



Estudio numérico de esquemas de interpolación y métodos de resolución de ecuaciones no lineales

Tobio Santiago¹, Wasinger Pablo²

Departamento de Ingeniería, Universidad de San Andrés, Victoria, Buenos Aires, Argentina.

E-mails: ¹stobio@udesa.edu.ar, ²pwasinger@udesa.edu.ar

Abstract: Este trabajo evalúa el rendimiento de diversos métodos de interpolación numérica y resolución de ecuaciones no lineales, aplicados a funciones de una y dos variables, y a un problema de trilateración. Se analizaron métodos como interpolación de Lagrange, splines cúbicos, interpolación bilineal y bicúbica, variando el número y distribución de nodos. También se exploró el método de Newton-Raphson para resolver sistemas no lineales. Los resultados mostraron que los splines cúbicos y la interpolación bicúbica ofrecen la mejor precisión en una y dos dimensiones respectivamente, mientras que los nodos de Chebyshev mitigaron eficazmente el fenómeno de Runge. El experimento de trilateración reveló la sensibilidad del método de Newton-Raphson al punto inicial y la presencia de múltiples mínimos locales.

Keywords: Métodos numéricos, interpolación, Polinomio de Lagrange, Método de Newton-Raphson, Splines cúbicos.

1 Introducción

En numerosos campos de la ciencia y la ingeniería, la comprensión y modelización de fenómenos complejos es fundamental.

Frecuentemente, al estudiar el comportamiento de un fenómeno de interés, nos gustaría poder describirlo con una función matemática que dependa de una o varias variables. Sin embargo, en la mayoría de los casos, solo contamos con un conjunto de datos discretos acerca del comportamiento de nuestro objeto de estudio a través del tiempo o el espacio.

Por este motivo, es necesario contar con métodos de **interpolación** que nos provean de una función matemática a partir de estos datos. El objetivo es aproximar el comportamiento de nuestro objeto de estudio en regiones sobre las que no tenemos mediciones directas. Métodos como la interpolación polinómica de Lagrange, los splines cúbicos o la interpolación bilineal son herramientas fundamentales en este proceso.

En algunos casos, los datos que obtenemos no nos proveen información directa, pero quizás esta información es útil para resolver ecuaciones que gobiernan el comportamiento de nuestro objeto de estudio. No obstante, las soluciones a estas ecuaciones pueden ser muy difíciles de encontrar analíticamente, por lo que es necesario recurrir a métodos numéricos para hallarlas como el de Newton-Raphson.

En el presente trabajo se realizan experimentos numéricos con el fin de evaluar la *performance* de distintos esquemas de interpolación bajo la variación de hiperparámetros como la cantidad de puntos seleccionados, la distancia entre los puntos o la cantidad de variables de la función. El objetivo principal de este informe es determinar qué métodos de interpolación son más efectivos en diferentes escenarios y cómo estos hiperparámetros afectan la precisión de las aproximaciones.

2 Marco teórico

A lo largo de esta sección, no solo detallaremos los métodos utilizados en nuestros experimentos numéricos, sino que también proporcionaremos el contexto teórico necesario para entender por qué estos métodos son efectivos, bajo que condiciones son aplicables y qué precauciones deben tomarse al utilizarlos. Este marco teórico servirá como base para interpretar los resultados de nuestros experimentos.

2.1 Método de interpolación lineal

La interpolación lineal es uno de los métodos más simples de interpolación. Este método consiste en conectar dos puntos de datos conocidos con una línea recta y usar esta línea para estimar los valores intermedios. Dados dos puntos (x_0, y_0) y (x_1, y_1) , la interpolación lineal asume que la función entre estos dos puntos es una línea recta. La ecuación de esta línea se puede expresar como:

$$y = y_0 + (x - x_0) \frac{y_1 - y_0}{x_1 - x_0} \quad (1)$$

Esta ecuación se deriva de la fórmula general de una línea recta, $y = mx + b$, donde m es la pendiente y b es la intersección con el eje y . En nuestro caso:

1. La pendiente m se calcula como el cambio en y dividido por el cambio en x :
2. $m = \frac{y_1 - y_0}{x_1 - x_0}$
3. Para encontrar b , usamos el punto (x_0, y_0) :
 $y_0 = mx_0 + b \Rightarrow b = y_0 - mx_0$
4. Sustituyendo estos valores en la ecuación general:
 $y = mx + b \Rightarrow y = \frac{y_1 - y_0}{x_1 - x_0} x + (y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0)$
5. Simplificando:
 $y = \frac{y_1 - y_0}{x_1 - x_0} x + y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0 \Rightarrow y = y_0 + \frac{y_1 - y_0}{x_1 - x_0} (x - x_0)$

Para cualquier valor x entre x_0 y x_1 , podemos usar esta ecuación para estimar el valor correspondiente de y .

La interpolación lineal tiene la ventaja de ser computacionalmente eficiente. Además, siempre produce una función continua en el intervalo de interpolación.

Sin embargo, este método también tiene limitaciones. Solo produce una aproximación de primer orden, lo que significa que la precisión puede ser baja, especialmente para funciones no lineales. La función interpolada no es suave en los puntos de datos (es decir, no tiene derivadas continuas en estos puntos) y no captura bien el comportamiento de funciones con curvatura significativa.

En la práctica, la interpolación lineal se usa a menudo como un método simple y rápido cuando se necesita una estimación aproximada y la suavidad de la función no es crítica.

2.2 Método de interpolación de Lagrange

El método de interpolación de Lagrange es una técnica poderosa que nos permite construir un único polinomio que pasa exactamente

por un conjunto dado de puntos. Este método es particularmente útil cuando necesitamos aproximar una función compleja utilizando un polinomio simple. Dado un conjunto de $n+1$ puntos de la forma $(x_0, f(x_0)), \dots, (x_n, f(x_n))$, nuestro objetivo es construir un polinomio de grado n que pase por todos estos puntos. Para lograr esto, primero definimos un objeto matemático fundamental llamado *base de Lagrange*:

$$L_i(x) = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j}, \quad i = 0, 1, \dots, n \quad (2)$$

La base $L_i(x)$ de Lagrange es una función que cumple la siguiente propiedad: $L_i(x_i) = 1$ y $L_i(x_j) = 0$ para $i \neq j$, donde x_i, x_j son los puntos que buscamos interpolar. Utilizando estas bases, podemos ahora definir el **Polinomio de Lagrange**:

$$L(x) = \sum_{i=0}^n y_i L_i(x) \quad (3)$$

La construcción de este polinomio garantiza que pase por todos los puntos de interpolación, como podemos ver al evaluarlo en cualquier x_j :

$$\begin{aligned} L(x_j) &= y_0 \underbrace{L_0(x_j)}_{=0} + \dots + y_j \underbrace{L_j(x_j)}_{=1} \\ &+ \dots + y_n \underbrace{L_n(x_j)}_{=0} = y_j \end{aligned} \quad (4)$$

El método de interpolación de Lagrange también nos proporciona una cota teórica del error que cometemos al aproximar la función utilizando un polinomio de grado n . La demostración de la existencia de esta cota se encuentra en [1]:

$$f(x) = P(x) + \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i) \quad (5)$$

donde ξ es algún valor en el intervalo que contiene todos los puntos de interpolación.

2.3 Método de interpolación de Splines cúbicos

El método de interpolación mediante Splines cúbicos es una técnica de interpolación que ofrece una interesante alternativa frente al método Lagrange. En vez de utilizar un único polinomio de alto grado para interpolar todos los *nodos*, los Splines cúbicos dividen el intervalo de interpolación en sub-intervalos, utilizando un polinomio de tercer grado en cada uno de ellos.

Dado un conjunto de $n+1$ nodos $(x_0, f(x_0)), \dots, (x_n, f(x_n))$, donde $x_0 < x_1 < \dots < x_n$, el objetivo es construir una función $S(x)$ que sea continua en todo el intervalo $[x_0, x_n]$ y que cumpla las siguientes propiedades:

1. $S_j(x_j) = f(x_j)$ y $S_j(x_{j+1}) = f(x_{j+1})$, $\forall j = 0, 1, \dots, n-1$;
2. $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$, $\forall j = 0, 1, \dots, n-2$;
3. $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$, $\forall j = 0, 1, \dots, n-2$;
4. $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$, $\forall j = 0, 1, \dots, n-2$;
5. Se satisface una de las siguientes condiciones de contorno:
 - (a) $S''(x_0) = S''(x_n) = 0$ (frontera natural);
 - (b) $S'(x_0) = f'(x_0)$ y $S'(x_n) = f'(x_n)$ (frontera condicionada).

Para construir el sistema de ecuaciones, definimos cada spline cúbico $S_j(x)$ en el intervalo $[x_j, x_{j+1}]$ como:

$$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3 \quad (6)$$

donde a_j, b_j, c_j , y d_j representan los coeficientes a determinar en cada subintervalo. Las condiciones 1) y 2) imponen n ecuaciones cada una, mientras que las condiciones 3) y 4) aportan $n-1$ ecuaciones adicionales. Sumando las dos ecuaciones provenientes de las condiciones de contorno, obtenemos un total de $4n$ ecuaciones. Este sistema completo nos permite resolver las $4n$ incógnitas que constituyen los coeficientes de los splines en todos los subintervalos.

Por otro lado, es práctica común utilizar splines naturales ya que no requiere tener una estimación de la derivada de la función en la frontera como lo es en caso de los splines con frontera engrapada. Los *splines naturales* son un tipo de interpolación cúbica en los que se imponen las llamadas condiciones de contorno naturales, que establecen que la segunda derivada de la función spline en los extremos del intervalo es igual a cero. Es decir, en los puntos inicial y final, $S''(x_0) = 0$ y $S''(x_n) = 0$. Estas condiciones aseguran que la curva sea lineal en los extremos, proporcionando una interpolación suave y evitando comportamientos oscilatorios no deseados en los bordes.

2.4 Método de interpolación de Hermite

La interpolación de Hermite es una técnica que extiende la interpolación polinómica para incluir no solo los valores de la función en los puntos de interpolación, sino también los valores de sus derivadas. Este método permite construir un polinomio que coincide tanto con los valores de la función como con los valores de sus derivadas en los puntos dados, lo que resulta en una aproximación más precisa de la función original, especialmente cuando se dispone de información sobre las derivadas.

Dado un conjunto de $n+1$ puntos $(x_0, f(x_0), f'(x_0)), \dots, (x_n, f(x_n), f'(x_n))$, donde $f(x_i)$ representa el valor de la función en x_i y $f'(x_i)$ el valor de su primera derivada, el objetivo es construir un polinomio $H(x)$ de grado máximo $2n+1$ que satisfaga las siguientes condiciones:

$$H(x_i) = f(x_i) \quad \text{y} \quad H'(x_i) = f'(x_i), \quad \text{para } i = 0, 1, \dots, n \quad (7)$$

El polinomio de Hermite se construye utilizando una combinación de funciones base. Definimos dos tipos de funciones base:

$$h_{i,0}(x) = [1 - 2(x - x_i)L'_i(x_i)]L_i^2(x) \quad (8)$$

$$h_{i,1}(x) = (x - x_i)L_i^2(x) \quad (9)$$

donde $L_i(x)$ es la base de Lagrange definida anteriormente en la ecuación (2).

El polinomio de Hermite se expresa entonces como:

$$H(x) = \sum_{i=0}^n [f(x_i)h_{i,0}(x) + f'(x_i)h_{i,1}(x)] \quad (10)$$

La interpolación de Hermite ofrece varias ventajas significativas. En primer lugar, proporciona una mayor precisión al incorporar información sobre las derivadas, lo que permite capturar mejor el comportamiento local de la función. Además, el polinomio resultante es continuo y posee una primera derivada continua en los puntos de interpolación, lo que asegura una representación más suave de la función original. Otra ventaja importante es su flexibilidad, ya que permite incorporar información adicional sobre las derivadas cuando está disponible, mejorando así la calidad general de la interpolación.

