

# DETECCIÓN Y CONVERGENCIA DE APROXIMANTES PARA FALLAS VERTICALES LATENTES EN SUPERFICIES DISCONTINUAS

## DETECTION AND CONVERGENCE OF APPROXIMANTS FOR LATENT VERTICAL FAILURES ON DISCONTINUOUS SURFACES

Jessica Esthefania Paredes Morales<sup>1</sup>, Ramón Antonio Abancín Ospina<sup>2</sup>

{jessica.paredes@unach.edu.ec<sup>1</sup>, ramon.abancin@epoch.edu.ec<sup>2</sup>}

Fecha de recepción: 13/06/2025 / Fecha de aceptación: 01/07/2025 / Fecha de publicación: 01/07/2025

**RESUMEN:** La detección y/o aproximación de discontinuidades en funciones no regulares, definidas explícitamente por  $z = f(x, y)$  a partir de un conjunto de datos dispersos e irregulares de tipo Lagrange, representa un reto teórico con múltiples aplicaciones prácticas. Entre estas aplicaciones destaca la modelización geológica, que requiere la construcción de aproximaciones precisas para la localización de fallas (ya sean verticales, oblicuas, directas o inversas). En este contexto, desde una perspectiva teórica orientada a la aplicación práctica, el objetivo de la investigación fue construir aproximaciones para localizar fallas verticales en superficies discontinuas y estudiar su convergencia. Para ello, la metodología se basó en lo siguiente: partiendo de un conjunto de datos sintéticos de tipo Lagrange, dispersos e irregulares, el primer paso fue construir una secuencia de curvas poligonales que aproximan y localizan la falla vertical estudiada; y el segundo paso fue analizar la convergencia de estas aproximaciones. El principal resultado fue la implementación computacional en *Python* de la metodología para detectar fallas verticales latentes en superficies discontinuas, que utiliza datos dispersos de tipo Lagrange. Esta técnica genera familias de curvas poligonales aproximadas que convergen a la falla original. Se concluye que la metodología propuesta para la detección y convergencia de fallas verticales latentes en superficies discontinuas sienta las bases para construir aproximaciones óptimas. Donde, estas aproximaciones son lo suficientemente flexibles para adaptarse a los requisitos de entrada necesarios para su futura implementación numérica y computacional en algoritmos enfocados al proceso de aproximación de funciones no regulares.

**Palabras clave:** Superficies discontinuas, detección de discontinuidades, aproximación de discontinuidades, fallas verticales, curvas poligonales, convergencia de aproximantes

<sup>1</sup>Dirección de Posgrado, Programa de Maestría en Matemática aplicada con mención en Matemática Computacional, Universidad Nacional de Chimborazo (UNACH), Ecuador, <https://orcid.org/0000-0002-9163-3561>.

<sup>2</sup>Facultad de Ciencias, Carrera de Matemática, Grupo de Investigación CIED, Escuela Superior Politécnica de Chimborazo (ESPOCH). Doctorado en Matemáticas, Universidad Simón Bolívar (USB), Venezuela, <https://orcid.org/0000-0002-2417-6671>.

**ABSTRACT:** The detection and/or approximation of discontinuities in non-regular functions, explicitly defined by  $z = f(x, y)$  by from a set of dispersed and irregular Lagrangian-type data, represents a theoretical challenge with multiple practical applications. Among these applications, geological modeling stands out, which requires the construction of precise approximations for the location of faults (whether vertical, oblique, direct, or inverse). In this context, from a theoretical perspective oriented towards practical application, the objective of the research was to construct approximations to locate vertical faults on discontinuous surfaces and study their convergence. For this purpose, the methodology was based on the following: starting from a set of synthetic Lagrangian-type data, dispersed and irregular, the first step was to construct a sequence of polygonal curves that approximate and locate the vertical fault studied; and the second step was to analyze the convergence of these approximations. The main result was the computational implementation in Python of the methodology for detecting latent vertical faults on discontinuous surfaces, which uses sparse Lagrangian data. This technique generates families of approximate polygonal curves that converge to the original fault. It is concluded that the proposed methodology for the detection and convergence of latent vertical faults on discontinuous surfaces lays the groundwork for constructing optimal approximations. These approximations are flexible enough to adapt to the input requirements necessary for their future numerical and computational implementation in algorithms focused on the approximation process of non-regular functions.

**Keywords:** *Discontinuous surfaces, detection of discontinuities, approximation of discontinuities, vertical faults, polygonal curves, convergence of approximants*

## INTRODUCCIÓN

Los métodos para identificar y aproximar conjuntos de discontinuidades en funciones no regulares (o discontinuas) poseen aplicaciones diversas, cuya implementación depende críticamente del contexto y la distribución de los datos (1), (2). En el ámbito del procesamiento de imágenes, por ejemplo, los datos suelen estar regularmente distribuidos (*e.g.*, píxeles), lo que permite emplear técnicas de detección de bordes para extraer contornos y perfiles, con aplicaciones en infografía, reconocimiento de patrones, imágenes satelitales y diagnósticos médicos (3), (4). Por otro lado, en problemas geológicos como la localización de fallas verticales, es común trabajar con datos dispersos e irregulares, donde la reconstrucción de discontinuidades requiere enfoques adaptativos (3), (5).

El núcleo de esta problemática radica en la detección precisa o aproximada de discontinuidades al procesar datos de funciones no regulares, ya que su correcta identificación permite diseñar estrategias para aproximar la función subyacente mientras se mitigan oscilaciones no deseadas, como el fenómeno de Gibbs. Este último surge al aplicar métodos clásicos —diseñados para funciones regulares— a problemas con discontinuidades, generando oscilaciones espurias en las aproximaciones. Por ello, una caracterización *a priori* de las discontinuidades es esencial para seleccionar técnicas adecuadas y mejorar la precisión en etapas posteriores.

