UNIVERSIDADE FEDERAL DE SANTA CATARINA

CURSO DE ENGENHARIA DE COMPUTAÇÃO

CÁLCULO NUMÉRICO

APROXIMAÇÃO BILINEAR UTILIZANDO MÉTODO DOS MÍNIMOS QUADRADOS

GABRIEL ESTEVAM

RAUL ROSÁ

VINÍCIUS ZANON

ARARANGUÁ – SC

SUMÁRIO

INTRODUÇÃO

Nas mais diversas áreas da ciência ocorrem, frequentemente, situações em que sabemos o valor de uma função avaliada em um conjunto de pontos, mas não temos uma expressão analítica que nos permite calcular seu valor em um ponto arbitrário. Como por exemplo, funções advindas de resultados experimentais no campo da física, química e outros.

Neste relatório é apresentada uma perspectiva de aproximação de funções. Dado apenas os valores da função em certos pontos (domínio discreto), não exigimos que a 'função aproximadora' interpole a função dada nos pontos (interpolação polinomial), exigimos apenas que essa função tome valores de forma a minimizar a distância aos valores dados, ou seja, foi-se utilizado o método dos mínimos quadrados para determinar a função aproximadora.

No decorrer do relatório algumas habilidades foram essenciais para que o trabalho fosse realizado com êxito. Algumas como organização, trabalho em grupo, reconhecimento de dados, auxílio do *software Octave* para implementação de métodos numéricos e apresentação dos resultados via gráficos, raciocínio na aplicação de fórmulas, análise/construção de gráficos e aplicação teórica. Realizado em 23 de Novembro de 2016, no Campus Araranguá – UFSC.

RESUMO

Neste relatório realizamos um estudo de Aproximações de Funções envolvendo Métodos dos Mínimos Quadrados (MMQ) em duas dimensões. A partir da dedução de fórmulas do MMQ em uma dimensão, pôde-se por indução e construção algébrica formalizar nosso problema em duas dimensões, achando dessa forma um polinômio bilinear que aproximasse uma função com domínio bidimensional. Foi-se utilizado o *software Octave* para a implementação do método, e com recursos gráficos da própria ferramenta facilitou-se a visualização dos resultados.

OBJETIVO

Tem como objeto de estudo a implementação de algoritmos para aproximar funções tabeladas em um domínio computacional bidimensional e incluir a familiarização de se trabalhar com métodos numéricos na prática.

FUNDAMENTAÇÃO TEÓRICA

1.1 PROBLEMA

Dado uma função f(x,y) com domínio computacional bidimensional e dividindo esse domínio em subintervalos de espaçamentos iguais, é possível representá-lo como uma malha de pontos discretizados. Dessa forma, dado um ponto (x,y) pertencente a esta malha, avaliar a função f(x,y). Para tal efeito é preciso, em primeira instância, localizar os pontos vizinhos ao ponto dado (x,y) que pertença a esta malha. Em seguida, determinar o Polinômio Bilinear que melhor aproxime a função f(x,y) dada. E por fim, avaliar o erro cometido.

1.2 MÉTODOS NUMÉRICOS

Primeiramente para entender o princípio que envolve aproximação de funções é preciso saber como determinar o polinômio que melhor se aproxima de uma função dada. Foi-se utilizado o Método dos Mínimos Quadrados (ou simplesmente MMQ), ou seja, nosso polinômio não obrigatoriamente passa pelos pontos x_i ele propriamente se aproxima de forma mínima dos mesmos.

Aproximaremos uma função f(x) por outra função g(x) de uma família previamente conhecida, dado que o domínio da f(x) seja discreto. A função g(x) é uma função linear do tipo $g(x) = ax + b$. Nossa tarefa é determinar os coeficientes a e b de modo que a soma dos quadrados dos erros em cada ponto, seja mínimo. Vamos definir:

$$e_i = f(x_i) - g(x_i) \Longrightarrow e_i = f(x_i) - ax_i - b$$

Queremos determinar a e b de forma que

$$M(a,b) = \sum_{i=0}^{N} (f(x_i) - ax_i - b)^2 \text{ seja minimizada.}$$

É necessário que:

$$\frac{\partial M}{\partial a} = \frac{\partial M}{\partial b} = 0$$

Ou seja,

$$\frac{\partial M}{\partial a} = 2\sum_{i=0}^{N} (f(x_i) - ax_i - b)(-x_i) = 0 \Rightarrow -\sum_{i=0}^{N} f(x_i)(x_i) + \sum_{i=0}^{N} ax_i^2 + \sum_{i=0}^{N} bx_i = 0$$

$$\sum_{i=0}^{N} ax_i^2 + \sum_{i=0}^{N} bx_i = \sum_{i=0}^{N} f(x_i)\,(x_i) \qquad (1)$$

$$\frac{\partial M}{\partial b} = \ 2\sum_{i=0}^{N}(f(x_i) - ax_i - b)(-1) = 0 \ \Rightarrow \ -\sum_{i=0}^{N} f(x_i) \ + \ \sum_{i=0}^{N} ax_i \ + \ \sum_{i=0}^{N} b = 0$$

$$\sum_{i=0}^{N} ax_i \ + \ \sum_{i=0}^{N} b = \ \sum_{i=0}^{N} f(x_i) \qquad (2)$$

Com as equações (1) e (2), e montando o sistema em notação matricial, temos:

$$\begin{pmatrix} \sum_{i=1}^{N} 1 & \sum_{i=1}^{N} x_i \\ \sum_{i=1}^{N} x_i & \sum_{i=1}^{N} x_i^2 \end{pmatrix} \begin{pmatrix} b \\ a \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{N} f(x_i) \\ \sum_{i=1}^{N} f(x_i).x_i \end{pmatrix}$$

Conseguimos ainda escrever na forma de Sistema Normal:

$$\begin{cases} bN + a\sum_{i=1}^{N} x_i = \sum_{i=1}^{N} f(x_i) & (3) \\ b\sum_{i=1}^{N} x_i + a\sum_{i=1}^{N} x_i^2 \ = \ \sum_{i=1}^{N} f(x_i).x_i & (4) \end{cases}$$

Isolando b de (3), temos:

$$b = \frac{\sum_{i=1}^{N} f(x_i) - \ a\sum_{i=1}^{N} x_i}{N}$$

Substituindo o valor de b em (4):

$$\left(\frac{\sum_{i=1}^{N} f(x_i) - \ a\sum_{i=1}^{N} x_i}{N}\right)\sum_{i=1}^{N} x_i + a\sum_{i=1}^{N} x_i^2 \ = \ \sum_{i=1}^{N} f(x_i).x_i$$

$$\frac{\sum_{i=1}^{N} f(x_i)\sum_{i=1}^{N} x_i}{N} - \ \frac{a\sum_{i=1}^{N} x_i \sum_{i=1}^{N} x_i}{N} + a\sum_{i=1}^{N} x_i^2 \ = \ \sum_{i=1}^{N} f(x_i).x_i$$

$$-\frac{a\sum_{i=1}^{N} x_i \sum_{i=1}^{N} x_i}{N} + a\sum_{i=1}^{N} x_i^2 \ = \ \sum_{i=1}^{N} f(x_i).x_i \ - \ \frac{\sum_{i=1}^{N} f(x_i)\sum_{i=1}^{N} x_i}{N}$$

$$a\left(\sum_{i=1}^{N} x_i^2 - \frac{\sum_{i=1}^{N} x_i \sum_{i=1}^{N} x_i}{N}\right) = \ \sum_{i=1}^{N} f(x_i).x_i \ - \ \frac{\sum_{i=1}^{N} f(x_i)\sum_{i=1}^{N} x_i}{N}$$

$$a\left(\frac{N\sum_{i=1}^{N}x_i^2-\ \sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}{N}\right)=\left(\frac{N\sum_{i=1}^{N}f(x_i).x_i\ -\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i}{N}\right)$$

$$a=\left(\frac{N\sum_{i=1}^{N}f(x_i).x_i\ -\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i}{N}\right)\left(\frac{N}{N\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}\right)$$

$$a=\left(\frac{N\sum_{i=1}^{N}f(x_i).x_i\ -\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i}{N\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}\right)$$