Sin embargo, la interpolación de Hermite también presenta algunas limitaciones que deben ser consideradas. Un requisito fundamental es el conocimiento preciso de las derivadas; si esta información no está disponible o es inexacta, la calidad de la

interpolación puede verse significativamente afectada. Además, la construcción del polinomio de Hermite implica una mayor complejidad computacional en comparación con métodos más simples como la interpolación de Lagrange. Por último, al igual que otros métodos de interpolación polinómica de alto grado, la interpolación de Hermite puede sufrir del fenómeno de Runge cuando se utiliza un gran número de puntos, lo que puede resultar en oscilaciones no deseadas en la función interpolada.

2.5 Método de interpolación bilineal

La interpolación bilineal es una extensión de la interpolación lineal para funciones de dos variables. Este método se utiliza para estimar el valor de una función en un punto (x, y) dado, conociendo los valores de la función en los cuatro vértices de un rectángulo que contiene al punto.

Consideremos una función $f(x, y)$ y un punto (x, y) dentro del rectángulo formado por los puntos (x_1, y_1) , (x_2, y_1) , (x_1, y_2) y (x_2, y_2) , donde conocemos los valores de f en estos cuatro puntos. La interpolación bilineal se realiza en dos pasos:

1. Primero, se realizan dos interpolaciones lineales en la dirección x : $f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(x_1, y_1) + \frac{x - x_1}{x_2 - x_1} f(x_2, y_1)$
 $f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(x_1, y_2) + \frac{x - x_1}{x_2 - x_1} f(x_2, y_2)$
2. Luego, se realiza una interpolación lineal en la dirección y usando los resultados del paso anterior:
 $f(x, y) \approx \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y - y_1}{y_2 - y_1} f(x, y_2)$

Combinando estas ecuaciones, obtenemos la fórmula de interpolación bilineal:

$$f(x, y) \approx \frac{1}{(x_2 - x_1)(y_2 - y_1)} \times [f(x_1, y_1)(x_2 - x)(y_2 - y) + f(x_2, y_1)(x - x_1)(y_2 - y) + f(x_1, y_2)(x_2 - x)(y - y_1) + f(x_2, y_2)(x - x_1)(y - y_1)] \quad (11)$$

La interpolación bilineal es útil debido a que es continua en todo el dominio de interpolación, es lineal a lo largo de las líneas paralelas a los ejes x y y , no es una función lineal de x y y , sino que contiene términos de producto cruzado xy y preserva la exactitud para funciones lineales en x y y . Sin embargo, al igual que la interpolación lineal, la interpolación bilineal puede no capturar adecuadamente el comportamiento de funciones con alta curvatura o variaciones rápidas.

2.6 Método de interpolación bicúbica

La interpolación bicúbica es una técnica más avanzada que la interpolación bilineal para funciones de dos variables. Este método no solo considera los valores de la función en los cuatro vértices de un rectángulo que contiene al punto de interés, sino también las derivadas parciales de primer orden y la derivada cruzada en estos puntos.

Consideremos una función $f(x, y)$ y un punto (x, y) dentro del cuadrado unitario $[0, 1] \times [0, 1]$. La fórmula general para la interpolación bicúbica es:

$$f(x, y) \approx \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j \quad (12)$$

Los coeficientes a_{ij} se determinan resolviendo un sistema de ecuaciones lineales basado en 16 condiciones:

1. Los valores de f en los cuatro vértices: $f(x_0, y_0)$, $f(x_0, y_1)$, $f(x_1, y_0)$, $f(x_1, y_1)$
2. Las derivadas parciales $\frac{\partial f}{\partial x}$ en los cuatro vértices
3. Las derivadas parciales $\frac{\partial f}{\partial y}$ en los cuatro vértices

4. Las derivadas cruzadas $\frac{\partial^2 f}{\partial x \partial y}$ en los cuatro vértices

Estas 16 condiciones forman un sistema de ecuaciones lineales que puede expresarse matricialmente como:

$$\mathbf{A} = \mathbf{M}^{-1} \mathbf{F} \mathbf{M}^{-T} \quad (13)$$

donde \mathbf{A} es la matriz 4×4 de coeficientes a_{ij} , \mathbf{M} es una matriz 4×4 predefinida, y \mathbf{F} es una matriz 4×4 que contiene los valores de f y sus derivadas en los vértices del cuadrado unitario.

Una vez que se han determinado estos coeficientes, se puede evaluar la función interpolada en cualquier punto dentro del cuadrado unitario.

La interpolación bicúbica es útil debido a que produce una superficie más suave, con continuidad C^1 en todo el dominio. Captura mejor el comportamiento de funciones con curvatura significativa y preserva las derivadas en los puntos de la cuadrícula, lo que es importante en muchas aplicaciones físicas. Sin embargo, puede tener problemas de "overshooting" o "sobrestimación" (valores interpolados que exceden los valores originales), especialmente en regiones con saltos abruptos o cambios rápidos.

Además, es computacionalmente más costosa que la interpolación bilineal, ya que requiere resolver un sistema de ecuaciones y estimar derivadas si no están disponibles. Esto también puede introducir errores adicionales en la interpolación.

En la figura 1 se muestra una comparación visual de la interpolación bilineal y bicúbica.

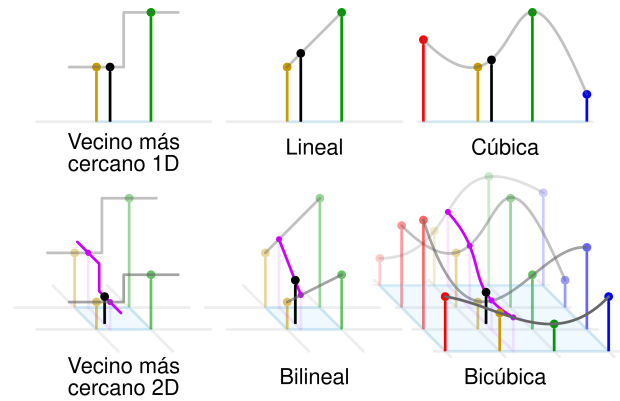


Fig. 1: Comparación de interpolación Bilineal vs Bicuúbica. Fuente: Wikimedia Commons.

2.7 Método de interpolación de Lagrange 2D

La interpolación de Lagrange en dos dimensiones es una extensión natural del método de interpolación de Lagrange unidimensional. Este método permite construir un polinomio bidimensional que pasa exactamente por un conjunto dado de puntos en un plano.

Dado un conjunto de puntos $(x_i, y_j, f(x_i, y_j))$ donde $i = 0, 1, \dots, m$ y $j = 0, 1, \dots, n$, el polinomio de interpolación de Lagrange en 2D se define como:

$$L(x, y) = \sum_{i=0}^m \sum_{j=0}^n f(x_i, y_j) L_i(x) L_j(y) \quad (14)$$

donde $L_i(x)$ y $L_j(y)$ son los polinomios base de Lagrange en una dimensión para las direcciones x y y respectivamente:

$$L_i(x) = \prod_{k=0, k \neq i}^m \frac{x - x_k}{x_i - x_k} \quad (15)$$

$$L_j(y) = \prod_{k=0, k \neq j}^n \frac{y - y_k}{y_j - y_k} \quad (16)$$

Estas bases de Lagrange bidimensionales tienen la propiedad de que $L_i(x_k)L_j(y_l) = 1$ si $i = k$ y $j = l$, y 0 en caso contrario. Esto garantiza que el polinomio interpolador pase exactamente por todos los puntos dados.

La interpolación de Lagrange 2D comparte muchas de las características de su contraparte unidimensional. Proporciona un único polinomio que pasa por todos los puntos dados y es exacta en los puntos de interpolación. El grado del polinomio resultante es (m, n) , es decir, de grado m en x y grado n en y . Al igual que en el caso unidimensional, puede presentar oscilaciones no deseadas (fenómeno de Runge) para un gran número de puntos, especialmente si estos están equiespaciados.

Sin embargo, la interpolación de Lagrange 2D también presenta algunos desafíos adicionales. El costo computacional aumenta significativamente con el número de puntos, siendo del orden $O((m+1)(n+1))$. La elección de los puntos de interpolación es crucial para minimizar las oscilaciones y obtener una buena aproximación en todo el dominio. Para funciones con variaciones rápidas o discontinuidades, puede ser necesario utilizar un gran número de puntos, lo que puede llevar a problemas de estabilidad numérica.

2.8 Método de interpolación de Splines bicúbicos

Los splines bicúbicos son una extensión bidimensional de los splines cúbicos, diseñados para interpolar funciones de dos variables. Este método combina la flexibilidad y suavidad de los splines cúbicos con la capacidad de manejar datos en un plano, lo que lo hace particularmente útil para interpolar superficies.

Consideremos una función $f(x, y)$ definida en una malla rectangular de puntos (x_i, y_j) , donde $i = 0, 1, \dots, m$ y $j = 0, 1, \dots, n$. El spline bicúbico $S(x, y)$ se construye de manera que sea continuo y tenga derivadas parciales continuas hasta el segundo orden en todo el dominio.

En cada rectángulo $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$, el spline bicúbico se define como un polinomio de grado 3 en ambas variables:

$$S(x, y) = \sum_{k=0}^3 \sum_{l=0}^3 a_{kl}(x - x_i)^k(y - y_j)^l \quad (17)$$

donde a_{kl} son los coeficientes a determinar. Para cada rectángulo, necesitamos 16 condiciones para determinar estos coeficientes. Estas condiciones incluyen:

1. Los valores de la función en los cuatro vértices del rectángulo.
2. Las derivadas parciales de primer orden $\frac{\partial f}{\partial x}$ y $\frac{\partial f}{\partial y}$ en los cuatro vértices.
3. Las derivadas parciales cruzadas $\frac{\partial^2 f}{\partial x \partial y}$ en los cuatro vértices.

La continuidad del spline y sus derivadas se asegura imponiendo condiciones de igualdad en las fronteras entre rectángulos adyacentes, lo que hace que los splines bicúbicos tengan una continuidad C^2 en todo el dominio, es decir, las primeras y segundas derivadas son continuas en todas las fronteras.

Además, los splines bicúbicos son **locales** en el sentido de que cambios en los datos en una región solo afectan a la interpolación en esa región y sus alrededores inmediatos. Esto contrasta con métodos de interpolación globales, como la interpolación de Lagrange en 2D, donde un cambio en un punto puede afectar a toda la superficie interpolada.

Sin embargo, la interpolación con splines bicúbicos también presenta algunos desafíos. La determinación de los coeficientes requiere la solución de un sistema de ecuaciones lineales, lo que puede ser computacionalmente costoso para mallas grandes. Además, si no se dispone de información sobre las derivadas en los puntos de la malla, estas deben ser estimadas, lo que puede introducir errores adicionales en la interpolación.

En comparación con la interpolación bicúbica directa, los splines bicúbicos tienden a ser más estables y menos propensos a producir oscilaciones no deseadas (overshooting), ya que minimizan la curvatura a lo largo del dominio. No obstante, están limitados a

mallas regulares, lo que los hace menos flexibles cuando se trabaja con datos dispersos o no estructurados.