En la literatura, destacan contribuciones como (6), (7), (8), quienes propusieron una metodología en cuatro etapas para detectar discontinuidades en funciones univariadas: Construcción de un espacio de aproximación con elementos finitos que incorporen discontinuidades; Ajuste por mínimos cuadrados a un conjunto de datos  $\mathbb{D} = \{(x_i, y_i) \in \mathbb{R}^2: i = 1, 2, \dots, N\}$ , donde  $f(x_i) = y_i$ , para una función  $f$  desconocida y discontinua; Detección de discontinuidades mediante el análisis de oscilaciones asociadas al fenómeno de Gibbs en los datos; y Localización precisa aplicando un análisis de error en forma de  $Z$  ( $Z$  – forma) a los candidatos identificados.

Más recientemente, (9) desarrollaron un marco teórico para detectar y aproximar discontinuidades en funciones explícitas no regulares a partir de datos dispersos de tipo Lagrange, extendiendo el método de (10) para fallas verticales. Su enfoque genera una curva poligonal que aproxima el conjunto de discontinuidades y propone un cubrimiento mediante discos cuyos radios dependen de la posición de los centros en dicha curva. Si bien el trabajo teórico se presenta en (9), la implementación computacional se detalla en (11), marcando un avance en la aplicabilidad de estas técnicas.

Estos métodos abren caminos para aplicaciones en geofísica, ingeniería y ciencias de materiales, donde la modelización de discontinuidades es crucial. No obstante, persisten desafíos como la generalización a dimensiones superiores, la integración de incertidumbre en datos medidos y la optimización computacional. Futuras investigaciones podrían explorar hibridaciones con aprendizaje automático o técnicas geoestadísticas para mejorar la robustez en escenarios reales.

Por tanto, el propósito de la presente investigación fue analizar la metodología de construcción de familia de aproximantes para localizar fallas verticales latentes en superficies discontinuas, a partir de un conjunto de datos sintéticos irregulares y dispersos de tipo Lagrange, con la finalidad de estudiar su convergencia y validar su consistencia a través del lenguaje de programación *Python*. Para lograrlo se realizó lo siguiente: Estudiar la aproximación de discontinuidades en funciones no regulares, definidas explícitamente por  $z = f(x, y)$ ; Construir una secuencia de curvas poligonales que aproximan y localicen la falla vertical estudiada; Analizar la convergencia de estas aproximaciones para la localización fallas verticales latentes en superficies discontinuas; y, Desarrollar una metodología para detectar fallas verticales latentes en superficies discontinuas, que utiliza datos dispersos de tipo Lagrange.

## MATERIALES Y MÉTODOS

Para esta investigación corresponde a un estudio aplicado y computacional, con enfoque cuantitativo y diseño experimental - sintético, dado que se trabajó con datos generados de forma artificial. De este modo, este apartado está dedicado a presentar los materiales y métodos utilizados para lograr alcanzar los objetivos formulados en la presente investigación.

## Materiales

Para efectuar el presente estudio, se utilizó una implementación computacional dentro del entorno de trabajo de *Google Colab*. Esta plataforma es un servicio gratuito en la nube de *Google* que permite ejecutar y programar en *Python* directamente desde el navegador, sin necesidad de configurar un entorno local. Este lenguaje muy popular, fue la herramienta más relevante utilizada para la consolidación de los resultados encontrados con respecto a la detección y convergencia de aproximantes para fallas verticales latentes en superficies discontinuas, para después realizar sus respectivos análisis. Específicamente, para el procesamiento, análisis y visualización de los datos sintéticos Lagrange, se aprovecharon librerías de código abierto como *NumPy* (cálculos numéricos), *SciPy* (manipulación) y *Plotly* (gráficos interactivos)

## Métodos

En un primer momento, se utilizó el algoritmo de (10) para detectar las líneas de falla de un conjunto de datos dispersos. Específicamente, esta metodología se esquematiza de la siguiente manera<sup>3</sup>:

$E_1$ ) Detección y aproximación de la localización de fallas verticales latentes en superficies discontinuas.

$P_{1,1}$ ) Localización de nodos cerca de una curva de falla vertical.

$P_{1,1,1}$ ) Generación de una malla triangular soportada por el conjunto  $X^N$ .

$P_{1,1,2}$ ) Clasificación de nodos en el conjunto de datos  $\mathcal{D}$  cerca de una curva de falla.

$P_{1,1,3}$ ) Separación de valores altos y bajos de los nodos en  $\mathcal{R}$ .

$P_{1,2}$ ) Cálculo de la curva poligonal aproximante  $\Gamma_1^p$  de  $\Gamma$ .

$E_{1,R}$ ) Aproximación de la curva poligonal que representa la falla vertical  $\Gamma$ .

En un segundo momento, se utilizó la técnica de (12) y (13) para la construcción de familias de aproximantes para fallas verticales latentes en superficies discontinuas:

$E_2$ ) Construcción de la familia de aproximantes para fallas verticales latentes en superficies discontinuas.

$P_{2,1}$ ) Generación de una malla triangular refinada soportada en el conjunto  $X^N \cup P_1^M$ .