Substituindo o valor de a em b:

$$b=\frac{\sum_{i=1}^{N}f(x_i)-\ a\sum_{i=1}^{N}x_i}{N}$$

$$b=\frac{\sum_{i=1}^{N}f(x_i)-\left(\frac{N\sum_{i=1}^{N}f(x_i).x_i\ -\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i}{N\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}\right)\sum_{i=1}^{N}x_i}{N}$$

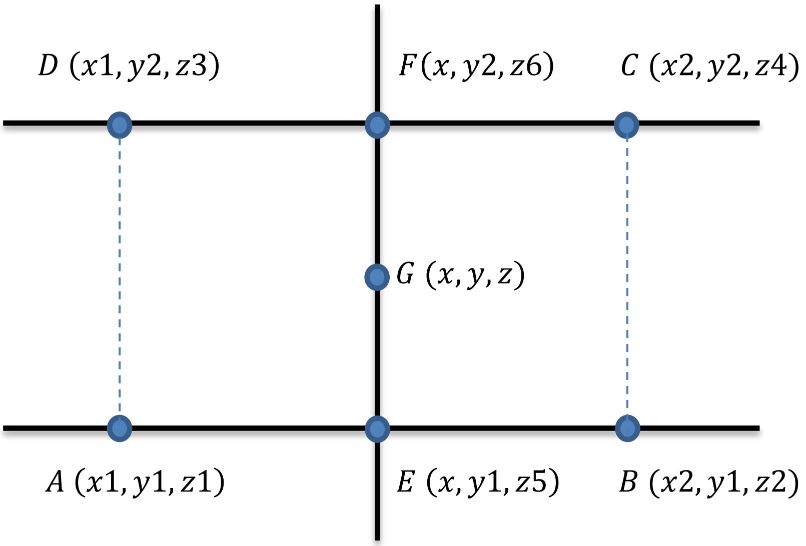
$$b=\frac{\sum_{i=1}^{N}f(x_i)-\left(\frac{N\sum_{i=1}^{N}f(x_i).x_i\ \sum_{i=1}^{N}x_i-\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}{N\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}\right)}{N}$$

$$b=\left(\frac{\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}f(x_i).x_i\ \sum_{i=1}^{N}x_i}{N\sum_{i=1}^{N}x_i^2-\ \sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}\right)$$

Dessa forma, a função minimizada pode ser escrita do seguinte modo:

$$M(a,b)=\left(\frac{N\sum_{i=1}^{N}f(x_i).x_i\ -\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i}{N\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}\right)x+\frac{\sum_{i=1}^{N}f(x_i)\sum_{i=1}^{N}x_i^2-\sum_{i=1}^{N}f(x_i).x_i\ \sum_{i=1}^{N}x_i}{N\sum_{i=1}^{N}x_i^2-\ \sum_{i=1}^{N}x_i\sum_{i=1}^{N}x_i}$$

O modelo geométrico apresentado abaixo colabora para a visualização de como achar o polinômio linear e bilinear que melhor se aproxima da função g(x,y):



Conhecidos os coeficientes a e b, as coordenadas do ponto A, B, C e D, resta nos encontrar $z,\ z_5\ e\ z_6$ dos pontos G, E e F, respectivamente. Lembrando que dados dois pontos é possível achar a equação que descreve a reta que passa por estes pontos (modelo linear), temos:

x_i	x_1	x_2
$f(x_i)$	z_1	z_2

N = 2;

$\sum_{i=1}^{N} f(x_i).x_i = x_1 z_1 + x_2 z_2;$

$\sum_{i=1}^{N} f(x_i) = z_1 + z_2;$

$\sum_{i=1}^{N} x_i = x_1 + x_2;$

$\sum_{i=1}^{N} x_i^2 = x_1^2 + x_2^2;$

$(\sum_{i=1}^{N} x_i)^2 = (x_1 + x_2)^2$;

Podemos determinar $z_5\ e\ z_6$ da seguinte forma:

$$z_5 = ax + b$$

$$z_5 = \left(\frac{N \sum_{i=1}^{N} f(x_i).x_i - \sum_{i=1}^{N} f(x_i) \sum_{i=1}^{N} x_i}{N \sum_{i=1}^{N} x_i^2 - \sum_{i=1}^{N} x_i \sum_{i=1}^{N} x_i} \right) x + \frac{\sum_{i=1}^{N} f(x_i) \sum_{i=1}^{N} x_i^2 - \sum_{i=1}^{N} f(x_i).x_i \sum_{i=1}^{N} x_i}{N \sum_{i=1}^{N} x_i^2 - \sum_{i=1}^{N} x_i \sum_{i=1}^{N} x_i}$$

$$z_5 = \left(\frac{2(x_1 z_1 + x_2 z_2) - (z_1 + z_2)(x_1 + x_2)}{2(x_1^2 + x_2^2) - (x_1 + x_2)^2} \right) x + \frac{(z_1 + z_2)(x_1^2 + x_2^2) - (x_1 z_1 + x_2 z_2)(x_1 + x_2)}{2(x_1^2 + x_2^2) - (x_1 + x_2)^2}$$

$$z_5 = \left(\frac{x_1 z_1 + x_2 z_2 - x_1 z_2 - x_2 z_1}{x_1^2 + x_2^2 - 2x_1 x_2} \right) x + \frac{x_1^2 z_2 + x_2^2 z_1 - x_1 x_2 z_2 - x_1 x_2 z_1}{x_1^2 + x_2^2 - 2x_1 x_2}$$

Analogamente:

$$z_6 = \left(\frac{x_1 z_3 + x_2 z_4 - x_1 z_4 - x_2 z_3}{x_1^2 + x_2^2 - 2x_1 x_2} \right) x + \frac{x_1^2 z_4 + x_2^2 z_3 - x_1 x_2 z_4 - x_1 x_2 z_3}{x_1^2 + x_2^2 - 2x_1 x_2}$$

Tendo quase todos os parâmetros definidos, fazendo-se valer ainda da definição de reta é possível achar a coordenada z do ponto G, já que conhecemos z_5 e z_6:

y_i	y_1	y_2
$f(x, y)$	$z_5(x)$	$z_6(x)$

$$z = \left(\frac{y_1 z_5 + y_2 z_6 - y_1 z_6 - y_2 z_5}{y_1^2 + y_2^2 - 2y_1 y_2} \right) y + \frac{y_1^2 z_6 + y_2^2 z_5 - y_1 y_2 z_6 - y_1 y_2 z_5}{y_1^2 + y_2^2 - 2y_1 y_2}$$

Ou ainda,

$$z(x, x_1, x_2, z_1, z_2) = \frac{x(x_1(z_1 - z_2) + x_2(z_2 - z_1)) + \ z_1 x_2(x_2 - x_1) + z_2 x_1(x_1 - x_2)}{(x_1 - x_2)^2}$$

$$z(x, x_1, x_2, z_1, z_2) = \frac{x(z_1 - z_2) - (z_1 x_2 - z_2 x_1)}{(x_1 - x_2)}$$

$$z(x, x_1, x_2, z_1, z_2) = \frac{z_1(x - x_2) + z_2(x_1 - x)}{(x_1 - x_2)}$$

A função acima descreve o **Polinômio Linear Aproximado**. De forma análoga, o **Polinômio Bilinear Aproximado**, pode ser escrito como:

$$z(x, y, x_1, x_2, y_1, y_2, z_1, z_2, z_3, z_4) = \frac{z(x, x_1, x_2, z_1, z_2)(y - y_2) + \ z(x, x_1, x_2, z_3, z_4)(y_1 - y)}{(y_1 - y_2)}$$

1.3 ALGORÍTMOS

Para entender como se foi implementado o método, apresentamos a seguir, em pseudocódigo, funções que possuem em seu cerne funcionalidades diferentes e essenciais para o desenvolvimento da solução do problema proposto.

Considerando:

I: matriz Antiga;

N: tamanho da matriz antiga;

x1: vetor com as abcissas;

y1: vetor com as ordenadas;

i: início do intervalo;

f: fim do intervalo;

a,b,c,d: os quatro pontos vizinhos;

A função Nova Malha tem por objetivo, duplicar a malha original gerando novos pontos (linha 2). Os pontos constituídos por abcissas e ordenadas são armazenados em um vetor com espaçamento de $\frac{1}{(N-1)}$ (linha 3 e 4). Após isso, são identificados os pontos vizinhos com suas limitações de ser um caso de fronteira ou não (linha 7) e assim é gerado uma nova malha com os pontos antigos e os novos pontos (linha 8).

```
1  Nova Malha (I, N, i, f)
2      N <- N*2
3      x1 <- [i : 1/(N-1) : f]
4      y1 <- x1
5       para i <- 1 ate N faca
6           para j <- 1 ate N faca
7               (a,b,c,d) <- Pontos Vizinhos(x1(i), y1(j), M, N)
8               I2(i,j) <- Bilinear(x1(i),y1(j),x(a),x(b),y(c),y(d),I(a,c),I(b,c),I(a,d),I(b,d))
9           fim
10      fim
11      retorna I2
12 fim
```

A função dos pontos vizinhos identifica os quatros pontos de vizinhança a, b, c, d, respectivamente da matriz antiga (linha 3 - 6).

```
1  Pontos Vizinhos (x1, y1, N)
2
3        a = (((N/2-1)*x1(i))+1)
4        b = a+1;
5        c = (((N/2-1)*y1(j))+1)
6        d = c+1;
7
8  fim
```

As funções Linear e Bilinear constituem em retornar o polinômio interpolador linear e bilinear, respectivamente, conforme foi abordado no Capítulo Métodos Numéricos - seção 1.2.

```
Linear(x,x1,x2,z1,z2)

     retorna ((z1*(x-x2)+z2*(x1-x))/(x1-x2))
fim
```

```
Bilinear (x,y,x1,x2,y1,y2,z1,z2,z3,z4)

     z5 <- Linear(x,x1,x2,z1,z2)
     z6 <- Linear(x,x1,x2,z3,z4)

     retorna ((z5*(y-y2)+z6(y1-y))/(y1-y2))
fim
```

A função Erro Cometido tem como objetivo avaliar num ponto contido na malha (linha 3 e 4), o erro em valor absoluto da diferença entre o polinomio aproximador bilinear (linha 7) e a função original proposta (linha 8).

```
1  Erro_Cometido (N, I)
2
3       x1 <- (1/256)
4       y1 <- x1
5
6       (a,b,c,d) <- Pontos Vizinhos (x1, y1, N)
7       z1 <- bilinear (x1, y1, x(a), x(b), y(c), y(d), I(a,c), I(b,c), I(a,d), I(b,d))
8       real <- cos(2*pi*x1)*sin(2*pi*y1)
9
10      return (abs(z1 - real))
11
12 fim
```

RESULTADOS

2.1 GRÁFICOS

Para teste do programa utilizamos a função $f(x,y) = \cos(2\pi x) * \sin(2\pi y)$ definida num domínio bidimensional [0,1]x[0,1] em escala de cores. O seguinte resultado pode ser avaliado pelos gráficos abaixo:

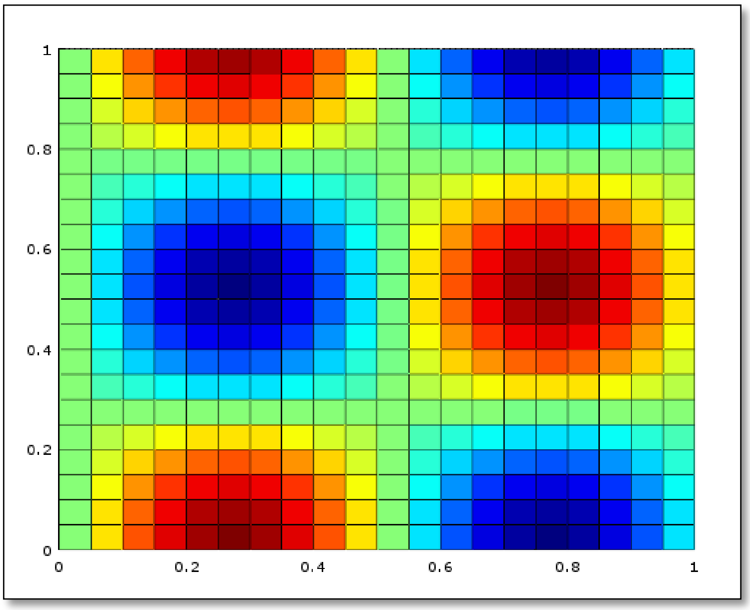


Gráfico 1- Matriz Original de ordem [20x20]