2.9 Método de Newton-Raphson

El método de Newton-Raphson es un método numérico iterativo que genera una secuencia de puntos para aproximarse a la raíz de una función a partir de un valor inicial p_0 que se supone cercano a la raíz. Es decir, busca soluciones a la ecuación $f(x) = 0$. La sucesión se define de la siguiente manera:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (18)$$

La idea de este método es utilizar el punto donde la recta tangente interseca el eje x y tomarlo como el siguiente punto de la sucesión. Este método se deriva a partir de la serie de Taylor de la función y su eficacia radica en que, dado $|p - p_0| < \epsilon$, se cumple que $(p - p_0)^2 \approx 0$.

Suponiendo que $f'(x_n) \neq 0 \forall x_n$ y que p_0 se encuentra cerca de la raíz real, este método no solo garantiza su convergencia sino también que su orden de convergencia es cuadrático. La demostración de la convergencia cuadrática se puede encontrar en el apéndice ??.

Sin embargo, en este informe nos interesará otra aplicación del método de Newton-Raphson: su uso para resolver sistemas de ecuaciones no lineales.

Para el caso de funciones de varias variables, sea $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ una función vectorial definida como:

$$F(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{bmatrix} \quad (19)$$

donde $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ es el vector de variables.

La secuencia iterativa para el método de Newton-Raphson en \mathbb{R}^n se define como:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [J_F(\mathbf{x}_k)]^{-1} F(\mathbf{x}_k) \quad (20)$$

donde $J_F(\mathbf{x}_k)$ es la matriz Jacobiana de F evaluada en \mathbf{x}_k :

Sin embargo, calcular la inversa del Jacobiano en cada iteración puede ser computacionalmente costoso. Por lo tanto, podemos reformular el método para evitar invertir esta matriz.

$$J_F(\mathbf{x}_k) \Delta \mathbf{x}_k = -F(\mathbf{x}_k) \quad (21)$$

donde $\Delta \mathbf{x}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$. Una vez resuelto este sistema para $\Delta \mathbf{x}_k$, actualizamos nuestra aproximación:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x}_k \quad (22)$$

Esta generalización del método de Newton-Raphson para sistemas de ecuaciones no lineales mantiene las propiedades de convergencia cuadrática bajo condiciones similares a las del caso de una dimensión.

Criterios de elección de puntos iniciales

El método de Newton-Raphson es altamente sensible a la elección del punto inicial p_0 . Una elección adecuada puede llevar a una convergencia rápida, mientras que una elección inadecuada puede resultar en divergencia o convergencia a una solución incorrecta. Por lo tanto, proponemos una serie de criterios para determinar cómo elegir este valor inicial, con el objetivo de estudiar la convergencia del método. Estos criterios dependen en gran medida del experimento particular en el que se utiliza el método. Se recomienda al lector consultar el apartado (3.3) para obtener un contexto inicial del problema en el cual se basan los criterios propuestos.

Criterio del centroide de los sensores:

$$\mathbf{x}_0 = \left(\frac{x_1 + x_2 + x_3}{3}, \frac{y_1 + y_2 + y_3}{3}, \frac{z_1 + z_2 + z_3}{3} \right) \quad (23)$$

Este criterio se basa en elegir el punto que es equidistante a los sensores

Estimación basada en la medición más corta:

$$i_{min} = \arg \min_i d_i \quad (24)$$

$$\mathbf{x}_0 = \mathbf{s}_{i_{min}} - \frac{simin}{|simin|} d_{i_{min}} \quad (25)$$

Este método utiliza la posición del sensor con la medición más corta como punto de referencia: Donde d_i son las distancias medidas, si son las posiciones de los sensores, y i_{min} es el índice del sensor con la medición más corta.

Punto aleatorio dentro del volumen definido por los sensores:

$$\mathbf{x}_0 = \alpha \mathbf{s}_1 + \beta \mathbf{s}_2 + \gamma \mathbf{s}_3 \quad (26)$$

Este método genera un punto aleatorio dentro del tetraedro formado por los sensores y el origen, donde α, β, γ son números aleatorios que cumplen $\alpha + \beta + \gamma \leq 1$ y $\alpha, \beta, \gamma \geq 0$.

Promedio ponderado por las distancias inversas:

$$\mathbf{x}_0 = \frac{\sum_i i = 1^3 \frac{1}{d_i} \mathbf{s}_i}{\sum_i i = 1^3 \frac{1}{d_i}} \quad (27)$$

Este método da más peso a los sensores más cercanos

2.10 Fenómeno de Runge

El fenómeno de Runge es un problema que surge en la interpolación polinómica, particularmente cuando se utilizan polinomios de alto grado con puntos equiespaciados. Este fenómeno fue descubierto por Carl Runge en 1901 [2] y tiene importantes implicaciones para la precisión de las interpolaciones.

Específicamente, el fenómeno de Runge se refiere a la oscilación que ocurre en los extremos del intervalo de interpolación cuando se aumenta el grado del polinomio interpolador. Estas oscilaciones pueden llevar a errores significativos en la aproximación, especialmente cerca de los bordes del intervalo.

El fenómeno de Runge se debe a la naturaleza oscilatoria de los polinomios de alto grado. A medida que aumentamos el grado del polinomio interpolador, este gana la flexibilidad para oscilar más rápidamente entre los puntos de interpolación, lo que puede llevar a grandes desviaciones de la función original en las regiones entre estos puntos.

Este problema es especialmente pronunciado cuando se utilizan puntos de interpolación equiespaciados. La razón es que los puntos equiespaciados no proporcionan suficiente "control" sobre el comportamiento del polinomio cerca de los extremos del intervalo.

Existen varias estrategias para mitigar el fenómeno de Runge:

- Utilizar puntos de interpolación no equiespaciados, como los nodos de Chebyshev, que se concentran más cerca de los extremos del intervalo.
- Emplear métodos de interpolación alternativos, como los splines, que utilizan polinomios de grado más bajo por tramos.
- Limitar el grado del polinomio interpolador, aunque esto puede reducir la precisión de la interpolación en algunos casos.

El fenómeno de Runge subraya la importancia de elegir cuidadosamente tanto el método de interpolación como los puntos de interpolación, especialmente cuando se trabaja con funciones que tienen variaciones rápidas o cuando se requiere una alta precisión

en todo el intervalo de interpolación. Es un recordatorio de que aumentar el número de puntos o el grado del polinomio no siempre lleva a una mejor aproximación, y que las propiedades matemáticas subyacentes de los métodos de interpolación deben ser consideradas cuidadosamente en aplicaciones prácticas.

2.11 Nodos de Chebyshev

Los nodos de Chebyshev son una elección especial de puntos de interpolación que ayudan a mitigar el fenómeno de Runge y mejoran la precisión de la interpolación polinómica. Estos nodos se derivan de los polinomios de Chebyshev y tienen propiedades matemáticas que los hacen particularmente útiles en la interpolación.

Para un polinomio interpolador de grado n , los nodos de Chebyshev en el intervalo $[-1, 1]$ se definen como:

$$x_k = \cos \left(\frac{(2k+1)\pi}{2n+2} \right), \quad k = 0, 1, \dots, n \quad (28)$$

Estos puntos tienen varias características importantes. Su distribución no es uniforme; se concentran más cerca de los extremos del intervalo y menos en el centro, lo que ayuda a controlar las oscilaciones del polinomio interpolador en los extremos del intervalo. Además, los nodos de Chebyshev minimizan asintóticamente la norma máxima del error de interpolación polinómica. Es interesante notar que estos nodos están relacionados con los ceros de los polinomios de Chebyshev y pueden visualizarse como las proyecciones sobre el eje x de puntos equidistantes en un semicírculo unitario.

Para utilizar los nodos de Chebyshev en un intervalo $[a, b]$ distinto de $[-1, 1]$, se puede aplicar la siguiente transformación:

$$y_k = \frac{a+b}{2} + \frac{b-a}{2} x_k \quad (29)$$

La interpolación utilizando nodos de Chebyshev ofrece varias ventajas. Principalmente, reduce el fenómeno de Runge al concentrar más puntos cerca de los extremos del intervalo, limitando así la capacidad del polinomio para oscilar excesivamente en estas regiones. Además, el problema de interpolación con nodos de Chebyshev está generalmente mejor condicionado que con puntos equiespaciados. Para funciones suficientemente suaves, la interpolación con nodos de Chebyshev converge uniformemente a la función en todo el intervalo cuando se aumenta el número de nodos.

Sin embargo, es importante notar que los nodos de Chebyshev no son una solución universal. En algunos casos, como cuando se requieren valores de la función en puntos específicos o cuando se trabaja con datos experimentales, puede no ser posible elegir los puntos de interpolación libremente.

2.12 Métodos de evaluación del error

Para evaluar la precisión de los métodos de interpolación, utilizamos cuatro métricas de error diferentes. Estas métricas nos permiten comprender diversos aspectos del rendimiento de cada método de interpolación.

2.12.1 Error medio absoluto (MAE)

El error medio absoluto es la media aritmética de los valores absolutos de las diferencias entre las predicciones y los valores reales. Se calcula como:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (30)$$

donde y_i son los valores reales y \hat{y}_i son los valores interpolados. El MAE proporciona una medida de la magnitud promedio del error, sin considerar su dirección.

2.12.2 Error mediano absoluto (MED)

El error mediano absoluto es el valor mediano de los errores absolutos. Se calcula ordenando todos los errores absolutos y seleccionando el valor central. Esta métrica es menos sensible a valores atípicos que el error medio absoluto.

El error mediano es útil cuando se desea una medida de tendencia central del error que no se vea afectada por errores extremos ocasionales.

2.12.3 Error máximo absoluto (MAX)

El error máximo absoluto es el mayor error absoluto observado en todo el conjunto de datos.

Esta métrica es importante para identificar la peor estimación realizada por el método de interpolación, lo que puede ser crítico en aplicaciones donde se requiere una garantía de precisión mínima.

2.12.4 Error cuadrático medio (RMSE)

El error cuadrático medio es la raíz cuadrada del promedio de los errores al cuadrado. Se calcula como:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (31)$$

El RMSE penaliza más los errores grandes debido al término cuadrático, lo que lo hace útil cuando los errores grandes son particularmente indeseables.

3 Desarrollo Experimental

Los experimentos numéricos se llevaron a cabo en el lenguaje de programación **Python** utilizando las siguientes librerías: *numpy*, *scipy*, *pandas*, *matplotlib*, *math*, *seaborn*. Estas librerías se utilizaron tanto como producir los gráficos y tablas de este informe como para aprovecharse de las operaciones optimizadas con arrays que provee *numpy* o las implementaciones de *scipy* de los métodos de interpolación utilizados.

3.1 Análisis de métodos de interpolación

Interpolación de función $f_1 : \mathbb{R} \rightarrow \mathbb{R}$

Diseñamos un experimento sistemático que permitió evaluar y comparar el desempeño de tres métodos de interpolación diferentes: interpolación lineal, interpolación de Lagrange, splines cúbicos y Hermite, aplicados a la siguiente función en el intervalo $[-1, 1]$:

$$f_1(x) = -0.4 \tanh(50x) + 0.6 \quad (32)$$

El experimento inició con la generación de un conjunto denso de 1000 puntos equiespaciados en el intervalo $[-1, 1]$ para representar la función original con alta precisión. Este conjunto de puntos sirvió como referencia para evaluar la calidad de las interpolaciones posteriores.

A continuación, generamos series de i puntos equidistantes en el intervalo $[-1, 1]$, donde i varía de 2 a 50. Este rango lo elegimos para evaluar el comportamiento de los métodos de interpolación bajo diferentes densidades de nodos, permitiendo una comprensión detallada de cómo el error de interpolación evoluciona al aumentar el número de puntos. El límite superior de 50 lo consideramos suficiente para capturar las tendencias relevantes en la precisión de las interpolaciones, ya que más allá de este punto las mejoras tienden a ser marginales para la función estudiada. Para cada conjunto i de nodos, aplicamos los cuatro métodos de interpolación mencionados anteriormente, utilizando las siguientes funciones de la biblioteca *scipy.interpolate*:

1. *interp1d*: para la interpolación lineal.

2. *lagrange*: para la interpolación de Lagrange.

3. *CubicSpline*: para la interpolación mediante splines cúbicos.

4. *CubicHermiteSpline*: para la interpolación de Hermite.

Para evaluar la precisión de cada interpolación, comparamos los valores de la función original $f_1(x)$ con los de la función interpolada $I(x)$ en los 1000 puntos equiespaciados generados inicialmente, calculando la diferencia $|f_1(x_i) - I(x_i)|$ para cada punto x_i del conjunto denso. A partir de estas diferencias, computamos cuatro métricas de error: error máximo absoluto, error medio absoluto, error mediano absoluto y error cuadrático medio (RMSE). Este enfoque proporciona una medida robusta de la exactitud de cada método de interpolación, permitiéndonos cuantificar con precisión el error a lo largo de todo el intervalo en función del número de nodos utilizados.

Sin embargo, reconocemos que el uso de puntos equidistantes en la interpolación podía llevar al fenómeno de Runge, especialmente en los casos que la interpolación resultaba en un polinomio de grado elevado, como lo es la interpolación de Lagrange. Este fenómeno se caracteriza por oscilaciones no deseadas en los extremos del intervalo de interpolación, que pueden aumentar dramáticamente el error de la aproximación.

Para abordar esta limitación, decidimos repetir todo el proceso experimental utilizando nodos de Chebyshev en lugar de puntos equidistantes. Los nodos de Chebyshev son conocidos por su capacidad para mitigar el fenómeno de Runge, ya que se concentran más en los extremos del intervalo. Implementamos una función para generar estos nodos y repetimos todas las interpolaciones y cálculos de error utilizando esta nueva distribución de puntos.

Este enfoque dual nos permitió no solo comparar el rendimiento de los diferentes métodos de interpolación, sino también evaluar el impacto de la elección de los nodos en la precisión de las aproximaciones.

3.2 Interpolación de función $f_2 : \mathbb{R}^2 \rightarrow \mathbb{R}$

Extendimos el experimento anterior para evaluar y comparar el desempeño de cuatro métodos de interpolación en dos dimensiones: interpolación bilineal, interpolación bicúbica, interpolación de Lagrange 2D y splines bicúbicos. Estos métodos los aplicamos a la siguiente función en el intervalo $[-1, 1] \times [-1, 1]$:

$$\begin{aligned} f_2(x_1, x_2) = & 0.75 \exp\left(-\frac{(9x_1 - 2)^2}{4} - \frac{(9x_2 - 2)^2}{4}\right) \\ & + 0.75 \exp\left(-\frac{(9x_1 + 1)^2}{49} - \frac{(9x_2 + 1)^2}{10}\right) \\ & + 0.5 \exp\left(-\frac{(9x_1 - 7)^2}{4} - \frac{(9x_2 - 3)^2}{4}\right) \\ & - 0.2 \exp\left(-\frac{(9x_1 - 7)^2}{4} - \frac{(9x_2 - 3)^2}{4}\right) \end{aligned} \quad (33)$$

Dada la expresión matemática de la función f_2 , a la hora de evaluar puntos en esta, los resultados obtenidos son susceptibles a diversas fuentes de error numérico. Estos errores se verían amplificados dado que estamos exponenciando el resultado de estas evaluaciones, lo que podría aumentar el número de impresiones en nuestros experimentos numéricos. Por lo que en primer lugar decidimos reescribir la expresión de la función f_2 para minimizar el número de operaciones y por ende el error final. La expresión resultó en:

$$\begin{aligned} f_2(x) = & 0.75 \exp\left(-\frac{1}{4}((9x_1 - 2)^2 + (9x_2 - 2)^2)\right) \\ & + 0.75 \exp\left(-\left(\frac{(9x_1 + 1)^2}{49} + \frac{(9x_2 + 1)^2}{10}\right)\right) \\ & + 0.3 \exp\left(-\frac{1}{4}((9x_1 - 7)^2 + (9x_2 - 3)^2)\right) \end{aligned} \quad (34)$$

El procedimiento del experimento sigue una estructura similar al caso unidimensional. Iniciamos generando una malla densa de 100 x 100 puntos equiespaciados en el intervalo $[-1, 1] \times [-1, 1]$ para representar la función original con alta precisión. Esta malla sirvió como referencia para evaluar la calidad de las interpolaciones.

A continuación, generamos series de $i \times i$ puntos equidistantes en $[-1, 1] \times [-1, 1]$, donde i variaba de 5 a 50 en incrementos de 5. Estos puntos los utilizamos como nodos para las interpolaciones. Optamos por este intervalo ya que incrementos más finos resultarían computacionalmente costosos en 2D sin aportar información significativa adicional. Para cada conjunto de nodos, aplicamos los cuatro métodos de interpolación mencionados, utilizando las siguientes funciones de la biblioteca *scipy.interpolate*:

1. *griddata*: para la interpolación bilineal y bicúbica.
2. *RectBivariateSpline*: para la interpolación mediante splines bucúbicos.

En el caso de Lagrange 2D, SciPy no proporciona una función para este método, por lo que la implementamos nosotros.

En este caso, comparamos los valores de $f_2(x_1, x_2)$ con los de la función interpolada $I(x_1, x_2)$ en los 10,000 puntos de la malla densa inicial. Luego le computamos a las diferencias las mismas cuatro métricas utilizadas anteriormente.

Al igual que en el caso anterior, repetimos todos los experimentos utilizando nodos Chebyshev.

Este enfoque nos permitió comparar el rendimiento de los diferentes métodos de interpolación en dos dimensiones y evaluar cómo la densidad tanto como la elección de posición de los nodos afectaba la precisión de las aproximaciones.

3.3 Trilateración de posiciones

El proceso de trilateración es una forma de determinar la posición de un objeto conocida solamente su posición relativo a un conjunto de sensores y es una de las bases de como funcionan los sistemas de GPS. Aquí armaremos un sistema de trilateración en tres dimensiones asumiendo conocimiento perfecto de las mediciones. Asumiendo que conoce solamente la posición de los sensores y las mediciones realizadas, la posición de la partícula puede recuperarse resolviendo ecuaciones del tipo

$$\begin{cases} \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2} = d_1 \\ \sqrt{(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2} = d_2 \\ \sqrt{(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2} = d_3 \end{cases} \quad (35)$$

donde (x, y, z) es la posición de la partícula en el momento dado, (x_i, y_i, z_i) con $i = 1, 2, 3$ son las posiciones (conocidas) de los sensores y d_i son las mediciones realizadas. El objetivo de este experimento es resolver el sistema de ecuaciones utilizando el método de Newton-Raphson para recuperar la trayectoria de la partícula, interpolarla y comparar la trayectoria obtenida la del **Ground truth**.

Sin embargo, también nos va a interesar discernir si el error cometido a la hora de comprar las trayectorias proviene del método utilizado para resolver el sistema de ecuaciones o del método con el que interpolamos. A su vez evaluaremos qué tan bueno es el método de Newton-Raphson para hallar soluciones

Para este experimento numérico contabamos con 3 archivos que conformaban nuestro dataset:

1. *sensor_positions.txt*: Nos proveía las coordenadas en las cuales se ubicaban los 3 sensores. A partir de estas, obtuvimos mediciones de cuál era la posición de una partícula en un tiempo dado.
2. *measurements.txt*: Reportaba la distancia de la partícula hacia los 3 sensores cada 0.5 segundos (con un total de 21 mediciones).

3. *trajectory.txt*: Era nuestro **Ground Truth**, es decir, la posición real de la partícula en cada instante de tiempo.

Antes de experimentar numéricamente reescribimos el sistema de ecuaciones para facilitar las cuentas.

$$\begin{cases} (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 - d_1^2 = 0 \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 - d_2^2 = 0 \\ (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 - d_3^2 = 0 \end{cases} \quad (36)$$

Sea $F : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ definida como:

$$F(\mathbf{X}) = \begin{bmatrix} (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 - d_1^2 \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 - d_2^2 \\ (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 - d_3^2 \end{bmatrix} \quad (37)$$

y queremos hallar soluciones a

$$F(\mathbf{X}) = \vec{0} \quad (38)$$

El Jacobiano de F , denotado como J_F , es:

$$J_F(\mathbf{x}) = \begin{bmatrix} 2(x - x_1) & 2(y - y_1) & 2(z - z_1) \\ 2(x - x_2) & 2(y - y_2) & 2(z - z_2) \\ 2(x - x_3) & 2(y - y_3) & 2(z - z_3) \end{bmatrix} \quad (39)$$

Una vez definidas las funciones necesarias y preparados los datos, procedimos a evaluar diferentes estrategias para la elección del punto inicial p_0 en el método de Newton-Raphson. Implementamos los cuatro criterios de selección de puntos iniciales descritos en la sección (2.9). Para cada una de estas estrategias, aplicamos el método de Newton-Raphson para resolver el sistema de ecuaciones en cada una de las 21 mediciones proporcionadas. Esto nos permitió obtener un conjunto de estimaciones de la posición de la partícula para cada criterio de selección del punto inicial. Con el fin de evaluar la precisión de cada estrategia, calculamos el error total acumulado. Este error se definió como la suma de las distancias euclidianas entre las posiciones estimadas y las posiciones reales proporcionadas en el archivo de trayectoria (Ground Truth):

$$E_{total} = \sum_{i=1}^N |\mathbf{x}_i^{est} - \mathbf{x}_i^{real}|_2 \quad (40)$$

Dada la naturaleza de nuestro experimento, con mediciones en intervalos relativamente cortos de tiempo, podemos inferir que la partícula no se habrá desplazado significativamente entre mediciones consecutivas. Aprovechando esta característica, propusimos un criterio adicional para la selección de p_0 :

Sea S_i la solución al sistema de trilateración en el tiempo i . Entonces, para el tiempo $i + 1$, definimos: $p_0(i + 1) = S_i$. Es decir, tomamos la solución del sistema en el tiempo anterior como punto inicial para el sistema actual. Sin embargo, antes S_0 (la primera medición), debemos utilizar uno de los criterios de selección de p_0 mencionados en la sección (2.9). Este enfoque aprovecha la continuidad temporal de la trayectoria, potencialmente mejorando la convergencia del método de Newton-Raphson en las iteraciones subsiguientes. También comparamos el error de este criterio de selección de puntos bajo la métrica (40) utilizando como p_0 inicial, la mejor solución inicial de los demás criterios.

Una vez obtenidas las posiciones estimadas de la partícula utilizando el método de Newton-Raphson, procedimos a parametrizar la trayectoria para realizar la interpolación. Parametrizamos la trayectoria en función del tiempo.

Definimos la trayectoria como una función vectorial $\sigma(t)$ tal que:

$$\sigma(t) = (x(t), y(t), z(t)), \quad \sigma : \mathbb{R} \rightarrow \mathbb{R}^3 \quad (41)$$

donde $x(t)$, $y(t)$ y $z(t)$ representan las coordenadas de la partícula en función del tiempo.

Para interpolar esta trayectoria, utilizamos splines cúbicos. Aplicamos la interpolación por splines cúbicos a cada coordenada por separado:

$$\begin{aligned} x(t) &= S_x(t) \\ y(t) &= S_y(t) \\ z(t) &= S_z(t) \end{aligned} \quad (42)$$

donde $S_x(t)$, $S_y(t)$ y $S_z(t)$ son los splines cúbicos para cada coordenada.