<sup>3</sup>  $E_i$  denota la etapa  $i$  de la metodología de detección del conjunto de discontinuidad  $\mathcal{D}$ ; mientras que  $P_{i,j}$   $E_i$  denota  $E_i$  denota la etapa  $i$  de la metodología de detección del conjunto de discontinuidad  $\mathcal{D}$ ; mientras que  $P_{i,j}$  correspondiente al paso  $j$  de la etapa  $i$  del mismo método; y asimismo,  $P_{i,j,k}$  significa el subpaso  $k$  del paso  $j$  de la etapa  $i$ . Además,  $E_{i,R}$  significa el resultado obtenido en la etapa  $i$ .

$P_{n,2}$ ) Identificación y separación de valores altos y bajos de los nodos en  $\mathcal{R} \cup P_{n-1}^M$ .

$P_{n,3}$ ) Identificación de los triángulos divisibles refinados en la malla triangular generada a partir del conjunto  $X^N \cup P_{n-1}^M$ .

$P_{n,4}$ ) Determinación del conjunto  $P_n^M$  de puntos medios.

$P_{n,5}$ ) Cálculo de la curva poligonal aproximante  $\Gamma_n^p$  de  $\Gamma$ .

$E_{n,R}$ ) Aproximación de la curva poligonal que representa la falla vertical  $\Gamma$ .

La implementación se realizó en Python utilizando Google Colab como entorno de desarrollo, donde el código se ejecutó en entorno basado en Jupyter Notebook, accesible desde la nube. Se debe mencionar que los datos sintéticos se generaron a partir de la simulación de una curva de falla polinómica de sexto grado en el intervalo 2 a 16 en el dominio. Los puntos se distribuyeron de forma aleatoria alrededor de la curva, en el que se generaron 32 nodos equiespaciados en 0.2 unidades. Además, se realizaron 7 iteraciones, donde el criterio de parada que se utilizó es un tipo de convergencia visual entre la curva poligonal y la curva de falla real. Como criterio numérico, se consideró que la distancia máxima entre puntos medios consecutivos en iteraciones sucesivas fuera menor a  $\varepsilon=0.01$ .

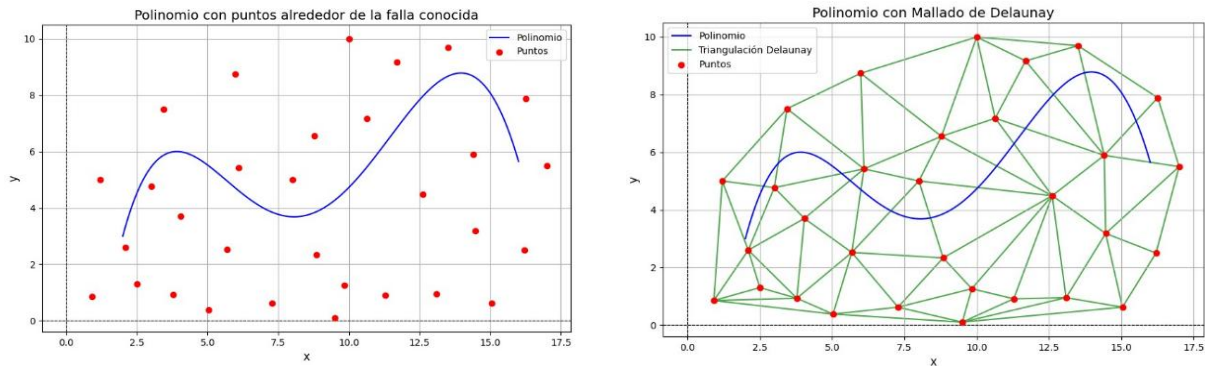
Adicionalmente, la metodología propuesta en esta investigación para la construcción de familias de aproximantes destinadas a la detección y aproximación de fallas verticales latentes en superficies discontinuas se complementa con los resultados de (9) y (11). La articulación de estos referentes teóricos, bajo supuestos razonables sobre el conjunto de datos considerado, permitió fortalecer el análisis de convergencia de dichas aproximantes para la detección de fallas verticales en superficies discontinuas.

## RESULTADOS

En esta sección se exponen los principales resultados obtenidos a través de una implementación computacional en *Python* para la detección de fallas verticales latentes en superficies discontinuas y el análisis de convergencia de sus aproximantes. Estos hallazgos permiten un análisis posterior que valida teórica y numéricamente la convergencia de la familia de aproximantes propuesta, sustentando así su eficacia.

Para iniciar la implementación numérica en *Python*, se utilizaron como datos de entrada un conjunto de datos sintéticos dispersos e irregularmente distribuidos de tipo Lagrange, denotado por:  $\mathfrak{D} = \{(\xi_j, f(\xi_j)) \in \mathbb{R}^2 \times \mathbb{R} : j = 1, 2, \dots, 32\}$ , con nodos  $\xi_j = (x_j, y_j) \in X^{32} \subset \mathbb{R}^2$ , compuesto por 12 muestras. En este sentido, en la Figura 1 se muestra el conjunto de nodos correspondiente al conjunto de datos  $\mathfrak{D}$ .

Iniciada la implementación numérica, los datos  $\mathfrak{D}$  ingresan en la etapa  $E_1$ ) de *Detección y aproximación de la localización de fallas verticales latentes en superficies discontinuas*. Específicamente, el primer paso  $P_{1,1}$ ) *Localización de nodos cerca de una curva de falla vertical*, se subdivide en tres subpasos tal y como se muestra en el esquema presentado en la sección de materiales y métodos. Para ello, se parte del conjunto de nodos de  $\mathfrak{D}$ , representado en la Figura 1. Así, el subpaso  $P_{1,1,1}$ ) *Generación de una malla triangular soportada por el conjunto  $X^N$* , se ilustra en la Figura 2 mediante una triangulación de Delaunay.



**Figura 1.** Conjunto de nodos correspondientes al conjunto  $\mathfrak{D}$  y curva de falla  $\Gamma$ .

**Figura 2.** Triangulación  $\mathcal{T}(X^{32})$  basada en el conjunto de nodos de  $\mathfrak{D}$ .

Seguidamente, se realiza el subpaso  $P_{1,1,2}$ ) *Clasificación de nodos en el conjunto de datos  $\mathfrak{D}$  cerca de una curva de falla*, en el cual se identifican los nodos interiores de la triangulación  $\mathcal{T}(X^{32})$  que están cercanos a la curva de falla desconocida  $\Gamma$ , con el propósito de coleccionarlos en el conjunto  $\mathcal{R}$ . Posteriormente, se ejecuta el subpaso  $P_{1,1,3}$ ) *Separación de valores altos y bajos de los nodos en  $\mathcal{R}$* , clasificando estos nodos en valores altos y bajos.

Dando continuidad al proceso, a partir del conjunto  $\mathcal{R}$ , se procede al segundo paso  $P_{1,2}$ ) *Cálculo de una curva poligonal aproximante  $\Gamma_1^p$  de  $\Gamma$*  que representa la falla vertical, enfocado en el cálculo de la aproximante  $\Gamma_1^p$ . En particular, al disponer en el conjunto  $\mathcal{R}$  de los nodos cercanos a la curva de falla vertical  $\Gamma_p$ , es posible identificar los triángulos involucrados, conocidos como *triángulos divisibles*. Estos triángulos permiten calcular e identificar los puntos medios entre los valores altos y bajos en cada uno de ellos.

Esta etapa se concreta con la obtención de la curva poligonal aproximante  $\Gamma_1^p$  de  $\Gamma$ , mediante la unión consecutiva de los puntos medios de los triángulos divisibles. Finalmente, la etapa  $E_{1,R}$ ) es la *Aproximación de la curva poligonal que representa la falla vertical  $\Gamma$* , se concreta con la obtención de la curva poligonal aproximante. El resultado de todo este proceso de la etapa 1 se ilustra en la Figura 3.

Dando continuidad a la construcción de la segunda curva poligonal aproximante a través de la implementación de la etapa 2,  $E_2$ ) *Construcción de la familia de aproximantes para fallas*

verticales latentes en superficies discontinuas, se procede secuencialmente con los siguientes pasos:  $P_{2,1}$ ) Generación de una malla triangular refinada soportada en el conjunto  $X^N \cup P_1^M$ ;  $P_{2,2}$ ) Identificación y separación de valores altos y bajos de los nodos en  $\mathcal{R} \cup P_1^M$ ;  $P_{2,3}$ ) Identificación de los triángulos divisibles refinados en la malla triangular generada a partir del conjunto  $X^N \cup P_1^M$ ;  $P_{2,4}$ ) Determinación del conjunto  $P_2^M$  de puntos medios;  $P_{2,5}$ ) Cálculo de la curva poligonal aproximante  $\Gamma_2^P$  de  $\Gamma$ . En definitiva, después de todo este proceso se obtiene la  $E_{2,R}$ ) Aproximación de la curva poligonal que representa la falla vertical  $\Gamma$ . La Figura 4 muestra el resultado obtenido al completar la segunda etapa del proceso.

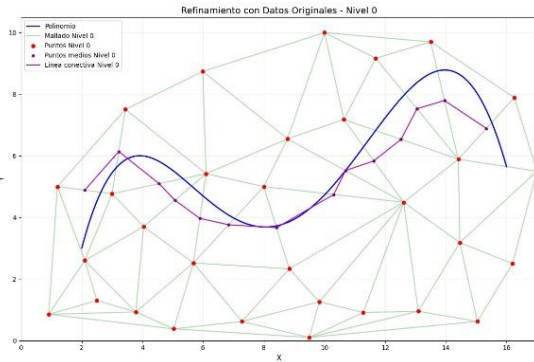


Figura 3. Curva poligonal aproximante  $\Gamma_1^P$  de la falla vertical  $\Gamma$ .

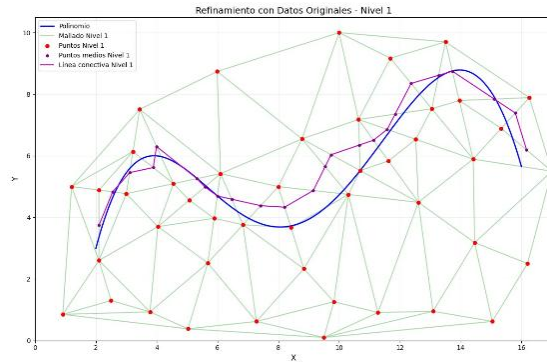


Figura 4. Curva poligonal aproximante  $\Gamma_2^P$  de la falla vertical  $\Gamma$ .

Continuando este proceso iterativamente a través de la etapa  $n$ , para la  $E_n$ ) Construcción de la familia de aproximantes para fallas verticales latentes en superficies discontinuas, se procede a realizar nuevamente cada uno de los siguientes pasos:  $P_{n,1}$ ) Generación de una malla triangular refinada soportada en el conjunto  $X^N \cup P_{n-1}^M$ ;  $P_{n,2}$ ) Identificación y separación de valores altos y bajos de los nodos en  $\mathcal{R} \cup P_{n-1}^M$ ;  $P_{n,3}$ ) Identificación de los triángulos divisibles refinados en la malla triangular generada a partir del conjunto  $X^N \cup P_{n-1}^M$ ;  $P_{n,4}$ ) Determinación del conjunto  $P_n^M$  de puntos medios;  $P_{n,5}$ ) Cálculo de la curva poligonal aproximante  $\Gamma_n^P$  de  $\Gamma$ ;  $E_{n,R}$ ) Aproximación de la curva poligonal que representa la falla vertical  $\Gamma$ . Finalmente, después de todo este proceso se obtiene la  $E_{2,R}$ ) Aproximación de la curva poligonal que representa la falla vertical  $\Gamma$ . En las Figuras 5 ( $n = 3$ ), 6 ( $n = 4$ ), 8 ( $n = 5$ ), 10 ( $n = 6$ ) y 12 ( $n = 7$ ) se muestran los resultados obtenidos al completar las etapas  $E_2, E_3, E_4, E_5$  y  $E_6$ , respectivamente.

