS) y el derecho a Kirchhoff (K).

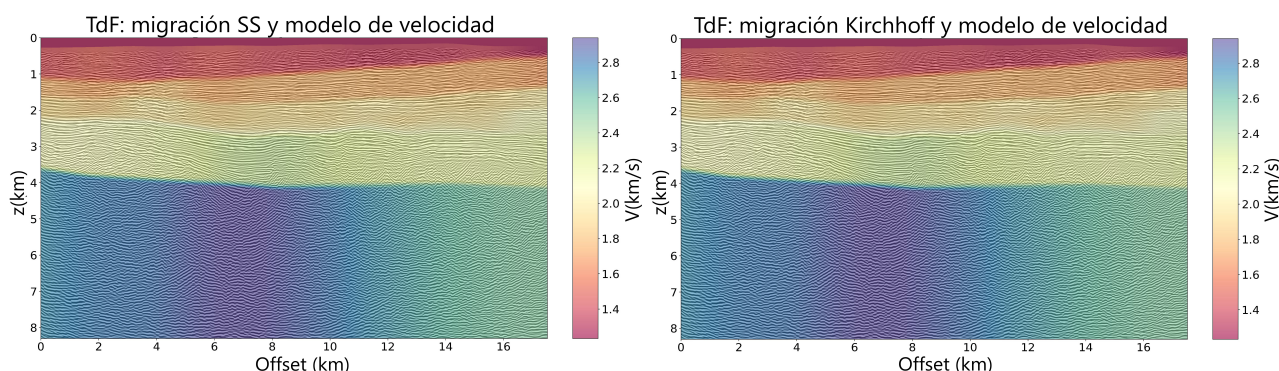


Figura 11. Secciones migradas mediante *split-step* (panel izquierdo) y Kirchhoff (panel derecho) superpuestas con el respectivo modelo de velocidad para el caso de los datos de Tierra del Fuego.

se puede apreciar que estos corresponden a estructuras más cerradas. También se puede observar que el conjunto de pliegues cercano al flanco suroeste del sinclinal Biabo (15 km de offset) tiene una curvatura cóncava previo a la migración, y que luego de aplicar los algoritmos de migración este paquete de pliegues pasa a ser convexo. Las zonas recién descritas se destacan en la [Figura 12](#) mediante dos áreas rectangulares.

Para destacar las principales diferencias en los resultados obtenidos por cada método, en la [Figura 13](#) se muestra un zoom de las zonas delineadas en la [Figura 12](#). En el caso del algoritmo *split-step* los estratos buzantes correspondientes al flanco suroeste se observan con mayor continuidad y nitidez que en el caso de la migración con el algoritmo Kirchhoff. Esto se advierte también en los estratos buzantes del lado noreste de la sección. Estas diferencias se pueden apreciar claramente en la [Figura 13](#) en los cuadros a) y b), respectivamente.

Por último, en la [Figura 14](#) se presentan las secciones migradas mediante ambos algoritmos de migración superpuestas con el modelo de velocidad correspondiente. Si bien el modelo de velocidad utilizado es suave y aproximado, se puede apreciar especialmente en las zonas menos profundas el acuerdo de las estructuras buzantes con el modelo de velocidad utilizado.

DISCUSIÓN

El método de *split-step* fue programado de forma modular, donde cada módulo ejecuta cada uno de los procesos del algoritmo. Como parte del procesamiento se aplicó una ventana de *Hamming* y una

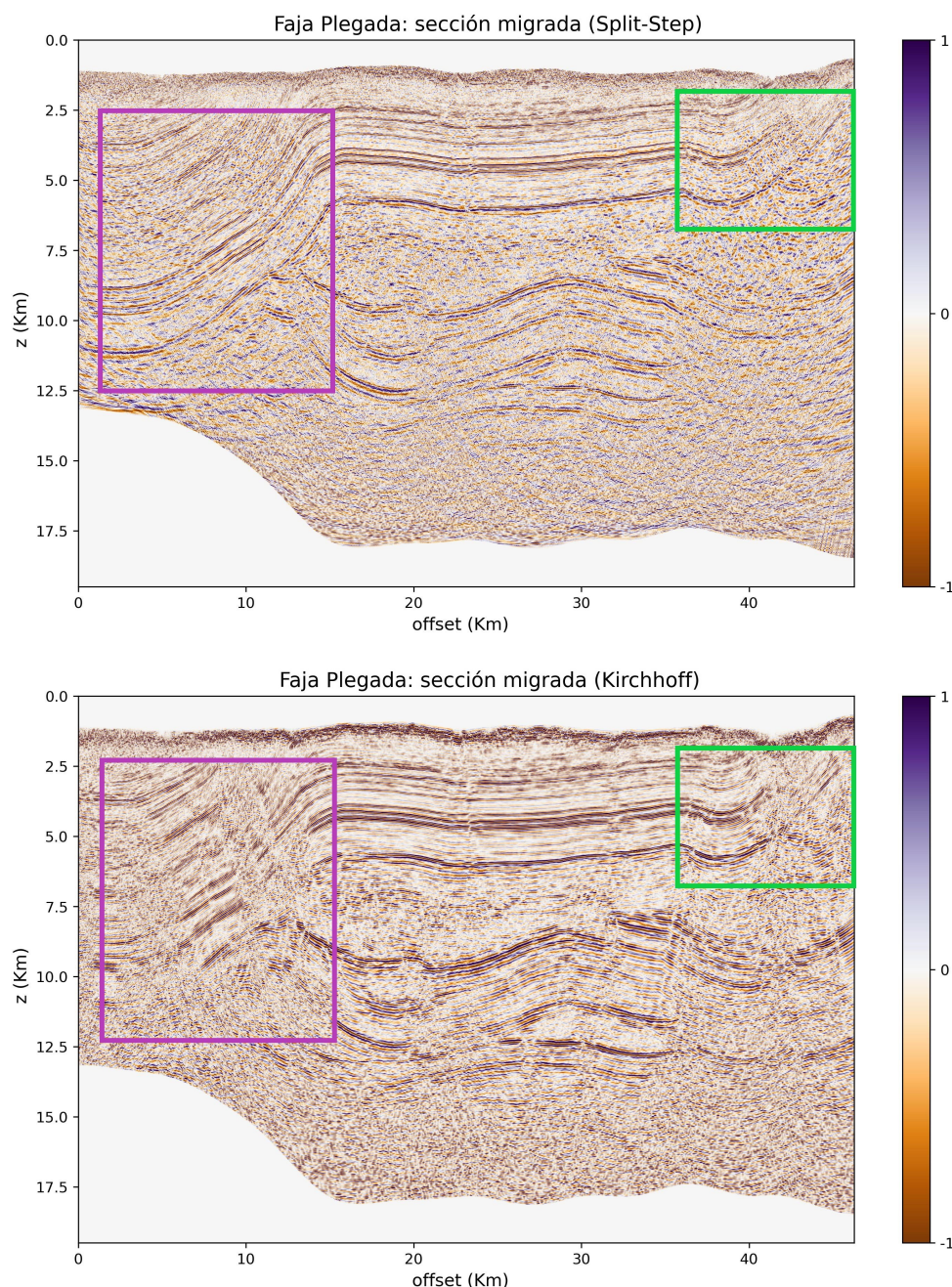


Figura 12. Datos de Faja Plegada migrados con el algoritmo split-step (panel superior) y con Kirchhoff (panel inferior). Las zonas donde se observan diferencias se encuentran remarcadas con rectángulos de diferentes colores.

condición de borde de tipo NBC (*non-reflecting boundary condition*). Por otro lado, la implementación del algoritmo de migración Kirchhoff se realizó a través del software EPOS™. Entre los parámetros de procesamiento requeridos por este software se deben definir el valor del filtro antialias a aplicar y la apertura de migración. Se optó por el filtro con menor intensidad disponible y con un valor de apertura asignado según las características de cada set de datos. A su vez, para la implementación de ambos métodos, se utilizaron parámetros específicos de los datos de entrada, tales como el rango de frecuencias, la altura del Datum de referencia (SRD) y los intervalos de muestreo temporales y espaciales. Los experimentos mediante datos sintéticos proporcionaron un ejemplo controlado en el que se conoce la salida deseada y se puede evaluar la efectividad de los métodos. Una vez realizada la validación de los algoritmos a partir de los experimentos con el set de datos sintéticos, se procesaron los datos reales. El dato de Tierra del Fuego contiene horizontes de bajo buzamiento,

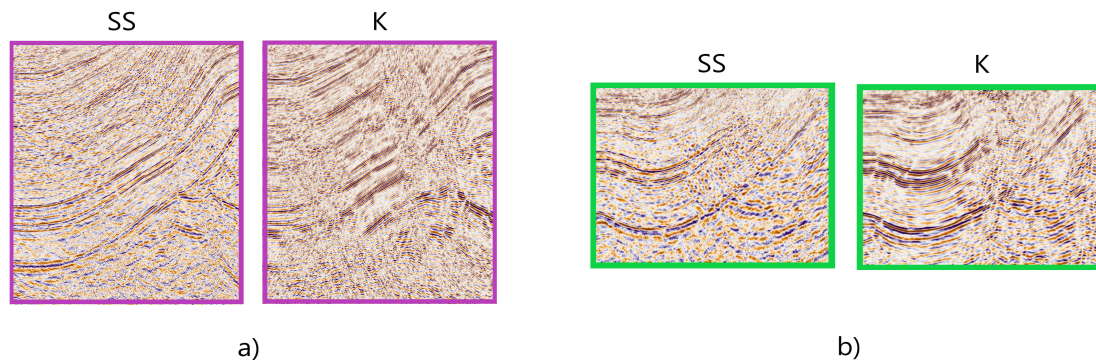


Figura 13. Recortes de las diferencias entre las secciones migradas en el caso de datos de Faja Plegada. En cada cuadro el panel izquierdo corresponde al algoritmo *split-step* (SS) y el derecho a Kirchhoff (K).

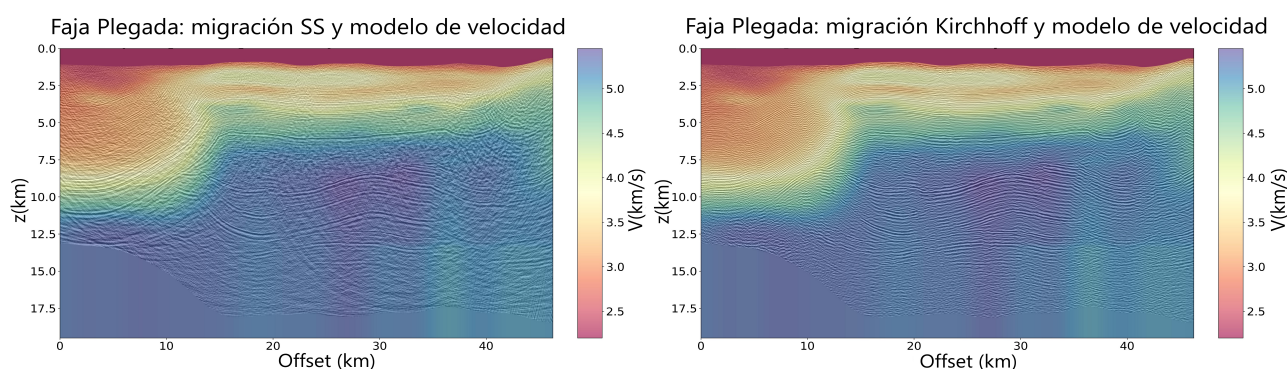


Figura 14. Secciones migradas mediante *split-step* (panel izquierdo) y Kirchhoff (panel derecho) superpuestas con el respectivo modelo de velocidad para el caso de datos de Faja Plegada.

mientras que el dato correspondiente a Faja Plegada representa el ejemplo de mayor complejidad estructural. En términos generales, ambos algoritmos produjeron secciones correctamente migradas y mayormente similares entre sí.

Tanto el ejemplo con los datos sintéticos como los ejemplos con datos reales contienen características en sus secciones de entrada que permiten evaluar el desempeño de un proceso de migración, tales como hipérbolas de difracción, estratos buzantes mal posicionados y anticlinales de apertura aumentada. Luego de haber implementado los algoritmos de migración estas distorsiones en la imagen sísmica fueron en su mayoría correctamente resueltas y corregidas por ambos métodos. Es decir, en términos generales la aplicación de los métodos de migración estudiados logró los resultados deseados tanto sobre datos sintéticos como en los ejemplos con datos reales, que presentan distintos niveles de complejidad geológica. Si bien se trata de dos implementaciones de distinta naturaleza, siendo uno de ellos un código de desarrollo académico y el otro parte de un software comercial, los resultados en general fueron coherentes entre sí. Esta situación podría ser aprovechada con fines de control, cotejando los resultados entre ambos métodos.

El algoritmo de *split-step* resolvió en forma más adecuada que el método de Kirchhoff la migración de estratos buzantes, generando una imagen sísmica con mayor claridad y continuidad. Esto se puede explicar por la naturaleza de los métodos. Mientras que el algoritmo de migración Kirchhoff colapsa la energía de las curvas según la trayectoria de los rayos, *split-step* utiliza la ecuación de onda para reconstruir la imagen del subsuelo. En escenarios donde el buzamiento es prominente, la energía difractada registrada no contiene la suficiente cobertura que permita colapsar adecuadamente las hipérbolas de difracción, lo que dificulta el desempeño del método de Kirchhoff. Por ello, en estos casos el algoritmo de Kirchhoff no brinda el mejor desempeño. Por el contrario, el método *split-step*

tiene en consideración las variaciones laterales de velocidad en su corrección por inhomogeneidad, por lo que resuelve mejor los marcos geológicos con altos buzamientos. Respecto de la presencia de fallamientos estructurales, los resultados fueron muy similares, si bien se observó en las secciones migradas por el método de Kirchhoff que las fallas resultan ligeramente mejor delineadas. Por último, se observó que el algoritmo de migración Kirchhoff resuelve mejor los efectos de borde ya que logra mitigarlos de manera más adecuada. Esto podría deberse a una parametrización no óptima en el caso del método *split-step*.

En el caso de los datos sintéticos puede observarse que en ambos métodos las secciones obtenidas no coinciden exactamente con el modelo de velocidad en presencia de estratos con buzamiento elevado. Esto puede deberse a que la aproximación de sección *stack* como sección de offset cero no es adecuada en este escenario y que la migración *poststack* resulta insuficiente. En el caso de Tierra del Fuego existe una leve diferencia entre ambos métodos en las profundidades obtenidas. Dado que los tiempos y velocidades utilizados en ambos casos son los mismos, no debería existir dicha discrepancia. Por lo tanto, podemos concluir que este comportamiento no se debe en sí al desempeño de los métodos sino a alguna cuestión externa que podría estar relacionada al Datum utilizado.

Las diferencias de desempeño entre los métodos son más notorias en los datos reales, lo que podría estar relacionado con la presencia de ruido inherente a la medición. En el caso de los datos sintéticos, al tratarse de datos generados de forma controlada, no contienen ruido de medición ni incertidumbre en los parámetros. Esto determina que las diferencias observadas en el ejemplo sintético se deban únicamente a las características de cada algoritmo, y genera resultados más consistentes entre ambos métodos. En contraste, los datos reales contienen distintos tipos de ruido, tanto aleatorio como coherente, que afectan los resultados. Estas condiciones influyen en la respuesta de los algoritmos y pueden explicar por qué las diferencias entre los métodos resultan más notorias para los ejemplos con datos reales. Asimismo, la implementación del filtro antialias también podría haber influido en las diferencias observadas, ya que en el método de Kirchhoff se optó por la opción de menor intensidad entre las disponibles en el software, lo que pudo haber aumentado el nivel de ruido en la imagen sísmica.

CONCLUSIONES

Se compararon dos algoritmos de migración *poststack*: el método de Kirchhoff, implementado a partir de un software comercial, y el método de *split-step*, implementado a partir de un código propio desarrollado en lenguaje Julia. Para ello, se utilizaron los mismos conjuntos de datos de entrada y modelos de velocidad. Los datos utilizados para la comparación están dados por un conjunto de datos sintéticos y dos conjuntos de datos reales. Estos últimos corresponden a una sección de la Cuenca Austral, en Tierra del Fuego, y otra de una zona de faja plegada y corrida, en la Cuenca del Huallaga, Perú.

En términos generales, ambos métodos generan una imagen precisa de las secciones sísmicas para los conjuntos de datos analizados. Sin embargo, debido a la naturaleza de los métodos, *split-step* proporciona una reconstrucción más definida de los estratos con mayores buzamientos, mientras que Kirchhoff maneja mejor los problemas de borde y ofrece un desempeño ligeramente superior en presencia de fallas.

Los resultados sugieren que resultaría de interés como trabajo a futuro adaptar el código en Julia para migraciones de datos pre-apilados (*prestack*). De esta manera, sería posible resolver escenarios geológicamente más complejos en los que la migración *poststack* es insuficiente.

AGRADECIMIENTOS

A YPF S.A. por facilitar el uso de los dos sets de datos reales estudiados y por permitir el uso académico del software EPOS™. Este trabajo fue parcialmente subsidiado por la Universidad Nacional de La Plata (Proyecto de Incentivos Docentes de UNLP código G190) y por la Agencia Nacional de Promoción de la Investigación, el Desarrollo Tecnológico y la Innovación (Proyecto PICT código 2019-1342).

REFERENCIAS

- Bancroft, J. C. (2001). Seismic imaging: Post-stack. *CSEG Recorder*, 26(7).
- Bjørlykke, K. (2015). *Petroleum geoscience: From sedimentary environments to rock physics* (2ª ed.). Springer-Verlag Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-34132-8>
- Cerjan, C., Kosloff, D., Kosloff, R. y Reshef, M. (1985). A nonreflecting boundary condition for discrete acoustic and elastic wave equations. *Geophysics*, 50(4), 705-708. <http://dx.doi.org/10.1190/1.1441945>
- Cohen, J. K. y Stockwell, J. M. (2007). *CWP/SU: Seismic Unix Release No. 4.0: An open source software package for seismic research and processing*. Center for Wave Phenomena, Colorado School of Mines.
- Dobrin, M. B. y Savit, C. H. (1988). *Introduction to Geophysical Prospecting* (4ª ed.). McGraw-Hill International Editions.
- Gazdag, J. (1978). Wave equation migration with the phase-shift method. *Geophysics*, 43(7), 1342-1351. <https://doi.org/10.1190/1.1440899>
- Gazdag, J. y Sguazzero, P. (1984). Migration of seismic data by phase shift plus interpolation. *Geophysics*, 49(2), 124-131. <https://doi.org/10.1190/1.1441643>
- Gray, S. H., Etgen, J., Dellinger, J. y Whitmore, D. (2001). Seismic migration problems and solutions. *Geophysics*, 66(5), 1622-1640. <https://doi.org/10.1190/1.1487107>
- Hale, D., Stockwell, J., Artley, C. y Holzrichter, M. (1998). *Finite-Difference MODELing (2nd order) for acoustic wave equation with PML absorbing boundary conditions*. Recuperado el 26 de noviembre de 2023 de https://sepwww.stanford.edu/oldsep/cliner/files/suhelp/sufdmod2_pml.txt.
- Hastings, F. D., Schneider, J. B. y Broschat, S. L. (1996). Application of the perfectly matched layer (PML) absorbing boundary condition to elastic wave propagation. *The Journal of the Acoustical Society of America*, 100(5), 3061-3069. <https://doi.org/10.1121/1.417118>
- Onajite, E. (2014). *Seismic Data Analysis Techniques in Hydrocarbon Exploration*. Elsevier.
- Paradigm™ 19. (2019). *User Guide, Overview of migrations and additional information on Kirchhoff Migrations*.
- Sabbione, J. I., Stanton, A. y Sacchi, M. D. (2016). SeismicJulia: moving forward. *Annual Report SAIG 17*.
- Schneider, W. A. (1978). Integral formulation for migration in two and three dimensions. *Geophysics*, 43(1), 49-76. <https://doi.org/10.1190/1.1440828>
- Signal Analysis and Imaging Group, University of Alberta. (2015). *SAIG velocity model*. Recuperado el 9 de octubre de 2023 de https://saigfileservr.physics.ualberta.ca/datasets/testing_datasets/
- SLB. (2023). *Antialias filter*. Recuperado el 9 de octubre de 2023 de <https://glossary.slb.com/Terms/a/antialias.filter.aspx>
- Stanton, A. (2015). *SeisMain.jl [Software]* Disponible en SeismicJulia. Recuperado el 7 de septiembre de 2023 de <https://seismicjulia.github.io/SeisMain.jl/dev/lib/public/>
- Stoffa, P. L., Fokkema, J. T., de Luna Freire, R. M. y Kessinger, W. P. (1990). Split-step Fourier migration. *Geophysics*, 55(4), 410-421. <https://doi.org/10.1190/1.1442850>
- Veeken, P. C. y van Moerkerken, B. (2013). *Seismic stratigraphy and depositional facies models*. EAGE. <https://doi.org/10.3997/9789073834675>
- Yilmaz, Ö. (2001). *Seismic data analysis*. Society of Exploration Geophysicists. <https://doi.org/10.1190/1.9781560801580>