

Proyecto de investigación: Métodos de Funciones Radiales para la Solución de EDP

<http://www.dci.dgsca.unam.mx/pderbf/>

Constrcción del interpolante mediante funciones de base radial usando kernel de capa delgada ¹

Objetivo: Ilustrar mediante un ejemplo simple en Matlab la forma de calcular el interpolante radial de la funcion bivariada :

$$u(x, y) = \frac{\frac{5}{4} + \cos(5.4y)}{6 + 6(3x-1)^2}$$

sobre un conjunto de N puntos aleatorios $\{(x_i, y_i)\}_{i=1}^N \subset \mathbb{R}^2$

El código correspondiente se encuentra en la pagina del proyecto, sección Docencia.

Definición del Problema

Se desea encontrar el interpolante :

$$Iu(x, y) = \sum_{j=1}^N \lambda_j \phi_j(r) + p^2(x, y)$$

en donde :

$$\begin{aligned}\phi(r) &= r^4 \log(r) \\ r &= \sqrt{(x - x_j)^2 + (y - y_j)^2} \\ p^2(x, y) &= a_1 x^2 + a_2 y^2 + a_3 xy + a_4 x + a_5 y + a_6\end{aligned}$$

Aquí $\phi(r)$ es un tipo de Kernel reproductor al que llamaremos spline de capa delgada en \mathbb{R}^2 y $p^2(x, y)$ es un polinomio de grado 2 en \mathbb{R}^2

Lo que implica el siguiente sistema:

$$\begin{bmatrix} M & P \\ P^t & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \\ a_1 \\ a_2 \\ \vdots \\ a_6 \end{bmatrix} = \begin{bmatrix} u(x_1, y_1) \\ u(x_2, y_2) \\ \vdots \\ u(x_N, y_N) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

¹Material elaborado en el curso: Métodos numéricos para la solución de EDP mediante Funciones de base Radial, Posgrado en Ciencias Matemáticas, UNAM, 2007, impartido por el Dr. Pedro González Casanova. Alumnos participantes: Diego A. Ayala Rodriguez y Francisco J. Martinez.

Algoritmo: Colocación Asimétrica

1. Supongamos que tenemos N números aleatorios $\{(x_i, y_i)\}_{i=0}^N$.
2. Construimos la matriz lineal algebraica de Gramm de $N + 6 \times N + 6$.

$$G = \begin{pmatrix} M & P \\ P^t & 0_M \end{pmatrix}$$

con

$$M = \begin{pmatrix} \phi_1(r_1) & \phi_2(r_1) & \cdots & \phi_N(r_1) \\ \phi_1(r_2) & \phi_2(r_2) & \cdots & \phi_N(r_2) \\ \vdots & & & \vdots \\ \phi_1(r_N) & \phi_2(r_N) & \cdots & \phi_N(r_N) \end{pmatrix}$$

$$P = \begin{pmatrix} x_1^2 & y_1^2 & x_1 y_1 & x_1 & y_1 & 1 \\ x_2^2 & y_2^2 & x_2 y_2 & x_2 & y_2 & 1 \\ \vdots & & & \vdots & & 1 \\ x_N^2 & y_N^2 & x_N y_N & x_N & y_N & 1 \end{pmatrix}$$

$$P^t = \begin{pmatrix} x_1^2 & x_2^2 & \cdots & x_N^2 \\ y_1^2 & y_2^2 & \cdots & y_N^2 \\ x_1 y_1 & x_2 y_2 & \cdots & x_N y_N \\ x_1 & x_2 & \cdots & x_N \\ y_1 & y_2 & \cdots & y_N \\ 1 & 1 & \cdots & 1 \end{pmatrix}$$

$$0_M = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

aquí $r_i = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

3. Para resolver el sistema, utilizamos el operador \backslash . Es decir; para encontrar el vector de las incógnitas se aplica la inversa de la matriz .