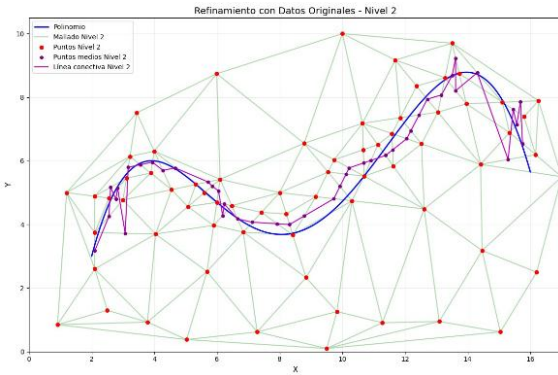


Figura 5. Curva poligonal aproximante  $\Gamma_3^p$  de la falla vertical  $\Gamma$ .

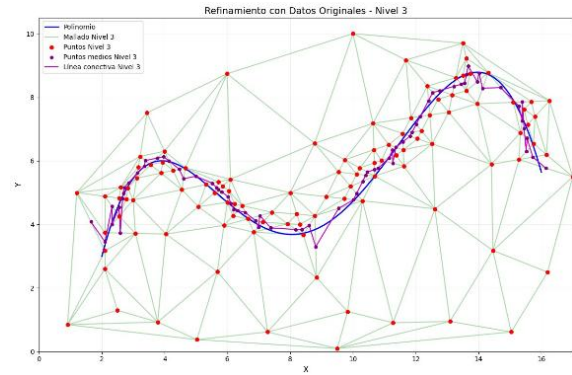


Figura 6. Curva poligonal aproximante  $\Gamma_4^p$  de la falla vertical  $\Gamma$ .

En la Figura 7, se muestra una comparación entre la curva de falla vertical  $\Gamma$  en cuestión y la curva poligonal aproximante  $\Gamma_4^p$  construida con la metodología estudiada.

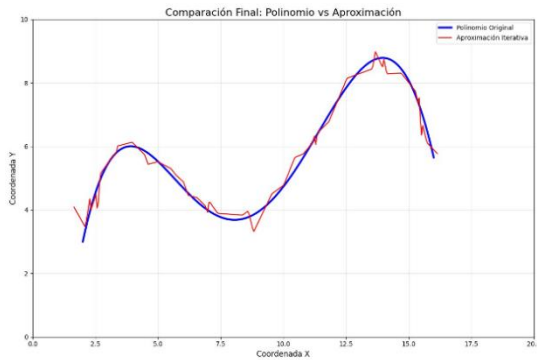


Figura 7. Comparación entre la curva de falla vertical  $\Gamma$  y la aproximante  $\Gamma_4^p$ .

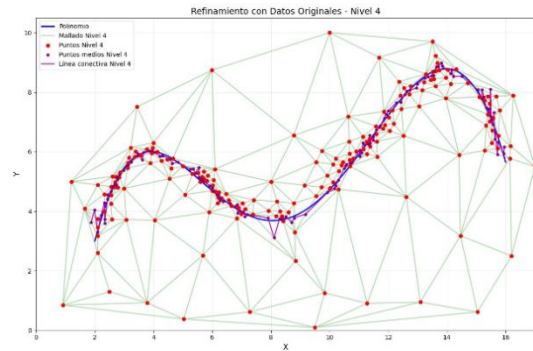


Figura 8. Curva poligonal aproximante  $\Gamma_5^p$  de la falla vertical  $\Gamma$ .

En la Figura 9, se muestra una comparación entre la curva de falla vertical  $\Gamma$  en cuestión y la curva poligonal aproximante  $\Gamma_4^p$  construida con la metodología estudiada.

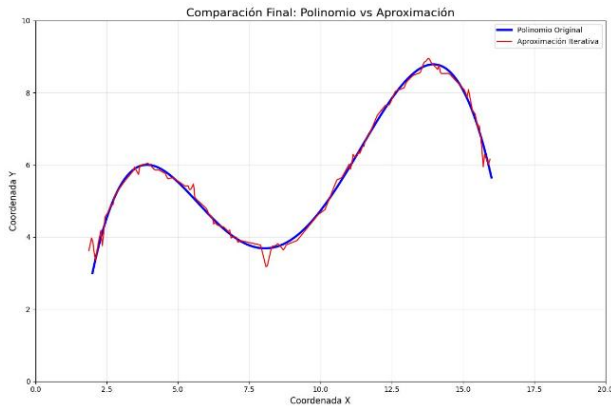


Figura 9. Comparación entre la curva de falla vertical  $\Gamma$  y la aproximante  $\Gamma_5^p$ .