Para evaluar la precisión de nuestra interpolación, comparamos la trayectoria interpolada con la trayectoria real proporcionada en el archivo *trajectory.txt*. Calculamos el error cuadrático medio (RMSE) entre la trayectoria interpolada y la real:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N ((\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2 + (\hat{z}_i - z_i)^2)} \quad (43)$$

donde $(\hat{x}_i, \hat{y}_i, \hat{z}_i)$ son las coordenadas interpoladas y (x_i, y_i, z_i) son las coordenadas reales en el instante i , y N es el número total de puntos en la trayectoria.

Este enfoque nos permite evaluar tanto la precisión del método de Newton-Raphson en la resolución del sistema de ecuaciones como la efectividad de la interpolación con splines cúbicos para reconstruir la trayectoria completa.

4 Resultados

4.1 Análisis de métodos de Interpolación

En esta sección, presentamos los resultados obtenidos al aplicar diversos métodos de interpolación a las funciones f_1 y f_2 . Nuestro análisis se centra en la precisión y el comportamiento de cada método bajo diferentes condiciones, utilizando tanto nodos equispaciados como nodos de Chebyshev.

Para cada función, examinamos la calidad de las aproximaciones mediante visualizaciones comparativas y un análisis cuantitativo del error. Prestamos especial atención a cómo el número de nodos y su distribución afectan el rendimiento de cada método, destacando fenómenos como la convergencia, la divergencia y el efecto Runge.

A continuación, presentamos los hallazgos para cada función, comenzando con la interpolación unidimensional de f_1 y luego pasando a la interpolación bidimensional de f_2 . Este análisis nos permitirá extraer conclusiones sobre la idoneidad de cada método para diferentes tipos de funciones y requisitos de interpolación.

Interpolación de $f_1 : \mathbb{R} \rightarrow \mathbb{R}$

4.1.1 Interpolación con nodos equispaciados

Al analizar los resultados de la interpolación de la función $f_1(x) = -0.4 \tanh(50x) + 0.6$ en el intervalo $[-1, 1]$ utilizando nodos equispaciados, observamos comportamientos distintivos para cada método de interpolación estudiado.

Para ilustrar visualmente cómo estos métodos de interpolación se ajustan a la función original, la Figura (2) muestra la función $f_1(x)$ junto con una interpolación mediante splines cúbicos utilizando 15 nodos.

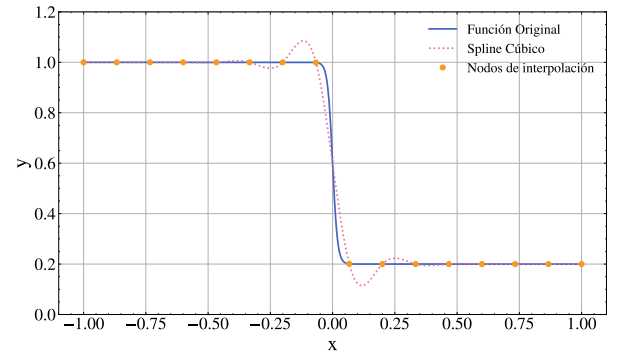


Fig. 2: Función original $f_1(x)$ e interpolación mediante splines cúbicos con 15 nodos equispaciados

Como se puede observar, la interpolación por splines cúbicos captura eficazmente la forma general de la función, con un ligero fallo en la transición abrupta cerca del origen. Los nodos de interpolación se distribuyen uniformemente en el intervalo, permitiendo una representación precisa tanto en las regiones de baja variación como en la zona de cambio rápido.

Pasando a analizar los errores, la Figura (3) muestra la evolución del Error Medio Absoluto (MAE) en función del número de nodos para los cuatro métodos de interpolación: lineal, Lagrange, spline cúbico y Hermite.

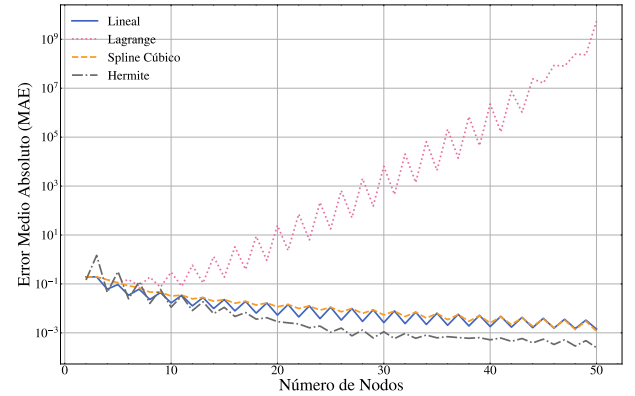


Fig. 3: Comparación de los cuatro métodos de interpolación en una variable utilizando el error medio absoluto

Como era de esperarse, es evidente que el método de interpolación de Lagrange presenta un comportamiento divergente a medida que aumenta el número de nodos. Esto se debe al previamente mencionado fenómeno de Runge (2.10). Debido al aumento en el grado del polinomio interpolador conduce a oscilaciones no deseadas, especialmente en los extremos del intervalo. Observamos que el error de Lagrange comienza a crecer exponencialmente a partir de aproximadamente 15 nodos, alcanzando valores muy elevados para 50 nodos.

Por otro lado, los métodos de interpolación lineal, spline cúbico y Hermite muestran una convergencia consistente a medida que aumenta el número de nodos. Entre estos, el método de Hermite demuestra ser el más preciso, manteniendo el error más bajo para cualquier número de nodos. Le siguen de cerca, casi a la par, los métodos de splines cúbicos y lineal, que muestran un comportamiento muy similar al de Hermite, especialmente para un número elevado de nodos.

Es interesante notar que todos los métodos, excepto Lagrange, muestran una disminución del error más pronunciada en los primeros nodos (hasta aproximadamente 10 nodos), tras lo cual la tasa de mejora se reduce. Esto sugiere que para esta función en particular,

aumentar el número de nodos más allá de cierto punto ofrece rendimientos decrecientes en términos de precisión.

También se observa una fluctuación periódica en el error para los cuatro métodos. Esta fluctuación parece estar relacionada con la paridad del número de nodos, siendo más evidente en la interpolación lineal y de Lagrange. Este fenómeno podría estar relacionado con la simetría de la función f_1 y la distribución de los nodos en el intervalo.

Es importante notar que para los errores máximo absoluto (MAX) y cuadrático medio (RMSE) se observa un comportamiento similar. Esto se puede ver en el Apéndice (6.1).

Al examinar el error mediano absoluto, se observa un comportamiento notablemente distinto al de las otras métricas de error, como se ilustra en la Figura (4).

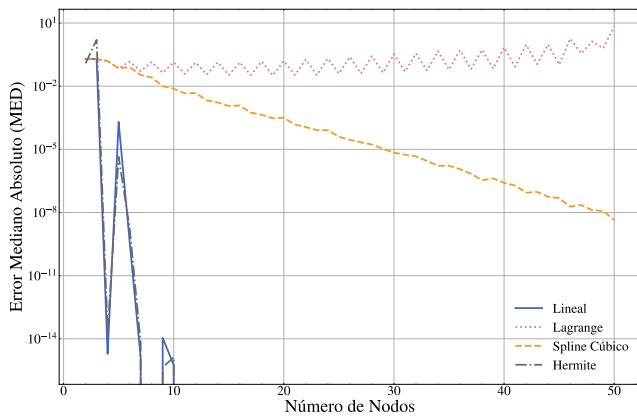


Fig. 4: Comparación de los cuatro métodos de interpolación en una variable utilizando el error mediano absoluto

Los métodos de Hermite y lineal exhiben una convergencia aparentemente rápida hacia cero, en contraste con los otros dos métodos. Esta discrepancia se atribuye a la morfología particular de la función f_1 en el intervalo estudiado. La función presenta regiones extensas de baja variación en los extremos, intercaladas con una transición abrupta en la región central. Consecuentemente, es probable que el error mediano, al seleccionar el valor central de la distribución de errores, coincida con una región donde las aproximaciones lineal y de Hermite son excepcionalmente precisas, aunque esto no sea representativo del rendimiento global en todo el intervalo.

Adicionalmente, el método de Lagrange muestra variaciones significativas en su comportamiento según la métrica de error empleada. Si bien mantiene su naturaleza divergente, la tasa de divergencia es considerablemente menor cuando se evalúa mediante el error mediano absoluto. Esta discrepancia se debe a que el error mediano es menos susceptible a las oscilaciones extremas características del fenómeno de Runge, que son más pronunciadas en los extremos del intervalo de interpolación.

4.1.2 Interpolación con nodos de Chebyshev

Para mitigar el fenómeno de Runge observado en la interpolación de Lagrange, se implementó una estrategia utilizando nodos de Chebyshev. La Figura (5) muestra la función $f_1(x)$ junto con una interpolación mediante Lagrange utilizando 20 nodos equispaciados.

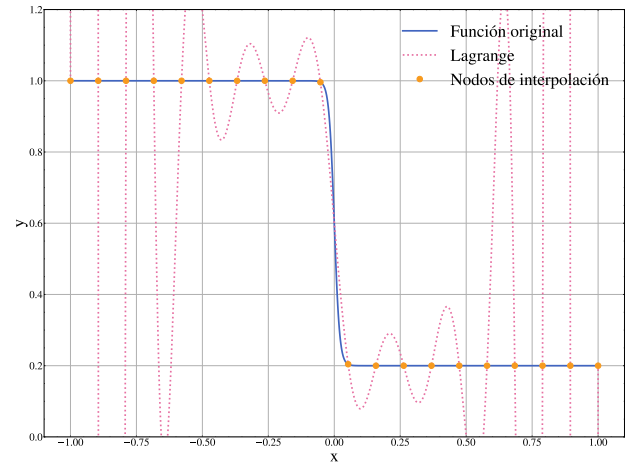


Fig. 5: Función original $f_1(x)$ e interpolación mediante Lagrange con 20 nodos equispaciados

Como se puede observar, al interpolarse con Lagrange utilizando nodos equispaciados el fenómeno de Runge se acentúa fuertemente en los extremos. En la Figura (6), se ve que sucede cuando interpolamos con el mismo método pero esta vez con nodos de Chebyshev.

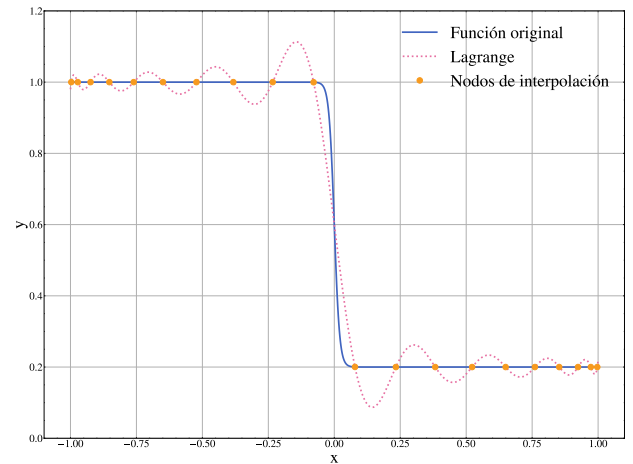


Fig. 6: Función original $f_1(x)$ e interpolación mediante Lagrange con 20 nodos con distribución Chebyshev

Como se puede apreciar, la utilización de nodos de Chebyshev reduce significativamente las oscilaciones en los extremos del intervalo, proporcionando una aproximación más precisa de la función original.

Para cuantificar esta mejora, la Figura (7) muestra la evolución del error medio absoluto (MAE) para la interpolación de Lagrange con nodos equispaciados y nodos de Chebyshev, variando el número de nodos de 2 a 50.

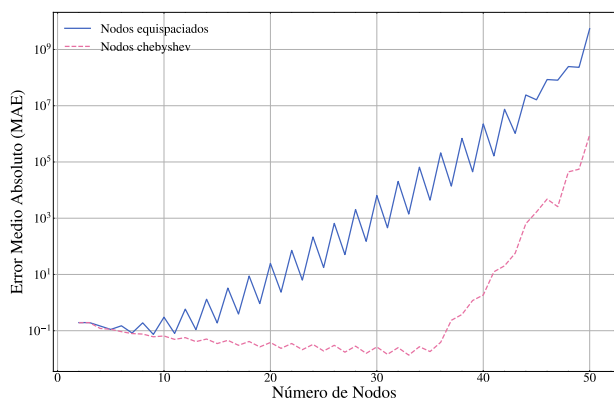


Fig. 7: Comparación del error medio absoluto para la interpolación de Lagrange con nodos equispaciados y nodos de Chebyshev

El gráfico demuestra claramente la superioridad de los nodos de Chebyshev para la interpolación de Lagrange. Mientras que el error con nodos equispaciados crece exponencialmente a partir de aproximadamente 15 nodos, el error con nodos de Chebyshev se mantiene acotado e incluso disminuye ligeramente al aumentar el número de nodos.

Es importante señalar que, si bien los nodos de Chebyshev mejoran significativamente el rendimiento de la interpolación de Lagrange, no tienen el mismo efecto positivo en otros métodos de interpolación. Por ejemplo, en el caso de los splines cúbicos, la utilización de nodos de Chebyshev resulta en un aumento del error, como se muestra en la Figura (8).

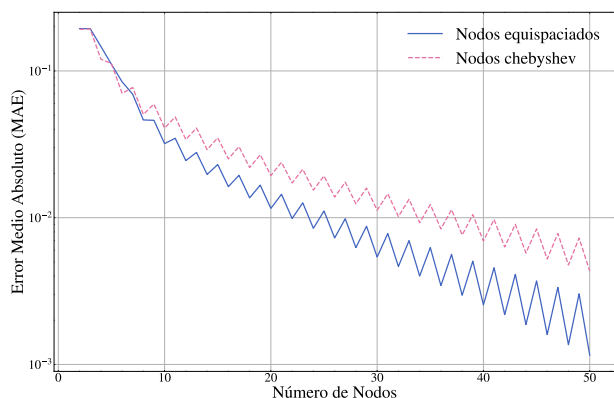


Fig. 8: Comparación del error medio absoluto para la interpolación con splines cúbicos utilizando nodos equispaciados y nodos de Chebyshev

Este comportamiento se debe a que los métodos como los splines cúbicos ya están diseñados para manejar eficientemente la interpolación en todo el intervalo, y la distribución no uniforme de los nodos de Chebyshev puede interferir con su rendimiento óptimo.

Para ver como influye en los demás métodos consultar el Apéndice (6.2).

Interpolación de $f_2 : \mathbb{R}^2 \rightarrow \mathbb{R}$

4.1.3 Interpolación con nodos equispaciados

En esta sección, analizaremos el comportamiento de diferentes métodos de interpolación bidimensional aplicados a la función $f_2(\mathbf{x})$ definida en el intervalo $[-1, 1] \times [-1, 1]$.

La Figura (9) muestra una representación tridimensional de la función original $f_2(\mathbf{x})$.

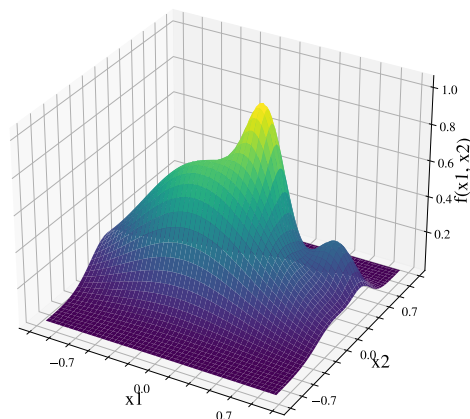


Fig. 9: Representación tridimensional de la función original $f_2(\mathbf{x})$

Para ilustrar el proceso de interpolación, la Figura (10) presenta la función $f_2(\mathbf{x})$ interpolada mediante splines bicúbicos utilizando una malla de 10x10 puntos equispaciados.

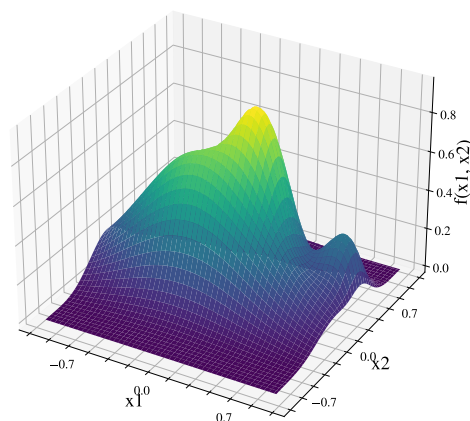


Fig. 10: Interpolación de $f_2(\mathbf{x})$ mediante splines bicúbicos con 10x10 puntos equispaciados

A continuación, analizaremos el rendimiento de cuatro métodos de interpolación bidimensional: bilineal, bicúbico, Lagrange 2D y splines bicúbicos. La Figura (11) muestra la evolución del Error Medio Absoluto (MAE) en función del número de nodos por dimensión para estos métodos.

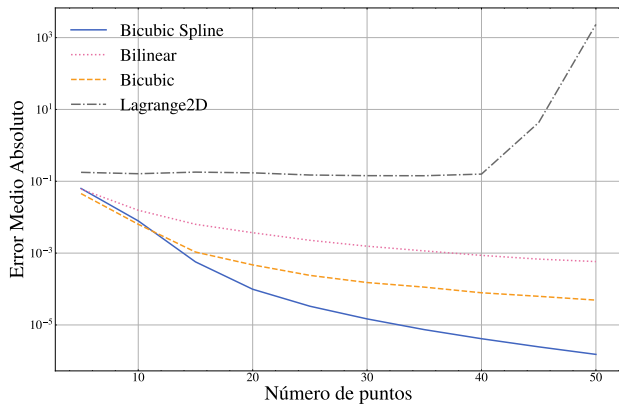


Fig. 11: Comparación del Error Medio Absoluto (MAE) para los cuatro métodos de interpolación bidimensional

Como se puede observar, el método de Lagrange 2D muestra un comportamiento divergente a medida que aumenta el número de nodos, similar al observado en el caso unidimensional. Este fenómeno se debe a la extensión del fenómeno de Runge a dos dimensiones.

Por otro lado, los métodos de interpolación bilineal, bicúbico y splines bicúbicos muestran una convergencia consistente al aumentar el número de nodos. Entre estos, el método de splines bicúbicos demuestra ser el más preciso, seguido de cerca por el método bicúbico. La interpolación bilineal, aunque también converge, mantiene un error ligeramente superior a los otros dos métodos.

Es notable que la tasa de convergencia de los métodos que convergen es más rápida inicialmente (hasta aproximadamente 10×10 nodos) y luego se desacelera. Esto sugiere que, para esta función en particular, aumentar el número de nodos más allá de cierto punto ofrece rendimientos decrecientes en términos de precisión.

Cabe mencionar que las otras métricas de error (error máximo absoluto, error mediano absoluto y RMSE) muestran tendencias similares a las observadas con el MAE. Estos resultados adicionales se pueden encontrar en el Apéndice ?? para una revisión más detallada.

4.1.4 Interpolación con Nodos de Chebyshev

Dado que la interpolación de Lagrange en dos dimensiones mostró un comportamiento divergente con nodos equiespaciados, similar al caso unidimensional, exploramos el uso de nodos de Chebyshev para mitigar este problema. Es importante señalar que, al igual que en el caso unidimensional, la aplicación de nodos de Chebyshev no mejora el rendimiento de los otros métodos de interpolación (bilineal, bicúbico y splines bicúbicos), por lo que nuestro análisis se centrará exclusivamente en el método de Lagrange 2D.

Esta figura muestra una comparación visual entre la interpolación de Lagrange 2D utilizando una malla de 20×20 nodos equiespaciados y la misma interpolación utilizando nodos de Chebyshev.

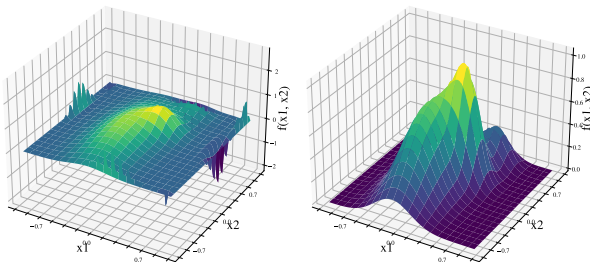


Fig. 12: Comparación de interpolación Lagrange 2D con nodos equiespaciados (izquierda) vs Chebyshev (derecha)

Como se puede observar, la interpolación de Lagrange 2D con nodos equiespaciados (izquierda) produce resultados extremadamente pobres, con oscilaciones violentas que no se asemejan en absoluto a la función original. En contraste, la interpolación utilizando nodos de Chebyshev (derecha) muestra una mejora significativa, capturando las características generales de la función $f_2(\mathbf{x})$, aunque aún presenta algunas imprecisiones en las regiones de cambio más abrupto.

Para cuantificar esta mejora, la Figura (13) muestra la evolución del error medio absoluto (MAE) para la interpolación de Lagrange 2D con nodos equiespaciados y nodos de Chebyshev, variando el número de nodos por dimensión de 5 a 50 con un intervalo de 5 puntos.

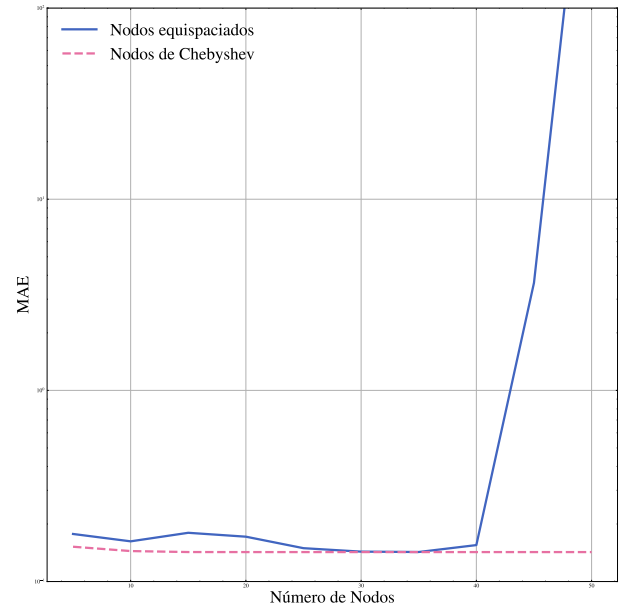


Fig. 13: Comparación del error medio absoluto para la interpolación de Lagrange 2D con nodos equiespaciados y nodos de Chebyshev

El gráfico demuestra claramente la superioridad de los nodos de Chebyshev para la interpolación de Lagrange 2D. Mientras que el error con nodos equiespaciados crece exponencialmente a medida que aumenta el número de nodos, el error con nodos de Chebyshev se mantiene relativamente constante e incluso muestra una ligera tendencia decreciente.

Esta estabilidad en el error con nodos de Chebyshev es particularmente notable, ya que sugiere que se puede aumentar la resolución de la interpolación sin incurrir en el fenómeno de Runge que plaga la versión con nodos equiespaciados. Sin embargo, es importante señalar que, a pesar de esta mejora, la interpolación de Lagrange 2D sigue siendo menos precisa que los métodos de splines bicúbicos o interpolación bicúbica para esta función en particular.

4.2 trilateración

En esta sección analizaremos los resultados obtenidos del experimento numérico de la trilateración de posiciones. En este experimento nos interesó dicotomizar el error que cometíamos al recuperar la trayectoria, es decir si el error que cometíamos comprando contra el **Ground Truth** se debía a que el método de Newton-Raphson encontraba soluciones con mucho error al sistema de trilateración o bien que la interpolación no correspondía con el recorrido real de la partícula.

El primer resultado que obtuvimos fue la trayectoria en estimada de la partícula luego de resolver el sistema de ecuaciones en (37)

Trayectoria estimada del vehículo (Newton-Raphson)

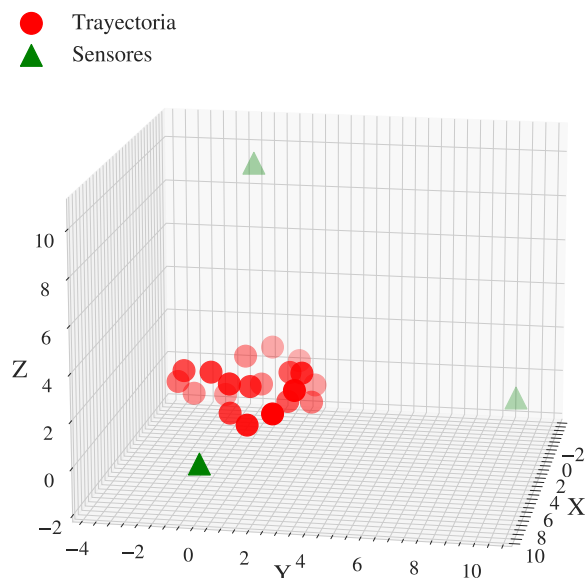


Fig. 14: Trayectoria recuperada utilizando Newton-Raphson

4.2.1 Evaluación del error de Newton-Raphson

A la hora de aplicar el método de Newton-Raphson para resolver el sistema de trilateración nos encontramos con variaciones en el error cometido según el método de selección de punto inicial empleado

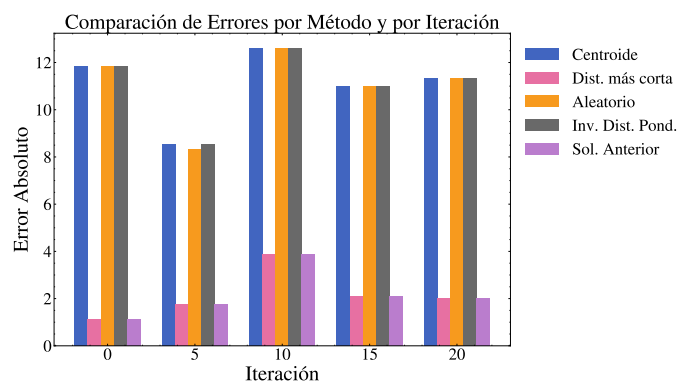


Fig. 15: Error absoluto según método de selección de punto inicial

Como se puede observar, los métodos de selección de punto inicial se dividen claramente en dos grupos:

Métodos de selección que convergen a una solución subóptima:

El método del centroide, el método aleatorio y el método de distancia inversa ponderada todos convergen consistentemente a la misma solución, a pesar de partir de puntos iniciales diferentes. Esta solución, sin embargo, tiene un error significativamente mayor que la otra solución encontrada.

Métodos de selección que convergen a una solución óptima:

El método de la distancia más corta y el método de la solución anterior convergen a una solución diferente, con un error mucho menor.

Al hallar estos resultados, nuestra primer hipótesis fue que no era posible que distintos criterios de selección de punto inicial, tuvieran

un mismo error absoluto. Sin embargo, esto si puede ser posible debido a la presencia de múltiples mínimos locales en la función objetivo (trilateración).

Table 1 Convergencia de Métodos de Inicialización en Trilateración

Método	Tiempo 0	Tiempo 10	Tiempo 20
Centroide	PI: [3.33, 3.33, 3.33] PC: [7.33, 6.33, 7.33]	PI: [3.33, 3.33, 3.33] PC: [8.67, 5.67, 8.67]	PI: [3.33, 3.33, 3.33] PC: [7.33, 6.33, 7.33]
Dist. más corta	PI: [-0.05, 0.00, 0.00] PC: [-0.00, -1.00, -0.00]	PI: [-0.44, 0.00, 0.00] PC: [-0.00, -3.00, -0.00]	PI: [0.00, 0.00, -0.05] PC: [-0.00, -1.00, 0.00]
Aleatorio	PI: [3.74, 2.81, 3.45] PC: [7.33, 6.33, 7.33]	PI: [3.18, 6.48, 0.34] PC: [8.67, 5.67, 8.67]	PI: [1.36, 0.17, 8.47] PC: [7.33, 6.33, 7.33]
Inv. Dist. Pond.	PI: [3.43, 3.14, 3.43] PC: [7.33, 6.33, 7.33]	PI: [3.57, 2.87, 3.57] PC: [8.67, 5.67, 8.67]	PI: [3.43, 3.14, 3.43] PC: [7.33, 6.33, 7.33]
Sol. Anterior	PI: [0.00, 0.00, 0.00] PC: [-0.00, -1.00, -0.00]	PI: [-0.87, -2.57, -0.81] PC: [-0.00, -3.00, -0.00]	PI: [-1.48, -0.67, 0.81] PC: [-0.00, -1.00, -0.00]

Table 1 Nota: PI = Punto Inicial, PC = Punto de Convergencia.

En la siguiente tabla podemos ver que a pesar de que todos los **Puntos Iniciales** que generan los criterios son distintos, el método de Newton-Raphson termina convergiendo a los mismos **Puntos de Convergencia** para los 2 grupos de métodos de selección. Esto se repite para cada sistema de ecuaciones que se resuelve utilizando este método numérico, por lo que el Error Absoluto termina siendo el mismo entre todos los miembros de un mismo grupo de criterios que convergen a la misma solución.

4.2.2 Interpolación y comparación con la trayectoria real

Una vez obtenida la trayectoria estimada mediante el método de Newton-Raphson, procedimos a interpolar estos puntos utilizando splines cúbicos. La Figura 16 muestra una comparación entre la trayectoria real y la trayectoria interpolada.

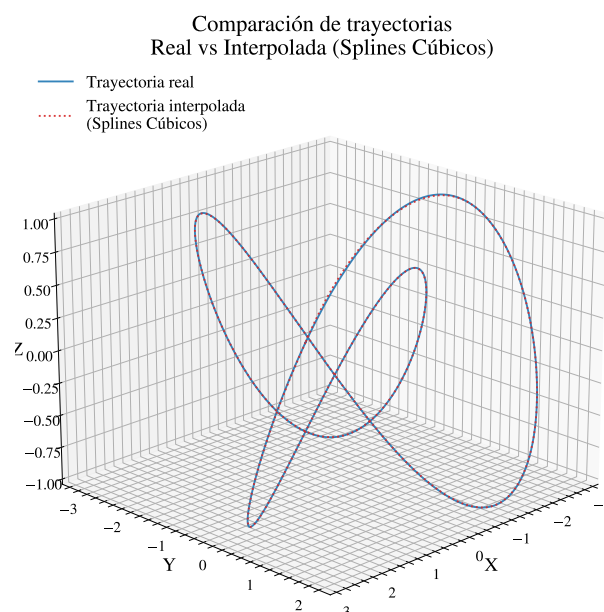


Fig. 16: Comparación entre la trayectoria real y la trayectoria interpolada con splines cúbicos

Como se puede observar, la trayectoria interpolada (en naranja) sigue de cerca la trayectoria real (en azul). Sin embargo, se aprecian algunas desviaciones, especialmente en los puntos de inflexión de la trayectoria. Estas diferencias pueden atribuirse tanto a errores en la estimación inicial de las posiciones como a las limitaciones inherentes de la interpolación con splines cúbicos.

Para cuantificar la precisión de nuestra interpolación, calculamos el Error Cuadrático Medio (RMSE) entre la trayectoria interpolada y

la trayectoria real. La Figura (17) muestra la evolución del RMSE a lo largo del tiempo.

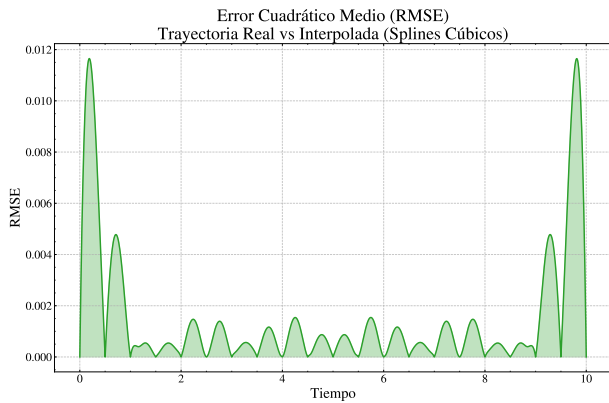


Fig. 17: Error Cuadrático Medio (RMSE) entre la trayectoria real y la interpolada

Estos resultados indican que nuestro método de estimación e interpolación produce errores relativamente pequeños en promedio. Sin embargo, la presencia de picos en el RMSE sugiere que hay momentos específicos donde la precisión disminuye significativamente.

4.2.3 Análisis de las fuentes de error

El error en nuestra estimación de la trayectoria puede atribuirse a dos fuentes principales:

1. **Error en la estimación inicial:** Como se observó en la sección (4.2), el método de Newton-Raphson converge a diferentes soluciones dependiendo del punto inicial elegido. Aunque seleccionamos el método que produjo el menor error, es posible que algunas estimaciones iniciales sean menos precisas que otras.
2. **Error de interpolación:** La interpolación con splines cúbicos, aunque generalmente precisa, puede introducir errores adicionales, especialmente en puntos donde la trayectoria cambia rápidamente de dirección.

La combinación de estos factores explica las desviaciones observadas entre la trayectoria estimada y la real. El aumento general del RMSE con el tiempo sugiere que los errores tienden a acumularse, posiblemente debido a la naturaleza secuencial de nuestro método de estimación.

5 Conclusiones

Nuestro estudio comparativo de métodos de interpolación tanto para funciones unidimensionales como bidimensionales ha revelado importantes insights sobre su rendimiento y aplicabilidad. En el caso unidimensional, los métodos de splines cúbicos y Hermite demostraron ser los más precisos y estables, manteniendo un error bajo incluso al aumentar el número de nodos. La interpolación de Lagrange, aunque inicialmente problemática debido al fenómeno de Runge, mostró una mejora significativa con el uso de nodos de Chebyshev, subrayando la importancia de la elección de los puntos de interpolación. Para funciones bidimensionales, los splines bicúbicos y la interpolación bicúbica ofrecieron los mejores resultados, con una convergencia consistente al aumentar la densidad de nodos. Notablemente, la extensión del método de Lagrange a dos dimensiones también se benefició sustancialmente de los nodos de Chebyshev, aunque sin superar el rendimiento de los métodos más avanzados. Estos hallazgos subrayan la importancia de seleccionar el método de interpolación adecuado según las características de la función y los requisitos de precisión, así como la consideración cuidadosa de la distribución de nodos para optimizar el rendimiento.

Nuestros resultados proporcionan una guía valiosa para la elección de métodos de interpolación en diversas aplicaciones prácticas, desde el análisis de datos hasta la modelización numérica.