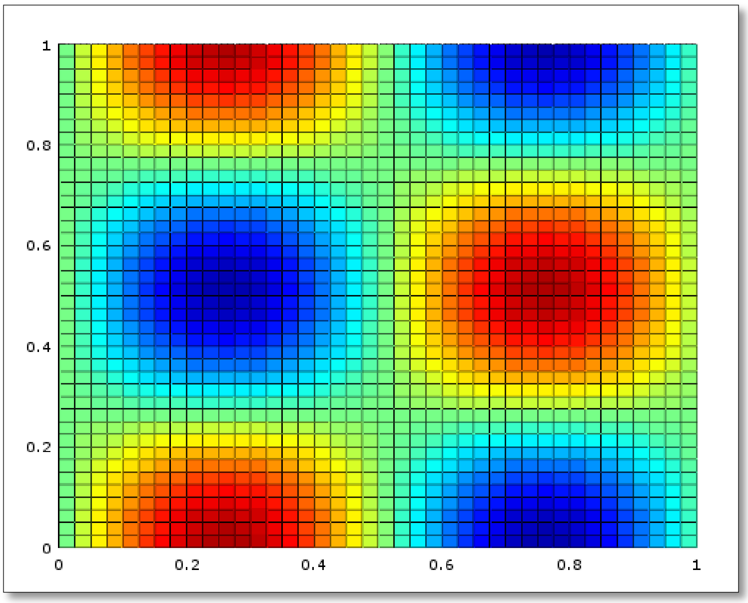


Gráfico 2 – Matriz Nova de ordem [40x40]

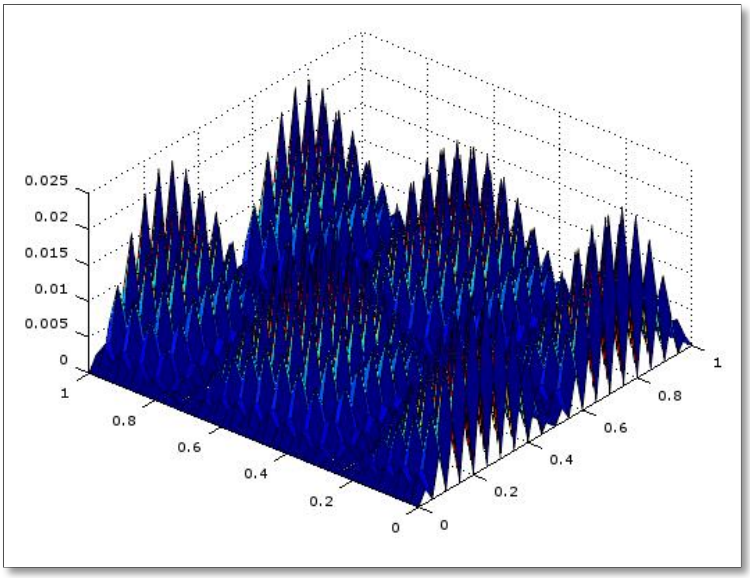


Gráfico 3 – Matriz $|p(x,y) - f(x,y)|$ [40x40]

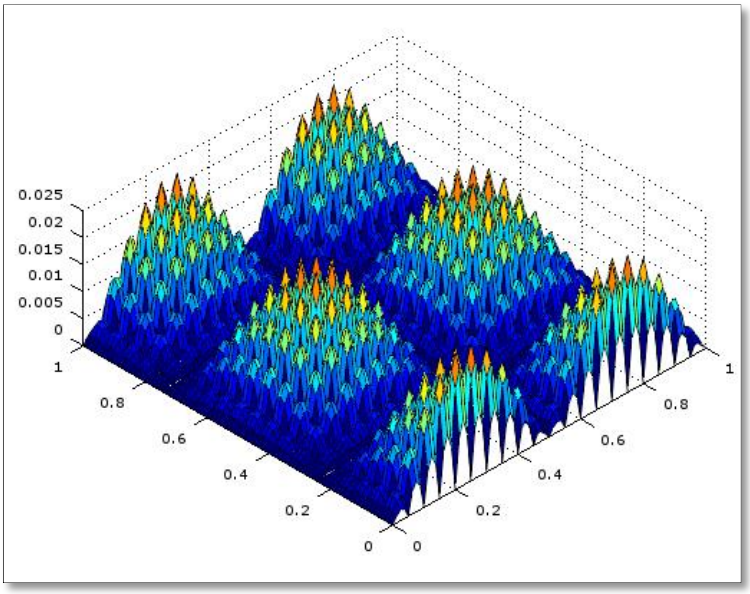


Gráfico 4 – Matriz $|p(x,y) - f(x,y)|$ [80x80]

2.2 ANÁLISE OBJETIVA

É pertinente observar que a após a duplicação da ordem da matriz original a malha refinou-se, apresentando-se sem tamanhos desfoques. O que de forma intrínseca, deixa evidente que o polinômio bilinear aproximador está sendo de fato colaborativo para o melhoramento ou refinamento da função plotada em escala de cores.

O fato da função ou "imagem" associada à função melhorar se dá porque a aproximação valida-se de dados conhecidos para estimar valores em pontos desconhecidos. Além do mais, o refinamento de uma imagem funciona em duas direções, tentando achar uma melhor aproximação da cor e intensidade de um pixel com base em valores de pixels adjacentes (representado por nós como os pontos vizinhos da nova malha ou matriz criada após a duplicação de sua ordem).

Para garantir nosso resultado, utilizamos o seguinte modelo de tabela para verificar se o programa implementado está ou não convergindo para um resultado esperado de ordem 2^p, sendo p a ordem de convergência do método utilizado.

N	Erro	Razão
8	$E_8 = \lvert p(x,y) - f(x,y) \rvert$	$-$
16	$E_{16} = \lvert p(x,y) - f(x,y) \rvert$	E_8/E_{16}
32	$E_{32} = \lvert p(x,y) - f(x,y) \rvert$	E_{16}/E_{32}
64	$E_{64} = \lvert p(x,y) - f(x,y) \rvert$	E_{32}/E_{64}
128	$E_{128} = \lvert p(x,y) - f(x,y) \rvert$	E_{64}/E_{128}

A tabela preenchida a seguir foi realizada escolhendo um ponto de forma que ele coincidisse em todas as novas malhas ou matrizes aumentadas criadas. Foi escolhido o ponto $\left(\frac{1}{256}, \frac{1}{256}\right)$.

N	Erro	Razão
8	$E_8 = 2.6390\text{x}10^{-3}$	$-$
16	$E_{16} = 7.2991\text{x}10^{-4}$	3.6155
32	$E_{32} = 2.0612\text{x}10^{-4}$	3.5412
64	$E_{64} = 5.9051\text{x}10^{-5}$	3.4905
128	$E_{128} = 1.4776\text{x}10^{-5}$	3.9964

Tabela 1 – Tabela de Erros Cometidos com Ordem de Convergência.

CONCLUSÃO

Apesar das dificuldades encontradas, pudemos perceber a grande importância do domínio do conhecimento dos métodos numéricos, em particular referente à obtenção de aproximações de funções tabeladas num domínio computacional bidimensional. Além de, obviamente, aprender a como aplicar este método computacionalmente, que viabilizou a obtenção de cálculos, antes, não facilmente obtidos à mão.

No quesito resultados, conforme a tabela de Erros Cometidos podemos tirar que a ordem de convergência do método utilizado é $p = 2$, isto é:

$$2^p \Rightarrow 2^2 = 4.$$

Como bem apresentado, na ultima linha da Tabela 1, a malha de ordem N = 128, possui razão aproximada do valor de convergência do método, ou numericamente expressando, $3.9964 \cong 4.00$. Conferindo ao nosso método de aproximação de funções um resultado satisfatório, coaduno ao valor de convergência esperado.

REFERÊNCIAS BIBLIOGRÁFICAS

PRESS, William H.; TEUKOSLSKY, Saul A.; VETTERLING, William T.; FLANNERY, Brian P. Numerical Recipes – The Art of Scientific Computing - Cambridge University Press. New York : 2007. - 3ª edição.

RUGIERO, Marcia A. Gomes; LOPES, Vera Lúcia da Rocha. Cálculo Numérico – Aspectos Teóricos e Computacionais – São Paulo – Pearson: 1998. – 2ª edição.