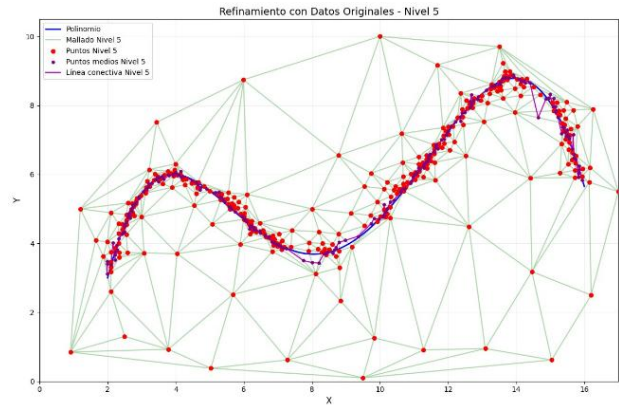


Figura 10. Curva poligonal aproximante  $\Gamma_6^p$  de la falla vertical  $\Gamma$ .

En la Figura 11, se muestra una comparación entre la curva de falla vertical  $\Gamma$  en cuestión y la curva poligonal aproximante  $\Gamma_6^p$  construida con la metodología estudiada.

En la Figura 12, presenta la curva poligonal aproximante  $\Gamma_7^p$  correspondiente al séptimo nivel de refinamiento.

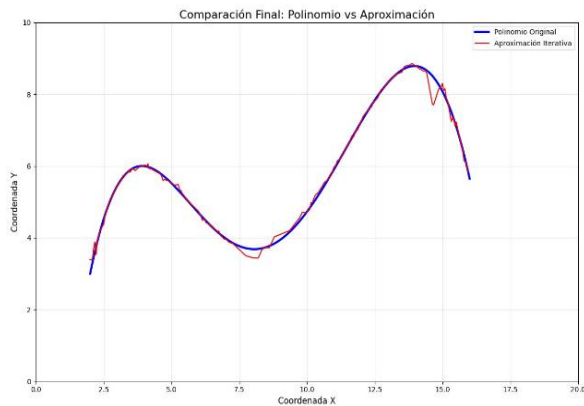


Figura 11. Comparación entre la curva de falla vertical  $\Gamma$  y la aproximante  $\Gamma_6^p$ .

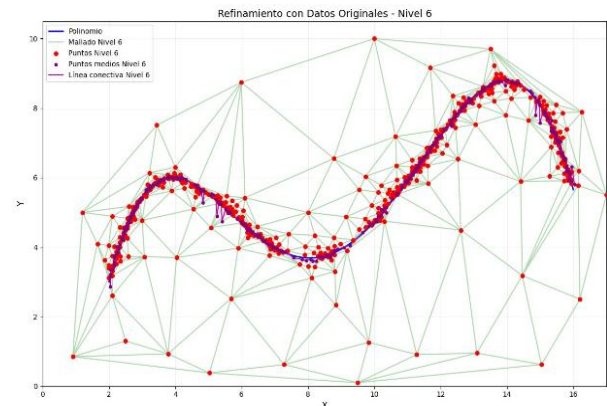


Figura 12. Curva poligonal aproximante  $\Gamma_7^p$  de la falla vertical  $\Gamma$ .

En la Figura 13, se muestra una comparación entre la curva de falla vertical  $\Gamma$  en cuestión y la curva poligonal aproximante  $\Gamma_7^p$  construida con la metodología estudiada.

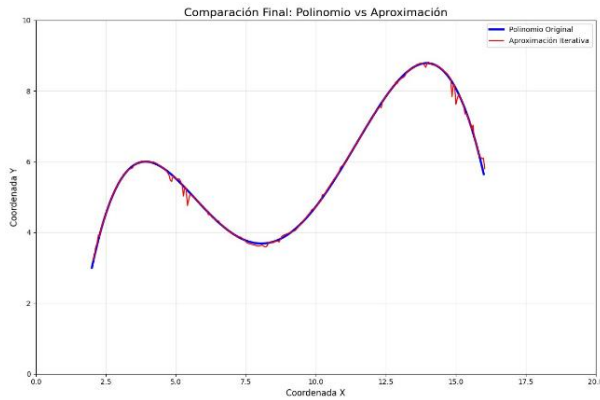


Figura 13. Comparación entre la curva de falla vertical  $\Gamma$  y la aproximante  $\Gamma_7^p$ .

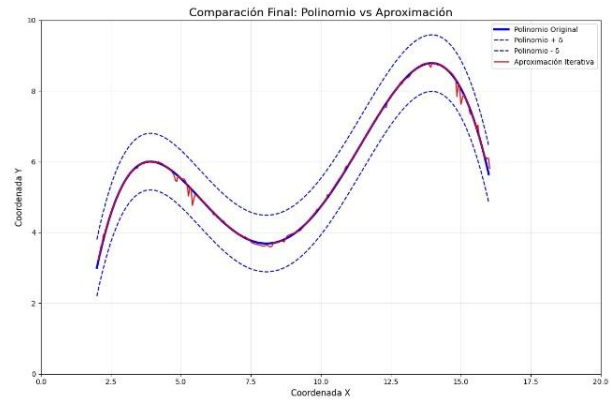


Figura 14. Ilustración de la convergencia de la aproximante  $\Gamma_7^p$  hacia la curva de falla vertical  $\Gamma$ .

## DISCUSIÓN

Este apartado está dedicado a discusión de los resultados obtenidos en la sección anterior. Específicamente, se comienza presentando el análisis de cada una de las gráficas resultantes de la implementación numérica en *Python*. Seguidamente, se detallan algunas observaciones clave relacionadas con los hallazgos.

En la Figura 1 se considera una curva de falla  $\Gamma$  (color azul) que se desea aproximar. Específicamente, es un polinomio de sexto grado el cual toma valores en el eje  $x$  desde 2 hasta 16. También se encuentra un conjunto de nodos (color rojo) sintéticos y dispersos tomados aleatoriamente alrededor de la curva de falla  $\Gamma$ . Tanto la curva como los nodos corresponden a la proyección vertical de la falla en cuestión y el conjunto de datos  $\mathfrak{D}$  sobre el plano  $xy$ , respectivamente.

