

**Université HADJ-LAKHDHAR - BATNA**

**Faculté de Technologie**

**Département de MECANIQUE**

**Licence - Génie Énergétique**

**Mémoire de Fin D'études**

**Présenté pour obtenir le diplôme de Licence**

**Résolution de l'équation de diffusion de la  
chaleur 2D instationnaire  
par la méthode des différences finies en utilisant  
le schéma explicite d'Euler  
avec le logiciel MAPLE**

**Présenté par:**

- Bezzouza Abderraouf**
- Hachachena Imed eddine**

**Encadré par:**

- Dr Laïd Messaoudi**

## Introduction

- En mathématiques et en physique théorique, l'équation de la chaleur est une équation aux dérivées partielles parabolique, pour décrire le phénomène physique de conduction thermique, introduite initialement en 1811 par Jean Baptiste Joseph Fourier, après des expériences sur la propagation de la chaleur, suivies par la modélisation de l'évolution de la température avec des séries trigonométriques, appelées depuis séries de Fourier et transformées de Fourier, permettant une grande amélioration à la modélisation mathématique des phénomènes, en particulier pour les fondements de la thermodynamique, et qui ont entraîné aussi des travaux mathématiques très importants pour les rendre rigoureuses, véritable révolution à la fois physique et mathématiques, sur plus d'un siècle.

### 1 - But

- L'objectif de notre travail est de déterminer la solution numérique de l'équation de la chaleur bidimensionnelle et instationnaire par le logiciel de calcul symbolique MAPLE et de la comparer avec la solution analytique existante obtenue par la méthode de séparation des variables. La discrétisation utilisée est basée sur la méthode des différences finies en utilisant le schéma explicite d'Euler.

- Le problème mathématique est défini comme suit:

Il s'agit de déterminer la répartition de la température à travers une plaque rectangulaire soumise aux conditions aux limites et initiale suivante:

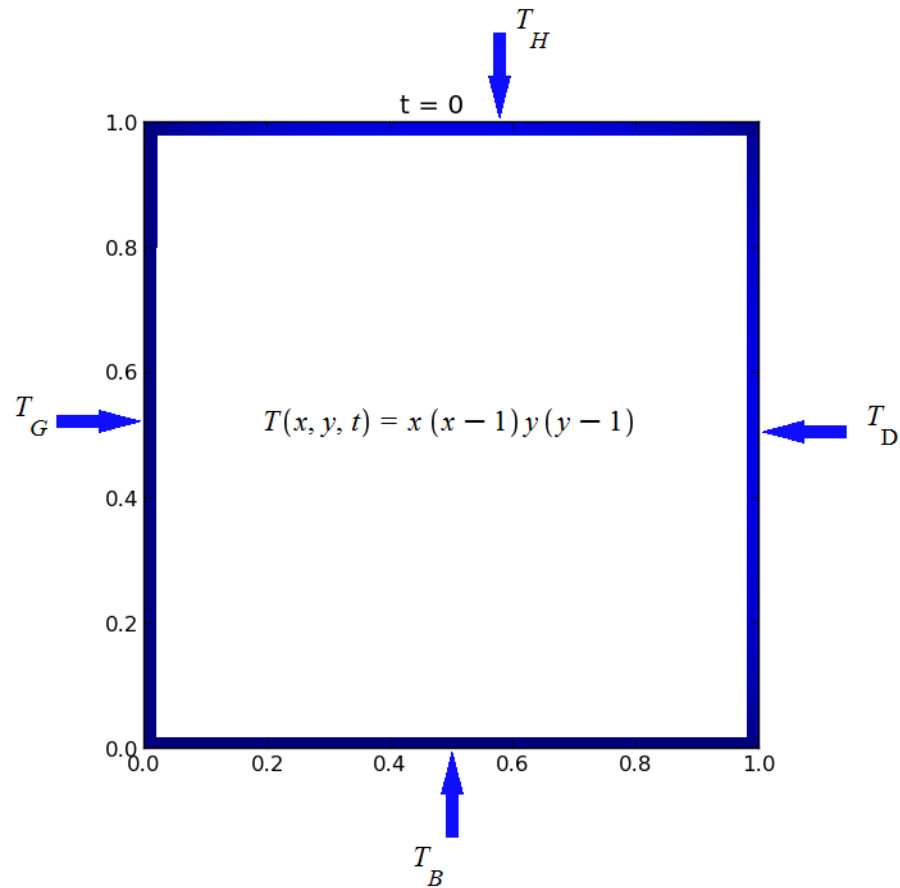
$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = \frac{\partial T}{\partial t} \cdot \frac{1}{\alpha}$$

conditions au limites:

$$\begin{array}{lll} T(0,y,t) = 0 & \text{et} & T(1,y,t) = 0 \\ T(x,0,t) = 0 & \text{et} & T(x,1,t) = 0 \end{array}$$

condition initiale:

$$T(x,y,0) = x(x-1)y(y-1)$$



## 1.1- Discrétisation

L'EDP modélisant l'équation de la chaleur est:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = \frac{\partial T}{\partial t} \cdot \frac{1}{\alpha} \quad \text{avec} \quad \alpha = \frac{k}{\rho \cdot Cp}$$

Sa discrétisation par le schéma explicite d'Euler nous donne :

$$\frac{T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n}{\Delta x^2} + \frac{T_{i,j-1}^n - 2T_{i,j}^n + T_{i,j+1}^n}{\Delta y^2} = \frac{1}{\alpha} \cdot \frac{T_{i,j}^{n+1} - 2T_{i,j}^n}{\Delta t} \quad \text{.....(1)}$$

on pose :

$$\lambda_1 = \frac{\Delta t}{\Delta x^2} \quad \text{et} \quad \lambda_2 = \frac{\Delta t}{\Delta y^2}$$

donc:

$$T_{i,j}^{n+1} = \lambda_1 \cdot \alpha \left[ T_{i-1,j}^n + T_{i+1,j}^n \right] + \lambda_2 \cdot \alpha \left[ T_{i,j-1}^n + T_{i,j+1}^n \right] + \left[ 1 - 2 \alpha \cdot (\lambda_1 + \lambda_2) \right] \cdot T_{i,j}^n \quad \text{.....(2)}$$

## 1.2 - Consistance

Les approximations des dérivées première et seconde sont:

$$\begin{aligned}\frac{\partial T}{\partial t} &\approx \frac{T^{n+1}_{i,j} - T^n_{i,j}}{\Delta t} + o(\Delta t) \\ \frac{\partial^2 T}{\partial x^2} &\approx \frac{T^n_{i-1,j} - 2T^n_{i,j} + T^n_{i+1,j}}{\Delta x^2} + o(\Delta x^2) \\ \frac{\partial^2 T}{\partial y^2} &\approx \frac{T^n_{i,j-1} - 2T^n_{i,j} + T^n_{i,j+1}}{\Delta y^2} + o(\Delta y^2) \\ f &= \frac{\partial T}{\partial t} - \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right)\end{aligned}$$

appliquons les opérateurs discret D et continu L à l'équation  $f$  :

$$|Df - \mathcal{L}f| = |o(\Delta t) + o(\Delta x)^2 + o(\Delta y)^2|$$

$$\text{si } o(\Delta t) \sim 0 \text{ et } o(\Delta x)^2 \sim 0 \text{ et } o(\Delta y)^2 \sim 0 \text{ alors } |o(\Delta t) + o(\Delta x)^2 + o(\Delta y)^2| \sim 0$$

Le schéma est donc consistant avec l'EDP.

## 1.3 - Stabilité

Utilisons l'analyse de Von-Neumann :

on suppose que le solution est de la forme :  $T(x, y, t) = T^n_{i,j} = \psi(t) e^{jB(x+y)}$

on a alors :

$$\begin{aligned}T^{n+1}_{i,j} &= \psi(t + \Delta t) e^{jB(x+y)} \\ T^n_{i-1,j} &= \psi(t) e^{jB(x-\Delta x+y)} \\ T^n_{i+1,j} &= \psi(t) e^{jB(x+\Delta x+y)}\end{aligned}$$

$$T^n_{i,j-1} = \psi(t) \, e^{jB(x+y-\Delta y)}$$

$$T^n_{i,j+1} = \psi(t) \, e^{jB(x+y+\Delta y)}$$

en remplaçant ces termes dans l'équation (2) et en simplifiant par  $e^{jB(x+y)}$

$$\frac{\psi(t + \Delta t)}{\psi(t)} = \lambda_1 \cdot \alpha [e^{jB\Delta x} + e^{-jB\Delta x}] + \lambda_2 \cdot \alpha [e^{jB\Delta y} + e^{-jB\Delta y}] + 1 - 2 \cdot \alpha (\lambda_1 + \lambda_2)$$

Le facteur d'amplification:

On a  $\xi = \frac{\psi(t + \Delta t)}{\psi(t)}$  donc:

$$\xi = \alpha \cdot \lambda_1 [2 \cos(\beta \Delta x)] + \alpha \cdot \lambda_2 [2 \cos(\beta \Delta y)] + 1 - 2 \cdot \alpha (\lambda_1 + \lambda_2) \geq -1$$

$$|\xi| < 1 \quad \Rightarrow \quad -1 < \xi < 1$$

$$\alpha \cdot \lambda_1 [2 \cos(\beta \Delta x)] + \alpha \cdot \lambda_2 [2 \cos(\beta \Delta y)] + 1 - 2 \cdot \alpha (\lambda_1 + \lambda_2) \geq -1$$

$$\alpha \cdot \lambda_1 [2 \cos(\beta \Delta x)] + \alpha \cdot \lambda_2 [2 \cos(\beta \Delta y)] + 1 - 2 \cdot \alpha (\lambda_1 + \lambda_2) \leq 1$$

donc la condition de stabilité de ce schéma  $\Rightarrow (\lambda_1 + \lambda_2) \cdot \alpha \neq \frac{1}{2}$

## 1.4 - Convergence

Puisque ce schéma est consistant et stable alors il est convergent.