En cuanto al experimento de trilateración, nuestros hallazgos revelan aspectos cruciales sobre la aplicación de métodos numéricos en problemas de localización:

1. **Sensibilidad al punto inicial:** El método de Newton-Raphson mostró una clara dependencia del punto inicial elegido. Los métodos de selección de punto inicial se dividieron en dos grupos, convergiendo a soluciones óptimas y subóptimas, lo que subraya la importancia de una cuidadosa elección del punto de partida en problemas de optimización no lineal.
2. **Múltiples mínimos locales:** La presencia de múltiples mínimos locales en la función objetivo de trilateración destaca la complejidad inherente a este tipo de problemas y la necesidad de estrategias robustas para evitar converger a soluciones subóptimas.
3. **Eficacia de la interpolación con splines cúbicos:** La interpolación de la trayectoria estimada mediante splines cúbicos demostró ser efectiva, logrando una buena aproximación a la trayectoria real. Sin embargo, las desviaciones observadas, especialmente en puntos de inflexión, sugieren limitaciones en la capacidad de capturar cambios abruptos en la trayectoria.
4. **Acumulación de errores:** El aumento del RMSE a lo largo del tiempo indica una acumulación de errores, posiblemente debido a la naturaleza secuencial del método de estimación. Esto sugiere la necesidad de estrategias de corrección o filtrado para aplicaciones en tiempo real o de larga duración.

Estos hallazgos subrayan la importancia de seleccionar el método de interpolación adecuado según las características de la función y los requisitos de precisión, así como la consideración cuidadosa de la distribución de nodos para optimizar el rendimiento. En el contexto de la trilateración, nuestros resultados destacan la necesidad de combinar métodos robustos de optimización con técnicas de interpolación avanzadas para lograr estimaciones precisas de trayectorias.

6 Apéndice

6.1 Comparativas de error para f_1 con nodos equispaciados

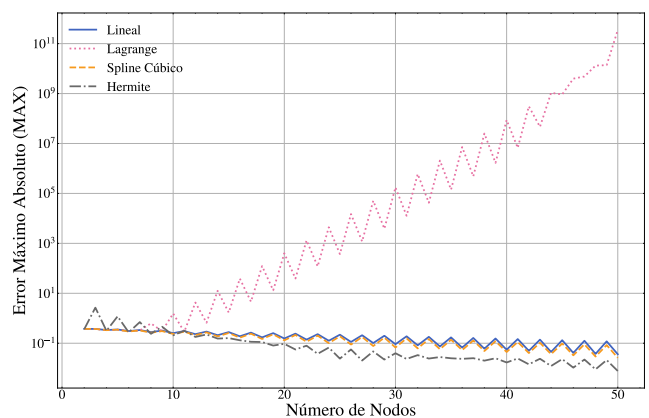


Fig. 18: Comparación de los cuatro métodos de interpolación en una variable utilizando el error máximo absoluto (MAX)

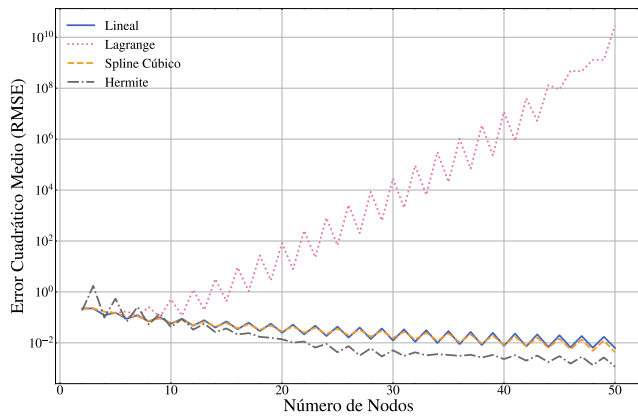


Fig. 19: Comparación de los cuatro métodos de interpolación en una variable utilizando el error cuadrático medio (RMSE)

6.2 Comparativas de error para f_1 con nodos de Chebyshev

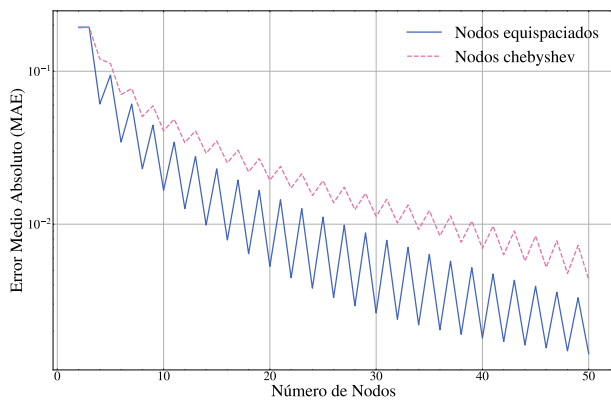


Fig. 20: Comparación del error medio absoluto para la interpolación lineal utilizando nodos equispaciados y nodos de Chebyshev

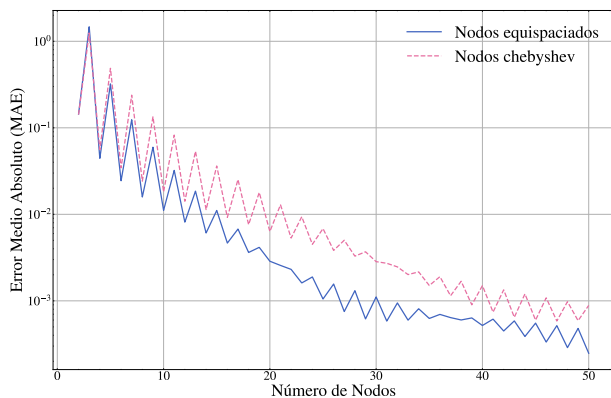


Fig. 21: Comparación del error medio absoluto para la interpolación con hermite utilizando nodos equiespaciados y nodos de Chebyshev

6.3 Comparativas de error para f_2

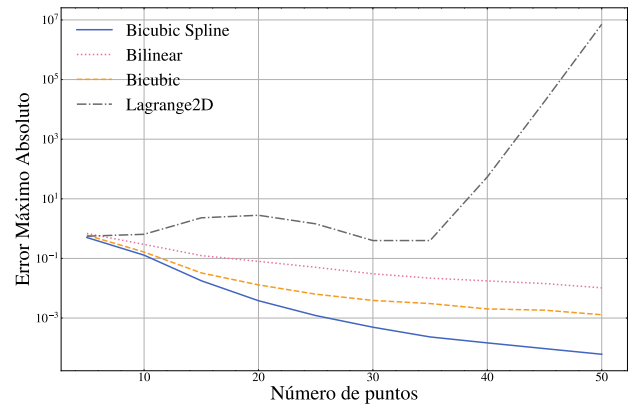


Fig. 22: Comparación del Error Máximo Absoluto (MAX) para los cuatro métodos de interpolación bidimensional

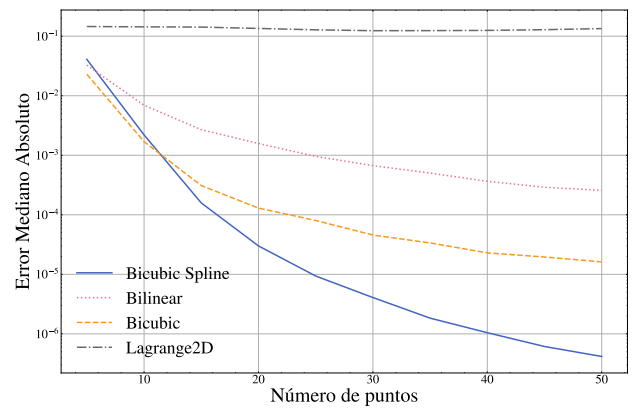


Fig. 23: Comparación del Error Mediano Absoluto (MED) para los cuatro métodos de interpolación bidimensional

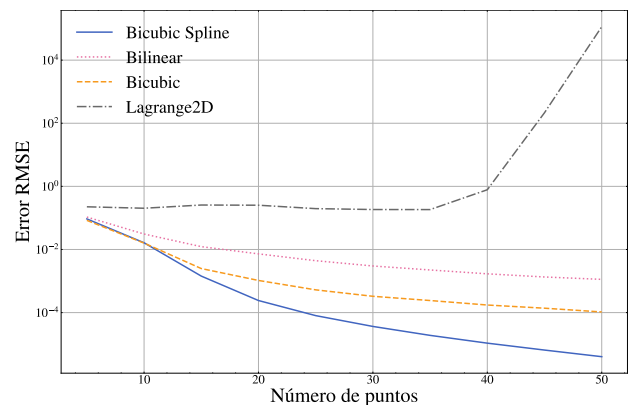


Fig. 24: Comparación del Error Cuadrático Medio (RMSE) para los cuatro métodos de interpolación bidimensional

6.4 Convergencia cuadrática de Newton-Raphson

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (44)$$

Definimos el error en la n -ésima iteración como $e_n = x_n - r$. Nuestro objetivo es demostrar que $|e_{n+1}| \leq K|e_n|^2$ para alguna constante $K > 0$.

1. Expandimos $f(x_n)$ alrededor de r usando el teorema de Taylor de segundo orden:

$$f(x_n) = f(r) + f'(r)(x_n - r) + \frac{1}{2}f''(\xi_n)(x_n - r)^2 \quad (45)$$

donde ξ_n está entre x_n y r .

2. Como r es una raíz, $f(r) = 0$. Sustituyendo esto y la definición de e_n :

$$f(x_n) = f'(r)e_n + \frac{1}{2}f''(\xi_n)e_n^2 \quad (46)$$

3. Similarmente, expandimos $f'(x_n)$ alrededor de r :

$$f'(x_n) = f'(r) + f''(\eta_n)(x_n - r) = f'(r) + f''(\eta_n)e_n \quad (47)$$

donde η_n está entre x_n y r .

4. Ahora, sustituyamos estas expresiones en la fórmula del método de Newton-Raphson:

$$e_{n+1} = x_{n+1} - r = x_n - r - \frac{f(x_n)}{f'(x_n)} \quad (48)$$

$$= e_n - \frac{f'(r)e_n + \frac{1}{2}f''(\xi_n)e_n^2}{f'(r) + f''(\eta_n)e_n} \quad (49)$$

5. Simplificando:

$$e_{n+1} = e_n \left[1 - \frac{f'(r)}{f'(r) + f''(\eta_n)e_n} \right] - \frac{\frac{1}{2}f''(\xi_n)e_n^2}{f'(r) + f''(\eta_n)e_n} \quad (50)$$

$$= \frac{f''(\eta_n)e_n^2 - \frac{1}{2}f''(\xi_n)e_n^2}{f'(r) + f''(\eta_n)e_n} \quad (51)$$

6. Tomando el valor absoluto:

$$|e_{n+1}| = \frac{|[f''(\eta_n) - \frac{1}{2}f''(\xi_n)]e_n^2|}{|f'(r) + f''(\eta_n)e_n|} \quad (52)$$

7. Como f' y f'' son continuas, existen constantes M_1 y M_2 tales que $|f''(x)| \leq M_2$ para x cerca de r , y $|f'(r)| \geq M_1 > 0$. Entonces:

$$|e_{n+1}| \leq \frac{\frac{3}{2}M_2|e_n|^2}{M_1 - M_2|e_n|} \quad (53)$$

8. Para n suficientemente grande, $|e_n|$ será pequeño, por lo que $M_1 - M_2|e_n| \geq \frac{M_1}{2}$. Entonces:

$$|e_{n+1}| \leq K|e_n|^2 \quad (54)$$

donde $K = \frac{3M_2}{M_1}$.

7 Referencias

- 1 Richard L Burden and J Douglas Faires. *Análisis numérico*. Cengage Learning Editores, 2017.
- 2 Carl Runge. Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten. *Zeitschrift für Mathematik und Physik*, 46:224–243, 1901.