Para obtener la gráfica de la Figura 2 se utilizó la triangulación de *Delaunay* con el propósito de crear un espacio convexo alrededor de la falla para poder realizar un estudio más profundo de esta. Específicamente, se realizó un mallado con todos los nodos que se habían proporcionado. Para ello, se utilizó la librería *scipy.spatial.Delaunay* de *Phyton*, la cual es esencial para realizar este mallado triangular (14).

Después de la triangulación  $\mathcal{T}(X^{32})$  realizada a partir del conjunto de nodos correspondientes a los datos  $\mathfrak{D}$ , se obtuvieron los nodos con valores bajos y altos, dando paso a la identificación de los triángulos divisibles. Este procedimiento permitió identificar los lados de los triángulos donde debían calcularse los puntos medios (representados por puntos morados) en cada elemento triangular involucrado. Seguidamente, se procede a unir estos nuevos puntos obtenidos a través de segmentos y de forma continua para obtener la poligonal  $\Gamma_1^p$  que se acerca a la falla original  $\Gamma$ , como se muestra en la Figura 3. Se observa que, los lados de los

triángulos divisibles se interceptan con la curva aproximada  $\Gamma$ , garantizando que el proceso de construcción de aproximantes es consistente. Esta nueva información emergente a partir de los datos iniciales es de suma relevancia, puesto que van a servir de base para realizar la siguiente iteración, es decir, la siguiente triangulación para obtener la siguiente curva poligonal  $\Gamma_2^p$ .

En las siguientes iteraciones, también llamadas *niveles*, se procede a repetir el mismo proceso anterior, donde se va a ir observando un refinamiento de malla original, principalmente alrededor de la curva de falla  $\Gamma$ . Además, en cada nivel van a existir más puntos, los cuales se originan a partir de los puntos medios de los lados de los triángulos divisibles. Su concatenación continua por segmentos de rectas sirve construir la curva poligonal aproximante en ese nivel.

Así, continuando el proceso, para obtener la gráfica de la Figura 4 se realizó el mismo procedimiento descrito para la Figura 3, donde se obtiene una nueva curva poligonal aproximante  $\Gamma_2^p$  que está más acerca a la curva de falla  $\Gamma$ . De la misma manera, sucede con las gráficas de las Figuras 5, 6, 8, 10 y 12. Al respecto, se debe resaltar que una nueva curva poligonal aproximante va a depender de la poligonal anterior, es decir, esto es un proceso recursivo. Es así como, el proceso de construcción de aproximantes puede seguir hasta cuantas veces sea necesario. En ese trabajo, este proceso se realizó hasta en nivel 7, con la salvedad que desde el nivel 4 se procede a representar gráficamente solo la falla original  $\Gamma$  y la curva poligonal aproximante  $\Gamma_j^p$ , para  $j = 4, 5, 6, 7$ . Lo importante es que cada nueva curva poligonal construida va a converger a la falla  $\Gamma$  en estudio. Esto último, se puede observar en la Figuras 7, 9, 11 y 13, donde se realiza una comparación entre la curva de falla vertical  $\Gamma$  y la aproximante  $\Gamma_j^p$ , para  $j = 4, 5, 6, 7$ .

Finalmente, en la Figura 14 se observa que a partir de cierto nivel ( $n \geq N$ ) la curva de aproximante  $\Gamma_n^p$  esta tan cerca de la curva de falla  $\Gamma$  como se quiera (dado  $\varepsilon > 0$ ), incluyendo las pequeñas oscilaciones que aparecen durante el proceso en los primeros niveles. Esto es precisamente, que existe convergencia de la familia de aproximantes  $\{\Gamma_n^p\}_{n=1}^{\infty}$ , es decir,  $\Gamma_n^p \rightarrow \Gamma$  cuando  $n \rightarrow \infty$ . Más aun, la convergencia es uniforme.

## CONCLUSIONES

El estudio de la convergencia de las familias de aproximantes basada en la metodología para la detección de fallas verticales latentes en superficies discontinuas explícitas demostró ser efectiva y consistente, siendo el aporte más destacado de esta investigación. En este contexto, las pruebas realizadas con datos sintéticos dispersos e irregulares de tipo Lagrange, utilizando el lenguaje de programación *Python* generaron las siguientes consideraciones: primero, la implementación en *Python* en cada una de las etapas de la metodología fue exitosa, permitiendo una transición fluida y efectiva desde el enfoque teórico hacia el numérico; segundo, tanto el mallado triangular inicial como cada uno de sus refinamientos en los distintos niveles, cristalizaron la construcción de una secuencia de curvas poligonales aproximantes óptimas, es decir, una por cada iteración; tercero, las curvas poligonales resultantes aproximan y localizan la falla vertical estudiada; y, cuarto, el análisis de convergencia de estas

aproximaciones demostró buenos resultados, más aún, existe convergencia uniforme de manera efectiva.

En términos generales, la metodología propuesta para la detección y convergencia de fallas verticales latentes en superficies discontinuas sienta las bases para construir aproximaciones óptimas. Donde, estas aproximaciones son lo suficientemente flexibles para adaptarse a los requisitos de entrada necesarios para su futura implementación numérica y computacional en algoritmos enfocados al proceso de aproximación de funciones no regulares.

Esta investigación evidencia la viabilidad del método propuesto para la detección y convergencia de aproximantes en fallas verticales latentes, aunque se reconoce que el uso exclusivo de datos sintéticos limita su validación práctica. Futuras investigaciones deberían incorporar datos reales, preferiblemente de campo o laboratorio, para evaluar el comportamiento del algoritmo en condiciones más heterogéneas y con ruido experimental. Asimismo, se sugiere explorar técnicas híbridas que combinen diferentes enfoques de interpolación y optimización, junto con un análisis riguroso de sensibilidad, para mejorar la precisión en la identificación de discontinuidades. Estos avances no solo fortalecerían el modelo, sino que también ampliarían su aplicabilidad en problemas geofísicos, estructurales o industriales donde las fallas latentes juegan un papel crítico.

Como perspectiva futura, se recomienda un análisis de los errores sistemáticos y estocásticos en el proceso de detección y aproximación de fallas, considerando tanto su origen como su propagación a lo largo del algoritmo. Sería valioso desarrollar un marco cuantitativo que discrimine entre errores inherentes a la metodología y aquellos asociados a la naturaleza de las discontinuidades, posiblemente mediante técnicas de incertidumbre como intervalos de confianza o métodos estocásticos. Esto permitiría no solo optimizar los parámetros del modelo, sino también establecer límites de confiabilidad para su aplicación en escenarios reales, donde la presencia de ruido y datos incompletos es habitual.

## **AGRADECIMIENTOS**

Este trabajo agradece el respaldo brindado por los Programas de Maestría en Matemática aplicada con mención en Matemática Computacional de la Universidad Nacional de Chimborazo (UNACH), el Doctorado en Matemáticas de la Universidad Simón Bolívar (USB) y la carrera de Matemáticas de la Escuela Superior Politécnica de Chimborazo (ESPOCH), fundamentales para la realización de esta investigación.

## **DECLARACIÓN DE INTERÉS**

Los autores declaran que no existe conflicto de interés posible.

## REFERENCIAS BIBLIOGRÁFICAS

1. Mallat S, Hwang. WL. Singularity detection and processing with wavelets. *IEEE Trans. Inform. Theory*, 1992; 38(2): 617–643.
2. Vasilenko VA, Rozhenko AI. Discontinuity localization and spline approximation of discontinuous functions with many variables scattered meshes. *Sov. J. Numer. Anal. Math. Modelling*, 1990; 5(4): 425-434
3. Parra MC. Sobre detección de discontinuidades y aproximación de funciones no regulares. Tesis de Doctorado. 1999. España: Universidad de Zaragoza.
4. Bozzini M, Rossini M. The detection and recovery of discontinuity curves from scattered data. *Journal of Computational and Applied Mathematics*, 2013. 240: 148 – 162. <https://doi.org/10.1016/j.cam.2012.06.014>
5. Parra M, López M, Torrens J. Vertical fault detection from scattered data. *Journal of Computational and Applied Mathematics*, 1996; 73(5): 225–239.
6. Palma PF, Gallo R, Manzanilla R. Detection of Discontinuity Points in one Variable Functions using Spaces of Trigonometric Functions. *Bull. Comput. Appl. Math. (CompAMa)*, 2021; 9(2): 9-22. Hal-03566889. DOI: <https://hal.science/hal-03566889>
7. Lino VA, Gallo R, Manzanilla R. Detección de discontinuidades en funciones de una variable utilizando espacios de tipo elementos finitos. *Revista Bases de la Ciencia*, 2022; 7, (Especial), Diciembre, 135-152. DOI: <https://doi.org/10.33936/revbasdelaciencia.v7iESPECIAL.4148>
8. Córdova HJ, Manzanilla R, Gallo R. Aproximación de fallas, utilizando el método de elementos finitos no conformes. *Revista Bases de la Ciencia*, 2022; 7(Especial): 166–183. DOI: <https://doi.org/10.33936/revbasdelaciencia.v7iESPECIAL.4247>
9. Abancín RA, Dávalos M, Morocho J. Detection and coverage of polygonal curves derived from vertical faults present on discontinuous surfaces. *Bull. Comput. Appl. Math. (CompAMa)*. 2024; 12(2): 1 – 25. Disponible en: [https://drive.google.com/file/d/1anbfrg7zt-YGo3TaAO\\_niPRt7ncGNA3f/view](https://drive.google.com/file/d/1anbfrg7zt-YGo3TaAO_niPRt7ncGNA3f/view)
10. Gutzmer T, Iske A. Detection of discontinuities in scattered data approximation. *Numerical Algorithms*, 1997. 16(2): 155–170.
11. Abancín RA, Parra M. Numerical implementation of an algorithm for detection and covering of latent vertical faults in discontinuous surfaces. In: VI Jornadas Ecuatorianas de Matemáticas (JEM); 2024 Nov 25-29; Riobamba, Ecuador: Escuela Superior Politécnica de Chimborazo (ESPOCH).
12. Abancín RA, Paredes J. Detection and convergence of approximants for latent vertical faults in discontinuous surfaces. In: VI Jornadas Ecuatorianas de Matemáticas (JEM); 2024 Nov 25-29; Riobamba, Ecuador: Escuela Superior Politécnica de Chimborazo (ESPOCH).
13. Abancín RA. Algorithm for the detection and covering of latent vertical faults in discontinuous surfaces. In: VI Jornadas Ecuatorianas de Matemáticas (JEM); 2024 Nov 25-29; Riobamba, Ecuador: Escuela Superior Politécnica de Chimborazo (ESPOCH).
14. Parra Chávez MV, Abancin Ospina RA. Implementación numérica del algoritmo de detección y cubrimiento de fallas verticales latentes en superficies discontinuas. *Revista Científica Multidisciplinaria InvestiGo*. 2024; 5(13): 269-280.