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \\ a_1 \\ a_2 \\ \vdots \\ a_6 \end{bmatrix} = \begin{bmatrix} M & P \\ P^t & 0 \end{bmatrix}^{-1} \begin{bmatrix} u(x_1, y_1) \\ u(x_2, y_2) \\ \vdots \\ u(x_N, y_N) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

4. Grafica la solución e imprime la condición del la matriz G.

Instrucciones para correr el ejemplo, tanto en matlab como en C++

Matlab:

1 Bajar la carpeta comprimida ubicada en sección docencia - Codigos en Matlab - programas para la interpolación de superficies - Rutina principal y auxiliares(comprimido)

2 Descomprimir la carpeta , una vez hecho esto abrir Matlab, abrir cada archivo de la carpeta y guardar todos los archivos en la carpeta work de Matlab, por ejemplo, si abrimos el archivo gna.m (ya en Matlab) elegimos la opción guardar como - Disco local C (esto depende de donde se haya instalado Matlab, puede ser otra unidad) - MATLAB -work

3 Ubicarse en la ventana de comandos de Matlab, para correr el programa solo hay que teclear:

```
mirbf_thinplate (100 )
```

Una vez que termina la ejecución del programa se desplegara la figura 1.

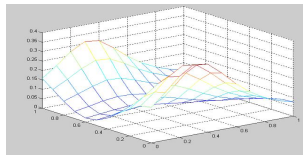


figura 1. Salida del programa mirf_thinplate con 100 puntos.

C++ Windows

Para compilar el programa en windows, usamos Dev C++. (se puede obtener la versión gratuita de Dev C++ en la siguiente pagina: <http://www.uptodown.com/buscar/bloodshed-dev-c++/>) Para visualizar los resultados de forma gráfica usamos, gnuplot.

1 Una vez instalados los programas, se debe bajar la carpeta comprimida, ubicada en la sección de docencia - Codigos en C++ - programas para la interpolación de superficies - Rutina principal y auxiliares(comprimido)

2 Ya que se guardo la carpeta es necesario descomprimirla. Despues abrir la carpeta, al hacer esto se tiene acceso a los archivos contenidos en la misma, entre ellos se encuentra un archivo llamado *Pry Int* que tiene un logotipo como el que se muestra en la figura 2:



figura 2

3 Seleccionar el archivo *Pry Int* y abrirlo.

Al abrir *Pry Int* se abrirá Dev C++ mostrando en el lado izquierdo de la pantalla una serie de archivos, de los cuales se debe elegir y abrir *main.cpp* (haciendo doble clic con el ratón).

Para poder ejecutar el programa primero se debe compilar, seleccionando el menú ejecutar - compilar.

El programa requiere de argumentos para poder funcionar correctamente, que en este caso son: el número de puntos para la interpolación, el nombre del archivo donde se podrán ver los resultados y el tamaño de la malla.

Para pasar estos argumentos al programa, seleccione en el menú ejecutar - Argumentos del programa, en el proyecto *Pry Int* ya están los argumentos establecidos, como se muestra en la figura 3.

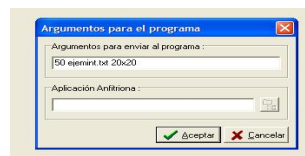


figura 3

En este proyecto se trabaja con 50 puntos, el nombre del archivo es *ejemint.txt* y el tamaño de la malla es de 20x20.

Si se desea, se pueden cambiar los argumentos del programa desde dicha ventana.

5 Para ejecutar el programa, en el menú ejecutar seleccionamos la opción ejecutar .

6 El archivo de salida, *ejemint.txt* de la ejecución se encuentra en la carpeta que se abrió al principio (*InterTPSCpp*)

Nota : si se desea visualizar los resultados de forma gráfica, abrir la carpeta de *gnuplot* y ubicar el icono :



figura 4

hacer doble clic, para abrir la ventana de comandos.

Para ubicar la carpeta donde está el archivo *ejemint.txt*, seleccionar el menú - *ChDir*, y buscar la carpeta donde está el archivo *ejemint.txt*.

Para graficar, se debe teclear en la ventana de comandos la siguiente instrucción:

```
plot'ejemint.txt'
```

Aparecera entonces la superficie generada:

