République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITÉ MOHAMED KHIDER, BISKRA

FACULTÉ des SCIENCES EXACTES et des SCIENCES de la NATURE et de la VIE

DÉPARTEMENT DE MATHÉMATIQUES

Mémoire présenté en vue de l'obtention du Diplôme **:**

MASTER en Mathématiques

Option :**Analyse**

Par

HELIS Rahma

Titre :

Interpolation 2D

Membres du Comité d'Examen :

Dr.	**BELLAGOUNE Abdelghani**	**UMKB**	**Président**
Pr.	**KHELIL Naceur**	**UMKB**	**Encadreur**
Dr.	**SOLTANI Siham**	**UMKB**	**Examinateur**

Juin 2021

Dédicace

Tout d'abord, je remercie bien "ALLAH" qui m'a accordé la volonté, la puissance et le courage pour terminer ce mémoire.

Je dédie ce modeste travail :

À mes chèrs parents , pour mon père " Abderrahim" .

Et ma mère "Saida".

À mes très chèrs frères et mes soeurs et leurs enfants chacun par son nom.

À mes meilleurs amies, Merci pour les très bons moments qu'on avait partagé ensemble.

Je dédie aussi mon travail à tout qui m'ont enseigné au long de ma vie scolaire.

À tous ceux qui m'aiment .

À tous ceux que j'aime.

À tous mes collègues de la promotion 2021.

A vous tous, Merci.

REMERCIEMENTS

Je tiens tout d'abord à remercier Mr. KHELIL Naceur, Professeur à l'Université de Biskra, qui m'a donné ce sujet et m'a encadré pendant mon Master. Il a toujours été à mon écoute et son point de vue complémentaire est souvent été trés utile.

Je suis trés reconnaissant envers Dr. BELLAGOUNE Abdelghani, et Dr.SOLTANI Siham d'avoir manisfesté de l'interêt pour mon travail en acceptant de participer à ce jury. Leurs remarques et commentaires constructifs m'ont permis d'en améliorer le manuscrit.

Enfin, que toutes les personnes qui ont partcipé de prés ou de loin à la réalisation de ce travail trouvent ici l'expression de mes sincères remerciements.

Mes derniers remerciements vont à mes parents, mes frères et soeurs et tous les amis.

ملخص

لا احد يشك في أهمية الاستقطاب, الاستقطاب يكون مفيدا عندما نريد تقريب دالة ذات متغير واحد بواسطة دالة أخرى بسيطة (مثلا كثيرات الحدود).

لسوء الحظ , أكثر الدوال انتشارا في الطبيعة هي الدوال ذات متغيرين (معالجة الصور).

في هذه الأطروحة سوف ندرس الاستقطاب لدالة ذات متغيرين مع التركيز على الاستقطاب ثنائي الخطية.

RESUME

Nul ne doute de l'importance de l'interpolation ; L'interpolation est utile lorsque on souhaite approximer une fonction d'une seule variable par une autre fonction mathématiquement plus simple, un polynôme par exemple, ou de prédire un modèle.
Malheureusement les applications les plus répandues dans la nature sont les applications de deux variables (traitement d'images).
Dans ce mémoire nous allons présenter l'interpolation de deux variables et nous mettons l'accent sur l'interpolation bilinéaire.

Abstract

No one doubts the importance of interpolation ; Interpolation is useful when we want to approximate a function of a single variable by a another mathematically simpler function, a polynomial for example,or predict a pattern.
Unfortunately the most widespread applications in nature are applications of two variables (image processing).
In this thesis we will present the interpolation of two variables and we focus on bilinear interpolation.

Table des matières

Table des figures

Introduction

Motivation

Nul ne doute de l'importance de l'interpolation ; L'interpolation est utile lorsque on souhaite approximer une fonction d'une seule variable par une autre fonction mathématiquement plus simple, un polynôme par exemple, ou de prédire un modèle (voir 4 ou 5).
Malheureusement, les applications les plus répandues dans la nature sont les applications de deux variables (traitement d'images) 4.

L'interpolation peut être étendue à des fonctions de plusieurs variables. La procédure consiste en interpoler sur une variable, fixer les valeurs des autres variables puis combiner les résultats.

On définit le polynôme d'interpolation de Lagrange en deux dimensions par :

$$P(f; x, y) = \sum_{i=0}^{n} \sum_{j=0}^{m} f(x_i, y_j) L_i(X; x) L_j(Y; y)$$

avec $|X| = n, |Y| = m$ et $L_i(X; x)$ et $L_j(Y; y)$ sont calculés à partir d'une interpolation 1D en fixant l'autre variable et vice versa (voir 3)

Organisation du mémoire

La partie principale de ce mémoire est composée de quatre chapitres, Le mémoire comporte également, un résumé, une introduction et une conclusion.

Le chapitre 1 est consacré à un rappel de l'interpolation 1D avec des exemples et de quelques programmes Matlab.

Dans le chapitre 2 on définit l'interpolation 2D qui consiste en interpoler sur une variable, fixer les valeurs des autres variables puis combiner les résultats. Ce chapitre est doté, aussi avec des exemples et de quelques programmes Matlab.

Dans le chapitre 3 nous mettons l'accent sur l'interpolation bilinéaire l'interpolation 2D la plus utilisée dans le traitement d'image.

Enfin dans le chapitre 4, on montre l'utilité de l'interpolation pour le traitement d'image. Et on termine par une conclusion 4.3 .

Chapitre 1

L'interpolation 1D

1.1 introduction

Définition 1.1.1 *E'tant donné les points* (xi, fi) *,*$i = 0, 1, 2, ..., n$ *d'une fonction* f *différentiable inconnue , on veut trouver une fonction* g *pour l'approcher.*

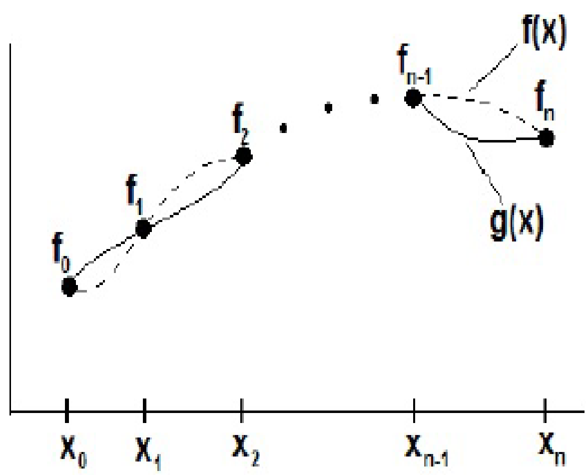


Fig. 1.1 – Interpolation simple

L'interpolation est une technique mathématique qui permet de décrire un ensemble de points en utilisant une certaine fonction. L'interpolation est utile lorsque on souhaite approcher une fonction par une autre fonction mathématiquement plus simple, un poly par exemple :$g(x)$ =$p_n(x) = a_0x^n + a_1x^{n-1} + ... + a_{n-1}x + a_n$

Ce polynôme p est appelé polynôme d'interpolation.Voici quelques fondements de base liés au polynôme d'interpolation. 4

Existance du polynôme d'interpolation

Le polynôme d'interpolation p doit inclure en chaque point donné :

$x = x_0 : p_n(x_0) = a_0x_0^n + a_1x_0^{n-1} + ... + a_{n-1}x_0 + a_n = f_0$

$x = x_1 : p_n(x_1) = a_0x_1^n + a_1x_1^{n-1} + ... + a_{n-1}x_1 + a_n = f_1$

...

$x = x_n : p_n(x_n) = a_0x_n^n + a_1x_n^{n-1} + ... + a_{n-1}x_n + a_n = f_n$

Exprimé en notation matricielle :

$$\begin{pmatrix} x_0^n & x_0^{n-1} & .. & x_0 & 1 \\ x_1^n & x_1^{n-1} & .. & x_1 & 1 \\ .. & .. & .. & .. & .. \\ x_n^n & x_n^{n-1} & .. & x_n & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ .. \\ a_n \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ .. \\ f_n \end{pmatrix}$$

Exemple 1.1.1 *on cherche à déterminer le polynôme de degré* 1, $p_1(x) = a_0x + a_1$ *passant par les points* $(-3, -1)$ *et* $(0, 2)$. *Les* a_i *peuvent être déterminés par la résolution du système linéaire :*

$$\begin{pmatrix} -3 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} -1 \\ 2 \end{pmatrix}$$

$$\begin{cases} -3a_0 + a_1 = -1 \\ a_1 = 2 \end{cases} \Rightarrow \begin{cases} a_0 = 1 \\ a_1 = 2 \end{cases}$$

La solution est $(1, 2)^T$. Donc ,le polynôme recherché est : $p_1(x) = x + 2$.

Unicité du polynôme d'interpolation

La matrice de coefficients (matrice de Vandermonde)

$$D = \begin{pmatrix} x_0^n & x_0^{n-1} & .. & x_0 & 1 \\ x_1^n & x_1^{n-1} & .. & x_1 & 1 \\ .. & .. & .. & .. & .. \\ x_n^n & x_n^{n-1} & .. & x_n & 1 \end{pmatrix}$$

Le déterminant

$$\mid D \mid = \prod_{j<i}^{n} (x_j - x_i) \neq 0 \quad ssi \quad x_j \neq x_i$$

Compte tenu des donnée $(x_i, f_i), i = 0, 1, 2, ..., n$, le polynôme d'interpolation qui inclut tous les points est unique. Pour la démonstration, il suffit de supposer qu'il existe deux polynômes et de démontrer qu'ils sont identiques.

1.2 Polynôme d'interpolation de Lagrange

1.2.1 L'interpolation de lagrange(1736-1813)

Soit $x_0, x_1, ..., x_n, n+1$ réeis distincs . Il existe $n+1$ polynômes L_i pour $i = 0$ $\bar{a}$ n définis par :

$$L_i = \frac{(x-x_0)(x-x_1)....(x-x_{i-1})(x-x_{i+1})(x-x_n)}{(x_i-x_0)(x_i-x_1)....(x_i-x_{i-1})(x_i-x_{i+1})(x_i-x_n)}$$

Ou sous une forme contractée :

$$L_i = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j}$$

Les propriétés suivantes peuvent notamment être démontrées :

1. Li est un polynôme de degré n
2. $L_i(x_i) = 1$
3. $L_i(x_j) = 0, \quad j \neq i$

Les polynômes de Lagrange forment une base (famille libre) de $\mathbb{R}_n[x]$ et par corollaire, on en déduit que tout polynôme de degré n s'écrit comme combinaison linéaire des L_i.
En pacticulier, interpoler la fonction f par un polynôme p de degré $\leq n$, aux $n+1$ points $x_0, x_1, ..., x_n$, revient à trouver p tel que :

$$p(x_i) = f(x_i) \quad \forall i = \{0, ..., n\}$$

Si un tel polynôme existe, il s'écrit de manière unique :

$$p(x) = \sum_{i=0}^{n} \alpha_i L_i(x)$$

En prenant $x = x_j$, on a alors :

$$p(x_j) = \alpha_0 \underbrace{L_0(x_j)}_{0} + ... + \alpha_j \underbrace{L_j(x_j)}_{1} + ... + \alpha_n \underbrace{L_n(x_j)}_{0} = \alpha_j = f(x_j)$$

Par conséquent :

$$p_n(x) = \sum_{i=0}^{n} f(x_i) L_i(x) \tag{1.1}$$

Remarquons tout d'abord que $p_n(x)$ est un polynôme de degré n (somme des $L_i(x)$ qui sont des polynômes de degré n) et $p_n(x_i) = f_i$; c-à-d $p_n(x)$ est un polynôme d'interpolation.

Exemple 1.2.1 *Le polynôme d'interpolation de Lagrange qui passe par les points :*

(2,5) , (4,6) , (5,3) est :

$$\begin{aligned} p_2(x) &= f_0L_0(x) + f_1L_1(x) + f_2L_2(x) \\ &= 5(\tfrac{x^2-9x+20}{6}) + 6(\tfrac{x^2-7x+10}{-2}) + 3(\tfrac{x^2-6x+8}{3}) \\ &= -\tfrac{7}{6}x^2 + \tfrac{15}{2}x - \tfrac{16}{3}. \end{aligned}$$

Exemple 1.2.2 *On connaît trois points* $(0,1),(2,5),(4,17)$. *Les polynômes de Lagrange associés sont :*

$$\begin{aligned} L_0(x) &= \tfrac{(x-2)(x-4)}{8}. \\ L_1(x) &= \tfrac{x(x-4)}{-4}. \\ L_2(x) &= \tfrac{x(x-2)}{8}. \end{aligned}$$

Le calcule du polynôme d'interpolation fournit

$$p(x) = L_0(x) + 5L_1(x) + 17L_2(x) = x^2 + 1.$$

1.2.2 Erreur d'interpolation de Lagrange

Théorème 1.2.1 *Pour* $(x_i, f_i), i = 0, 1, ..., n, f$ *étant inconnue.* $p_n(x)$ *étant le polynôme d'interpolation,* $(p_n(x_i) = f_i, i = 0, 1, ..., n.)$.*L'erreur en un point qelque* t *est par définition :*

$E_n(t) = f(t) - p_n(t)$ **Formule de l'erreur**.

On considère la fonction auxiliaire $h(x) = f(x) - P_n(x) - K(x-x_0)(x-x_1).....(x-x_n)$ le K étant choisi de telle sorte que $h(\overline{x}) = 0, \overline{x} \neq x_i, \forall i = \overline{0,n}$.

On Vérifie que $h(x)$ s'annulle en $(n+2)$ points, et en utilisant le théorème de Rolle (pour tout fonction définie) , on démontre l'éxistence d'un z compris entre x_0 et x_n tel que :

$f(x) - p_n(x) = \frac{f^{(n+1)}(z)}{(n+1)!}(x-x_1)....(x-x_n)$

donc **Formule de l'erreur :**

$$E(x) = \frac{f^{(n+1)}(z)}{(n+1)!}(x - x_0)(x - x_1)...(x - x_n) \quad x \neq x_i, z \in [x_0, x_n] \tag{1.2}$$

Remarque 1.2.1

– Si $f(x)$ est un polynôme d'ordre $p \leq n$ alors l'erreur s'annule car :

$$f^{(n+1)}(x) = 0.$$

– l'erreur diminue en prenant plus de points et sera d'autant plus petite que x est proche d'un point d'interpolation f ne varie pas trop brusquement.

Corollaire 1.2.1 *Supposons que f soit $(n+1)$ fois continûment dérivable sur l'intervalle $[a, b]$.Alors si p_n est le polynôme d'interpolation de degré n , on a :*

$$| f(x) - p_n(x) | \leq \frac{M_{n+1}}{(n+1)!} | \prod_{i=0}^{n}(x - x_i) | \tag{1.3}$$

où

$$M_{n+1} = \max_{x \in [a,b]} | f^{(n+1)}(x) | .$$

En particulier,

$$\max_{x \in [a,b]} | f(x) - p_n(x) | \leq \frac{M_{n+1}}{(n+1)!} \max_{x \in [a,b]} | g(x) | \tag{1.4}$$

Où $g(x)$ est le polynôme de degré $n + 1$ définie par :

$$g(x) = \prod_{i=0}^{n}(x - x_i) = (x - x_0)(x - x_1)...(x - x_n) \tag{1.5}$$

Théorème 1.2.2 (de Rolle généralisé) *Rappelons que théorème de Rolle ordinaire affirme que si f est une fonction continue sur $[a, b]$ et dérivable sur $]a, b[$ telle que $f(a) = f(b)$,il existe $c \in]a, b[$ tel que $f\prime(c) = 0$, Ici, nous aurons besoin de la forme plus générale suivante.*

Si u est un fonction continue sur $[a, b]$ et k fois dérivable sur $]a, b[$ qui s'annule en $k + 1$ points x_i,$i = 0, ..., k$, alors il existe $c \in]a, b[$ tel que $u^{(k)}(c) = 0$.

Preuve. (du théorème 1.2.1 **)** ■

Démonstration. Fixons $x \in [a, b]$, si $x \in A = \{x_0, x_1, ..., x_n\}$, n'importe quel z convient ,(Dans ce cas ,la formule donne seulement $0 = 0$),Nous supposerons que $x \notin A$. Notons g le polynôme défini par : $g(t) = (t - x_0)(t - x_1)...(t - x_n)$ et prenons $k = k(x) \in \mathbb{R}$ tel que :

$$f(x) - p_n(x) = k(x)g(x) \tag{1.6}$$

Un tel nombre existe ; il suffit de prendre

$$k(x) = \frac{f(x) - p_n(x)}{g(x)} \tag{1.7}$$

qui est bien défini car, comme $x \neq x_i, (i = \overline{0, n})$, le dénominateur ne s'annule pas. Considérons maintenant le fonction u définie sur l'intervalle $[a, b]$ par la relation :

$$u(t) = f(t) - p_n(t) - k(x)g(t) \quad t \in [a, b]. \tag{1.8}$$

Il faut prendre garde ici que x est un paramètre fixé et t est la variable. Puisque $f \in \varphi^{(n+1)}[a, b]$, nous avons aussi $u \in \varphi^{(n+1)}[a, b]$.De plus :

$$u(x_i) = f(x_i) - p_n(x_i) - k(x) \times 0 = f(x_i) - f(x_i) = 0 \quad 0 < i < n$$

et par définition de $k(x)$,

$$u(x) = f(x) - p_n(x) - k(x)g(x) = 0$$

Il suit que u s'annule en $n+2$ points, à savoir les $n+1$ points de A aux quels s'ajoute le point x. Le théorème1.2.2 de Rolle généralisé nous permet d'affirmer l'existence de $z \in \varphi^{(n+1)}]a,b[$ tel que $u^{(n+1)}(z) = 0$. Nous pouvons facilement calculer la dérivée $(n+1)-ième$ de u. D'apord, puisque $p_n(t)$ est un polynôme de degré n, sa dérivée $(n+1)-ième$ est nulle. Quant à $g(t)$, puisque :

$$g(t) = (t-x_0)(t-x_1)...(t-x_n) = t^{n+1} + (polynôme\ de\ degré < n).$$

Sa dérivée $(n+1)-ième$, est la constante $(n+1)!$. Finalement,

$$u^{(n+1)}(t) = f^{(n+1)}(t) - (n+1)!k(x). \tag{1.9}$$

En prenant $t = z$,il vient

$$0 = f^{(n+1)}(z) - (n+1)!k(x) \tag{1.10}$$

Revenant à la définition de $K(x)$, nous obtenons :

$$f(x) - p_n(x) = k(x)g(x) = \frac{f^{(n+1)}(z)}{(n+1)!}(x-x_0)(x-x_1)...(x-x_n) \tag{1.11}$$

Nous avons bien trouvé un nombre z dans $]a,b[$ vérifiant la formule annoncée.

Exemple 1.2.3 *On considère la fonction* $f(x) = \sin\frac{\pi}{2}x$.

1. Donner le polynôme d'interpolation $p(x)$ de degré 2 associé à f aux points $x_i = 0, 1, 2$.
2. Donner l'expression de l'erreur $f(x) - p(x)$.
3. Montrer que :

$$| E(x) |=| f(x) - p(x) |\leq| \frac{\pi^3}{48} x(x-1)(x-2) | .$$

Estimer l'erreur pour $x = \frac{1}{2}$ et comparer la à l'erreur effective.

Solution

1.On a $f(0) = 0$, $f(1) = 1$, $f(2) = 0$. On cherche $p(x)$ tel que $p(x) = a_0 + a_1 x + a_2 x^2$. avec $p(0) = 0, p(1) = 1, p(2) = 0$, on a alors :

$$\begin{cases} a_1 + a_2 = 1 \\ 2a_1 + 4a_2 = 2 \end{cases} \Leftrightarrow \begin{cases} a_2 = -1 \\ a_1 = 2 \end{cases}$$

Donc :

$$p(x) = -x^2 + 2x.$$

2.L'erreur d'interpolation s'écrit sous la forme :

$$E(x) = f(x) - p(x) = \frac{f^{(3)}(\zeta_x)}{3!} \pi(x).$$

avec $\pi(x) = x(x-1)(x-5)$ et $\zeta_x \in [0, 2]$.

3.On a $f^{(3)}(x) = -\frac{\pi^3}{8} \cos \frac{\pi}{2} x$, donc :

$$\begin{aligned} | f(x) - p(x) | &\leq \tfrac{\pi^3}{48} \sup_{x \in [0,2]} | \cos \tfrac{\pi}{2} x || \pi(x) | \\ &\leq | \tfrac{\pi^3}{48} x(x-1)(x-2) | . \end{aligned}$$

L'estimation de l'erreur pour $x = \frac{1}{2}$ est $\frac{\pi^3}{128}$ et on trouve :

$$\frac{\pi^3}{128} \backsimeq 0.242 = 24.2 \times 10^{-2} \tag{1.12}$$

On calcule maintenant l'estimation de l'erreur effective, on a :

$$f(\tfrac{1}{2}) = \sin \tfrac{\pi}{4} = \tfrac{\sqrt{2}}{2} \quad et \quad p(\tfrac{1}{2}) = \tfrac{3}{4}$$

alors

$$| f(\frac{1}{2}) - p(| \frac{1}{2} |) |=| \frac{\sqrt{2}}{2} - \frac{3}{4} |\leq 4 \times 10^{-2}.$$

Donc l'estimation de l'erreur donnée par 1.12 est moins fine que l'erreur effective.

Programmation de la méthode de Lagrange

```
function p=lagrange(x,f,v)
 n=length(x);
 syms t;          % Variable pour le polynôme
 p=0;
  for i=1:n
    L=1;
     for j=1:n
         if i~=j
           L=L*(t-x(j))/(x(i)-x(j));
         end
     end
   p=p+L*f(i);          % obtentionde p(t) symboliquement
  end
  p=expand(p);             % simplification algébrique
    if nargin==3;             % vérifie s'il y a un paramètre supplémentaire
       t=v;
      p=eval(p);              % obtention du résultat de p évalué en v
   end
end
```

Application

```
>> x=[2,4,5];
>> f=[5,6,3];
>> p=lagrange(x,f)
p =
(15*t)/2 - (7*t^2)/6 - 16/3
```

Exemple 1.2.4 *Soit $x = \{-1, 1, 2\}$ et $y = exp(x)$, nous déterminerons, analytiquement, le polynôme $P_2(x)$ par l'utilisation de la formule d'interpolation de Lagrange 1.1 et pour $i = 0, 1, 2$, il vient :*

$$
\begin{aligned}
p_2(x) &= \sum_{i=0}^{2} y_i L_i(x) \\
&= y_0 L_0(x) + y_1 L_1(x) + y_2 L_2(x) \\
&= y_0 \tfrac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + y_1 \tfrac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + y_2 \tfrac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\
&= (\exp)^{-1} \tfrac{(x-1)(x-2)}{6} + (\exp) \tfrac{(x+1)(x-2)}{(-2)} + (\exp)^2 \tfrac{(x+1)(x-1)}{3} \\
&= \left(\tfrac{(\exp)^{-1} - 3(\exp) + 2(\exp)^2}{6}\right) x^2 + \left(\tfrac{-3(\exp)^{-1} + 3(\exp)}{6}\right) x + \tfrac{2(\exp)^{-1} + 6(\exp) - 2(\exp)^2}{6}
\end{aligned}
$$

On va tracer, sur la même figure, la fonction et le polynôme d'interpolation :

Voici le script Matlab :

```
n=3;   x=[-1 1 2];
y=exp(x);
m=10;   dx=(x(3)-x(1))/m;
xx=x(1) :dx :x(3);
polyn=0;
plot(xx,exp(xx),'-r','linewidth',2)
for i=1 :n
lag=1;
for j=1 :n
```

```
      if (i~=j)
        lag=((xx-x(j))./(x(i)-x(j))).*lag;
      end
    end
      polyn=polyn+lag.*y(i);
end
hold on
plot(xx,polyn,'–b','Linewidth',2);
 for i=1 :n
     plot(x(i),y(i),'k.','MarkerSize',25,'Linewidth',1.5);
 hold on
 end
xlabel('\fontsize{12}\fontname{Tex} x');
ylabel('\fontsize{12}\fontname{Tex} y=exp(x)');
legend('Solution Exacte','Solution Approchée',2);
title('Interpolation de Lagrange de y=exp(x)');
```

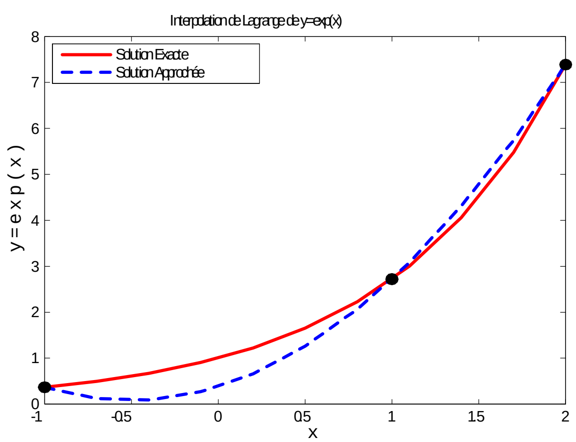


FIG. 1.2 – Interpolation de Lagrange de y=exp(x)

Chapitre 2

L'interpolation 2D

2.1 Généralisation

L'interpolation peut être étendue à des fonctions de plusieurs variables. La procédure consiste en interpoler sur une variable, fixer les valeurs des autres variables puis combiner les résultats.

On définit le polynôme d'interpolation de de Lagrange en deux dimensions par

$$P(f; x, y) = \sum_{i=0}^{n} \sum_{j=0}^{m} f(x_i, y_j) L_i(X; x) L_j(Y; y)$$

avec $|X| = n, |Y| = m$ et $L_i(X; x)$ et $L_j(Y; y)$ sont calculés Ă partir d'une interpolation 1D en fixant l'autre variable et vice versa Si $|X| = 2, |Y| = 2$ et $L_i(X; x)$ Alors on parle d'interpolation bilinéaire et on aura :

$$P(f; x, y) = \sum_{i=0}^{2} \sum_{j=0}^{2} f(x_i, y_j) L_i(X; x) L_j(Y; y)$$

Dans ce mémoire, nous utiliserons le polynôme de Lagrange dans un exemple contenant des données d'une fonction qui dépend de deux variables. Il n'est pas intéressant de trouver la forme analytique du polynôme d'interpolation qui aurait des termes avec plus d'une variable.

Exemple 2.1.1

Utiliser toutes les données tabulées ci-dessous, d'une fonction f de deux variables indépendantes x, y ,pour estimer $f(3, 12)$ par interpolation :

$x \diagdown y$	5	10	15	20
2	3.7	4.2	5.8	7.1
4	4.1	5.3	6.1	7.9
6	5.6	6.7	7.4	8.2

Nous interpolons d'abord pour $x = 3$ avec les données de chaque colonne $y = 5, 10, 15, 20$. Nous devons utiliser un polynôme du second degré car il y a trois données dans la direction x :

$$p_2(x) = \sum_{i=0}^{2} f_i L_i(x) = f_0 L_0(x) + f_1 L_1(x) + f_2 L_2(x).$$

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

La forme algèbrique n'est pas requise. La valeur est directement substituée pour interpoler $x = 3$.

$$L_0(3) = \prod_{\substack{j=0 \\ j\neq i}}^{2} \frac{(3-x_j)}{(x_0-x_j)} = \frac{(3-x_1)}{(x_0-x_1)} \frac{(3-x_2)}{(x_0-x_2)} = \frac{(3-4)}{(2-4)} \frac{(3-6)}{(2-6)} = 3/8.$$

$$L_1(3) = \prod_{\substack{j=0 \\ j\neq i}}^{2} \frac{(3-x_j)}{(x_1-x_j)} = \frac{(3-x_0)}{(x_1-x_0)} \frac{(3-x_2)}{(x_1-x_2)} = \frac{(3-2)}{(4-2)} \frac{(3-6)}{(4-6)} = 3/4.$$

$$L_2(3) = \prod_{\substack{j=0 \\ j\neq i}}^{2} \frac{(3-x_j)}{(x_2-x_j)} = \frac{(3-x_0)}{(x_2-x_0)} \frac{(3-x_1)}{(x_2-x_1)} = \frac{(3-2)}{(6-2)} \frac{(3-4)}{(6-4)} = -1/8.$$

Polynôme d'interpolation pour chaque colonne $y = 5, 10, 15, 20$:

$$p_2(3) = f_0L_0(x) + f_1L_1(x) + f_2L_2(x)$$
$$= f_0(3/8) + f_1(3/4) + f_2(-1/8).$$

Les valeurs de $L_i(3)$ sont les mêmes pour chaque colonne y :

Les valeurs de chaque colonne sont remplacées :

$y = 5 : p_2(3) = 3.7(3/8) + 4.1(3/4) + 5.6(-1/8) = 3.7625.$

$y = 10 : p_2(3) = 4.2(3/8) + 5.3(3/4) + 6.7(-1/8) = 4.7125.$

$y = 15 : p_2(3) = 5.8(3/8) + 6.1(3/4) + 7.4(-1/8) = 5.8250.$

$y = 20 : p_2(3) = 7.1(3/8) + 7.9(3/4) + 8.2(-1/8) = 7.5625.$

Avec les quatre résultats, il est interpolé en $y = 12$ avec un polynôme du troisième degré :

$x \diagdown y$	5	10	15	20
3	3.7625	4.7125	5.8250	7.5625

$$p_3(y) = \sum_{i=0}^{3} f_iL_i(y) = f_0L_0(y) + f_1L_1(y) + f_2L_2(y) + f_3L_3(y).$$
$$L_i(y) = \prod_{\substack{j=0 \\ j\neq i}}^{3} \tfrac{(y-y_j)}{(y_i-y_j)}. \quad i = 0, 1, 2, 3$$

La valeur à interpoler est directement substituée par l'autre variable $y = 12$:

$$L_0(12) = \prod_{\substack{j=0 \\ j\neq i}}^{3} \tfrac{(12-y_j)}{(y_0-y_j)} = \tfrac{(12-y_1)}{(y_0-y_1)} \tfrac{(12-y_2)}{(y_0-y_2)} \tfrac{(12-y_3)}{(y_0-y_3)} = \tfrac{(12-10)}{(5-10)} \tfrac{(12-15)}{(5-15)} \tfrac{(12-20)}{(5-20)} = -8/125.$$
$$L_1(12) = \prod_{\substack{j=0 \\ j\neq i}}^{3} \tfrac{(12-y_j)}{(y_1-y_j)} = \tfrac{(12-y_0)}{(y_1-y_0)} \tfrac{(12-y_2)}{(y_1-y_2)} \tfrac{(12-y_3)}{(y_1-y_3)} = \tfrac{(12-5)}{(10-5)} \tfrac{(12-15)}{(10-15)} \tfrac{(12-20)}{(10-20)} = 84/125.$$
$$L_2(12) = \prod_{\substack{j=0 \\ j\neq i}}^{3} \tfrac{(12-y_j)}{(y_2-y_j)} = \tfrac{(12-y_0)}{(y_2-y_0)} \tfrac{(12-y_1)}{(y_2-y_1)} \tfrac{(12-y_3)}{(y_2-y_3)} = \tfrac{(12-5)}{(15-5)} \tfrac{(12-10)}{(15-10)} \tfrac{(12-20)}{(15-20)} = 56/125.$$
$$L_3(12) = \prod_{\substack{j=0 \\ j\neq i}}^{3} \tfrac{(12-y_j)}{(y_3-y_j)} = \tfrac{(12-y_0)}{(y_3-y_0)} \tfrac{(12-y_1)}{(y_3-y_1)} \tfrac{(12-y_2)}{(y_3-y_2)} = \tfrac{(12-5)}{(20-5)} \tfrac{(12-10)}{(20-10)} \tfrac{(12-15)}{(20-15)} = -7/125.$$

Résultat final :

$y = 12 : p_3(12) = f_0L_0(12) + f_1L_1(12) + f_2L_2(12) + f_3L_3(12)$

$$= 3.7625(-8/125) + 4.7125(84/125) + 5.8250(56/125) + 7.5625(-7/125).$$

$$= 5.1121$$

$f(3, 12) = 5.1121$

Chapitre 3

L'interpolation bilinéaire

3.1 Introduction

Étant donné un ensemble de points d'échantillonnage $v(x_i) = f(x_i, y_i); (i = 1, ..., n)$ aux points 2-D $x_i = [x_i, y_i]^T$ dans une grille régulière, ou une grille irrégulière (points de données dispersés), on peut construire une fonction d'interpolation $f(X) = f(x, y)$ qui passe à travers tous ces points d'échantillonnage 3.

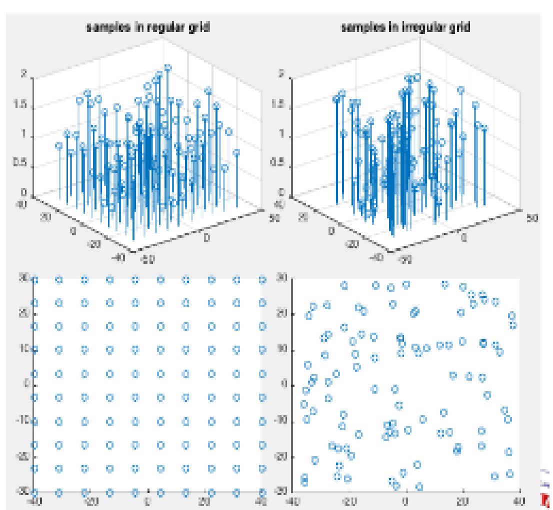


FIG. 3.1 – Données de la grille 3.

3.2 Interpolation bilinéaire

Ici, nous allons d'abord considérer les méthodes basées uniquement sur des grilles régulières. Dans la présente section, nous décrirons le cas le plus simple ; l'interpolation de 2D en cordonnées cartésiennes.

L'interpolation 2D traite la détermination des valeurs intermédiaires pour les fonctions de deux variables, $z = f(x_i, y_i)$. Comme le montre la figure **??**, nous avons des valeurs en quatre points : $f(x_1, y_1), f(x_2, y_1), f(x_1, y_2)$ et $f(x_2, y_2)$. Nous voulons estimer la valeur en un point intermédiaire $f(x_i, y_i)$. Si nous utilisons une fonction d'interpolation linéaire, le résultat est un plan reliant les points comme dans la Fig. **??**. Ces fonctions sont appelées bilinéaires.

De nombreux algorithmes de traitement d'image nécessitent de calculer la valeur d'un niveau sur un point qui ne coïncide pas avec un pixel. Cette valeur peut être obtenue par interpolation bilinéaire avec les 4 pixels voisins.

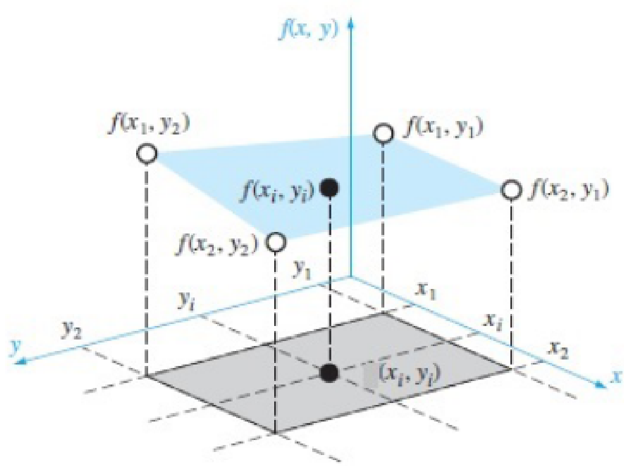


FIG. 3.2 – Représentation graphique de l'interpolation bilinèaire bidimensionnelle où une valeur intermédiaire (cercle plein) est estimée à partir de quatre valeurs données (cercles ouverts)

Une approche simple pour développer la fonction bilinéaire est représentée sur la figure 3.3.

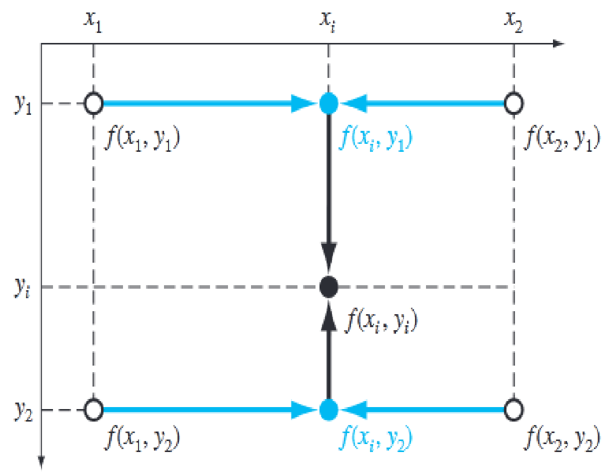


FIG. 3.3 – L'interpolation bilinéaire bidimensionnelle peut être mise en oeuvre en appliquant d'abord une interpolation linéaire unidimensionnelle dans la dimension x pour déterminer les valeurs en x_i. Ces valeurs peuvent ensuite être utilisées pour interpoler linéairement dans la dimension y afin d'obtenir le résultat final en $x_{i,}y_i$.

D'abord, nous pouvons maintenir la valeur y fixe et appliquer une interpolation linéaire unidimensionnelle dans la direction de x . En utilisant la forme de Lagrange, le résultat en (x_i, y_1) est

$$f(x_i, y_1) = \frac{x_i - x_2}{x_1 - x_2} f(x_1, y_1) + \frac{x_i - x_1}{x_2 - x_1} f(x_2, y_1) \tag{3.1}$$

et en (x_i, y_2) est

$$f(x_i, y_2) = \frac{x_i - x_2}{x_1 - x_2} f(x_1, y_2) + \frac{x_i - x_1}{x_2 - x_1} f(x_2, y_2) \tag{3.2}$$

Ces points peuvent ensuite être utilisés pour interpoler linéairement le long de la dimension y pour obtenir le résultat final

$$f(x_i, y_i) = \frac{y_i - y_2}{y_1 - y_2} f(x_i, y_1) + \frac{y_i - y_1}{y_2 - y_1} f(x_i, y_2) \tag{3.3}$$

Une équation unique peut être développée en remplaçant les équations. (3.1) et (3.2) dans l'équation. (3.3) pour donner

$$f(x_i, y_i) = \frac{x_i - x_2}{x_1 - x_2}\frac{y_i - y_2}{y_1 - y_2}f(x_1, y_1) + \frac{x_i - x_1}{x_2 - x_1}\frac{y_i - y_1}{y_2 - y_1}f(x_2, y_1) + \frac{x_i - x_2}{x_1 - x_2}\frac{y_i - y_1}{y_2 - y_1}f(x_1, y_2) + \frac{x_i - x_1}{x_2 - x_1}\frac{y_i - y_1}{y_2 - y_1}f(x_2, y_2) \tag{3.4}$$

Exemple 3.2.1 *Supposons que nous avons mesuré les températures à un certain nombre de coordonnées sur la surface d'une plaque chauffante rectangulaire :*

$$T(2,1) = 50 \quad T(9,1) = 56$$
$$T(2,6) = 57 \quad T(9,6) = 65.5$$

Utilisez l'interpolation bilinéaire pour estimer la température à $x_i = 4.5$ et $y_i = 3.8$

Solution En substituant ces valeurs dans Eq 3.4, on obtient :

$$\begin{aligned} f(4.5, 3.8) &= \frac{4.5 - 9}{2 - 9}\frac{3.8 - 6}{1 - 6}50 + \frac{4.5 - 2}{9 - 2}\frac{3.8 - 6}{1 - 6}56 \\ &+ \frac{4.5 - 9}{2 - 9}\frac{3.8 - 1}{6 - 1}57 + \frac{4.5 - 2}{9 - 2}\frac{3.8 - 1}{6 - 1}65.5 \\ &= 56.563 \end{aligned}$$

Notez qu'au-delà de la simple interpolation bilinéaire décrite dans l'exemple précédent, les polynômes d'ordre supérieur et les splines peuvent également être utilisés pour interpoler en deux dimensions. De plus, ces méthodes peuvent être facilement étendues à trois dimensions.

3.3 Programmation de la méthode de Lagrange 2D

La fonction suivante reçoit les points donnés et renvoie le polynôme d'interpolation de Lagrange.(voir le programme de lagrange 1.2.2)
En option, si la fonction reçoit comme paramètre en plus de la valeur à interpoler, le résultat livré est une valeur numérique, le résultat de l'interpolation.

On peut utiliser la fonction d'interpolation de lagrange d'une seule variable, pour interpoler une fonction de deux (ou plusieurs) variables. Lorsque la fonction est appliquée selon une direction, des résultats partiels seront obtenus. L'interpolation avec ces résultats produira le résultat final.

Pour l'exemple 2.1.1 ci-dessus :

$>> x = [2, 4, 6]\,;$ $\quad$ *% interpolation partielle en x pour chaque colonne de y*

$>> f = [3.7, 4.1, 5.6]\,;$

$>> r1 = lagrange(x, f, 3)\,;$

$>> f = [4.2, 5.3, 6.7]\,;$

$>> r2 = lagrange(x, f, 3)\,;$

$>> f = [5.8, 6.1, 7.4]\,;$

$>> r3 = lagrange(x, f, 3)\,;$

$>> f = [7.1, 7.9, 8.2]\,;$

$>> r4 = lagrange(x, f, 3)\,;$

$>> y = [5, 10, 15, 20]\,;$ $\quad$ *% interpolation en y avec les résultats partielles*

$>> f = [r1, r2, r3, r4]\,;$

$>> p = lagrange(y, f, 12)$

$p =$ Résultat final

5.1121

Si vous prévoyez d'utiliser fréquemment ce type d'interpolation,vous devez instrumenter une fonction en MATLAB pour interpoler en deux dimensions avec la formule de Lagrange.

Cette fonction sera appelée lagrange2, et elle utilisera la fonction lagrange en interne pour interpoler avec une variable, et avec les résultats, effectuez l'interpolation avec l'autre variable.

Soient :

x, y : vecteurs avec les valeurs des variables indépendantes X, Y

f : matrice avec les données de la variable dépendante organisées en lignes.

u, v : valeurs pour lesquelles l'interpolation sera effectuée en x et y respectivement.

Le programme interpolation 2D

```
function p=lagrange2(x,y,f,u,v)
 [n,m]=size(f) ;
  for i=1 :m  %chaque colonne est envoyé (résultats partielles)
    r(i)=lagrange(x,f( :,i),u) ;
  end
p=lagrange(y,r,v) ;  %interpolation finale dans l'autre direction
end
```

Application

```
>> x=[2,4,6] ;
>> y=[5,10,15,20] ;
>> f=[3.7, 4.2, 5.8, 7.1 ; 4.1, 5.3, 6.1, 7.9 ; 5.6, 6.7, 7.4, 8.2] ;
>> p=lagrange2(x,y,f,3,12)
p =
 5.1121
```

Chapitre 4

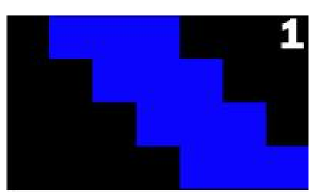
Application de l'interpolation

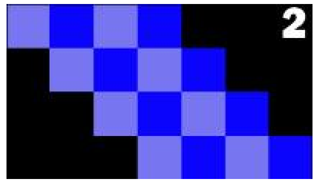
4.1 Introduction

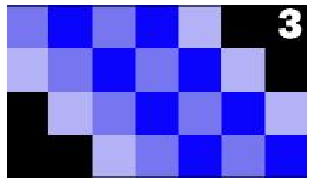
Dans ce chapitre nous éspérons vous donner ne serait-ce qu'une petite idée sur l'utilité de l'interpolation. Dans cette application nous nous concentrerons exclusivement sur la fonction de l'interpolation dans le contexte de la résolution d'image.

L'interpolation est un procédé d'estimation de valeur entre deux informations préexistantes. Il est utilisé dans les scanners, les appareils numériques et les logiciels d'édition d'images afin de produire une image avec une résolution (le nombre de pixels verticaux et horizontaux) plus grande que celle capturé par le capteur.

Simplement formulé, l'interpolation "invente" des pixels là où il n'y en avait pas et les intercale entre deux originaux pour augmenter la taille d'image. Si cela semble être de la triche-et c'est parfois effectivement le cas. Les images ci-dessous montrent le mécanisme d'une simple interpolation. Notez que dans une interpolation réelle des pixels de couleur, ceux-ci serait de couleur identique à celle des originaux environnants.







Il existe différentes méthodes pour interpoler une image : Linéaire, Bilinéaire, Bicubique et Fractale. Toutes tentent d'augmenter la taille de l'image tout en limitant au maximum les effets d'un tel processus.

Généralement, le problème majeur avec l'interpolation est qu'elle peut augmenter fortement un défaut de l'image en l'agrandissant. L'interpolation Bicubique est habituellement considérée comme la meilleure méthode, mais pourtant, il arrive qu'elle ne produise pas les meilleurs résultats.

L'interpolation fonctionne le mieux lorsqu'elle est appliquée à petite dose et sur une grande quantité d'informations. Une image constituée d'une faible quantité de pixels montrera plus de trace de dégradation qu'une image plus riche au départ. Dans tous les cas, l'interpolation n'inventera pas des détails qui ne se trouvaient pas dans l'image originale.

4.2 Utilité de l'interpolation

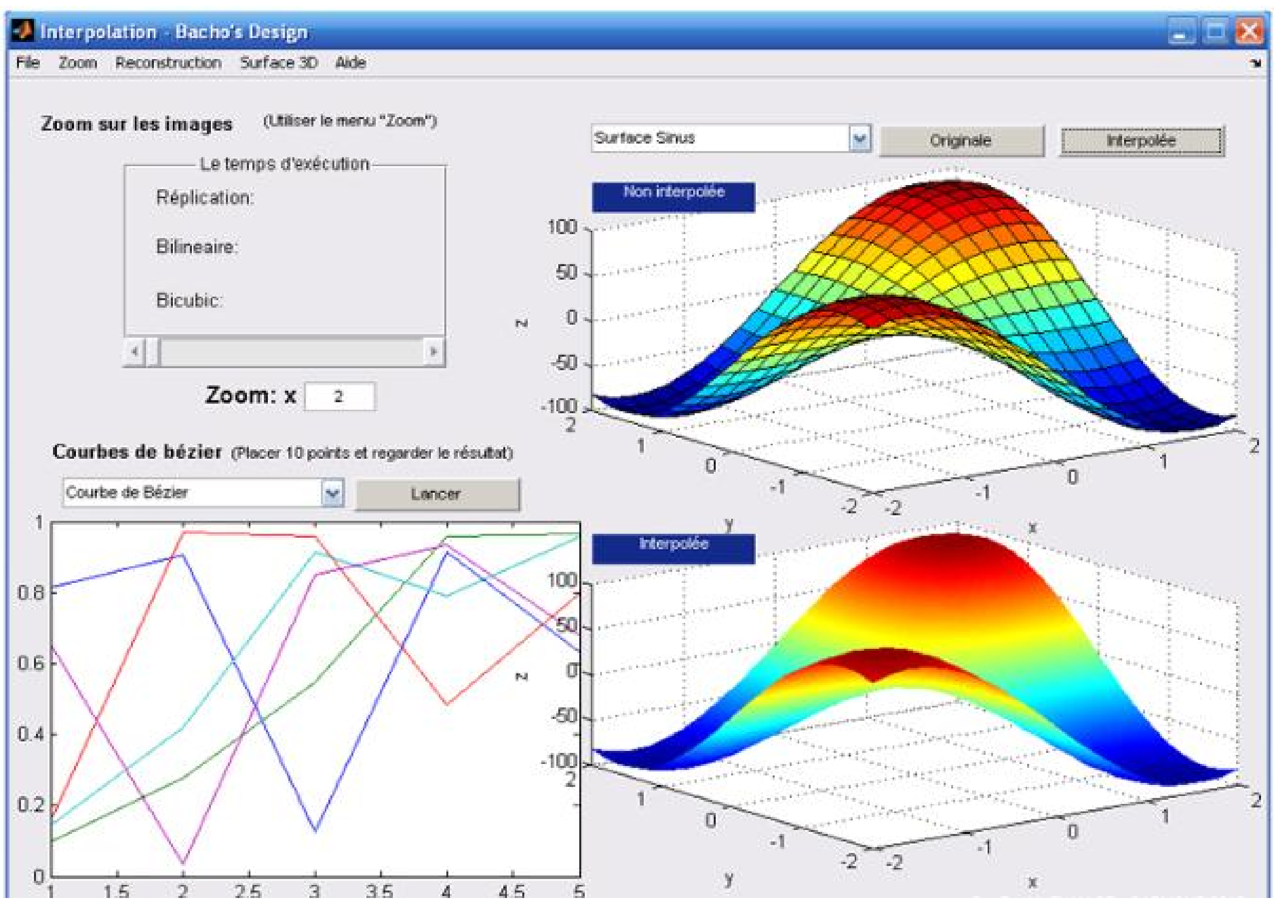
En terme informatique, l'interpolation est un procédé qui permet d'augmenter la résolution d'une image numérisée en définissant des pixels intermédiaires entre des pixels réels grâce à un algorithme mathématique, en calculant la moyenne des couleurs des pixels avoisinants. Cette technologie permet donc d'obtenir des résultats intéressants.

Toshiba a développé une nouvelle technique de super-résolution pour la création de haute résolution d'images ou vidéo à partir de ceux à faible résolution.

Si une partie de l'image est agrandie, la dégradation de la qualité d'image et l'image reste incertain, même si l'interpolation est effectuée en douceur entre les pixels.

4.3 Quelques résultats intéressants crées sous Matlab :

Avec une bonne connaissance de Matlab on arrive à créer son propre logiciel capable de faire un zoom sur les images en gardant une bonne qualité, de créer des surfaces en 3D coloriées en utilisant des fonctions mathématiques telles le sinus, le cosinus ou l'exponentielle. D'utiliser l'interpolation pour fusionner les couleurs,



Conclusion

Dans notre travail, nous avons rappelé l'interpolation 1D en la définissant comme étant une technique mathématique qui permet de décrire un ensemble de points à l'aide d'une fonction donnée. Nous avons également étudié ses propriétés.

Dans notre étude, nous avons parlé de l'interpolation bidimensionnelle, afin que l'interpolation peut être étendue à des fonctions de plusieurs variables. La procédure consiste en interpoler sur une variable, fixer les valeurs des autres variables puis combiner les résultats. Le polynôme d'interpolation a plusieurs méthodes qui conduisent au même résultat. Nous avons choisi dans notre travail la méthode de Lagrange et l'utilisation du Logisiel Matlab pour plus de précision.

Nous éspérons avoir donné ne serait-ce qu'une petite idée sur l'utilité de l'interpolation, par une application de l'interpolation Bilinéaire dans le contexte de la résolution d'image. Il existe différentes méthodes pour interpoler une image : Linéaire, Bilinéaire, Bicubique et Fractale. Toutes tentent d'augmenter la taille de l'image tout en limitant au maximum les effets d'un tel processus, pour cela nous espérons aussi avoir ouvert une brèche pour d'éventuels mémoires de Master.

Bibliographie

[1] Arkoub, N. (2012). Reconstruction de données manquantes dans une image (Doctoral dissertation, Université Mouloud Mammeri).

[2] CALVI, J. P. (2008). Analyse numérique.

[3] CHOPRA, S. C., & Canale, R. P. (2010). Numerical methods for engineers. Tata McGraw Hill, New York.

[4] KHELIL,N. (2020). Cours Analyse Numérique 1, Université de Biskra.

[5] KHELIL N. , DJEROU L. , Optimisation par Essaim de Particules contre le phénomne de Runge, ISBN-13 :978-613-1-5748-7, Maison d'edition : Editions universitaires europeenes. Site Web : https ://www.editions-ue.com/.

[6] RADI, B. & El Hami, A. (2010). Méthodes numériques pour l'ingénieur : utilisation de l'outil MATLAB : cours, exercices et problèmes de synthèse corrigés. Ellipses.