## 2 - Programation

### 2.1 - Solution Analytique

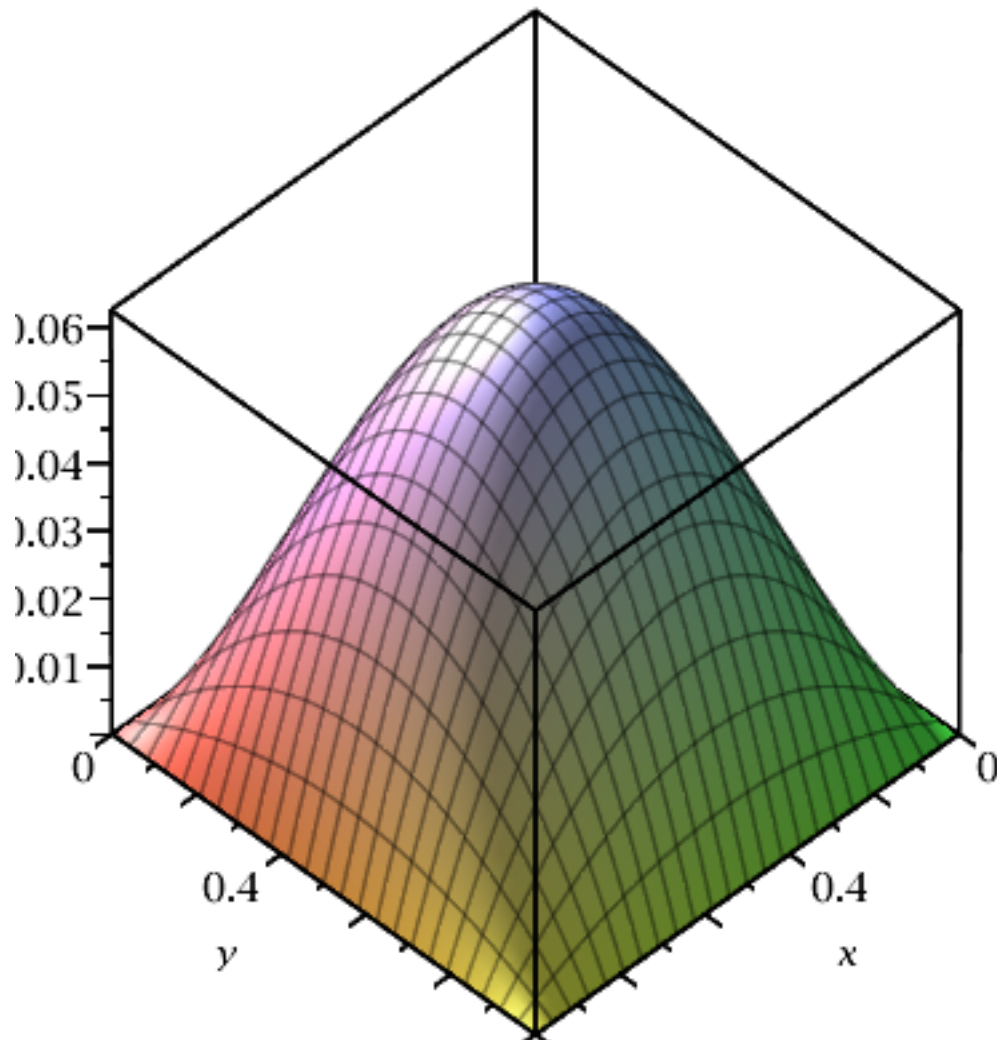
- La solution analytique obtenue par la méthode de séparation des variables est donnée par l'équation:



$$T(x, y, t) = \sum_{k=0}^{30} \left( \sum_{l=0}^{30} \frac{64 \cdot e^{-\frac{1}{80}((2 \cdot k + 1)^2 \cdot \pi^2 + (2 \cdot l + 1)^2 \cdot \pi^2)t} \cdot \sin((2 \cdot k + 1) \cdot \pi \cdot x) \cdot \sin((2 \cdot l + 1) \cdot \pi \cdot y)}{(2 \cdot l + 1)^3 \cdot \pi^6 \cdot (2 \cdot k + 1)^3} \right)$$

## Tracé graphique:

```
> restart, with(plots) :  
> T := sum_{k=0}^{30} \left( \sum_{l=0}^{30} \frac{64 \cdot e^{-\frac{1}{80}((2 \cdot k + 1)^2 \pi^2 + (2 \cdot l + 1)^2 \pi^2)t} \cdot \sin((2 \cdot k + 1) \cdot \pi \cdot x) \cdot \sin((2 \cdot l + 1) \cdot \pi \cdot y)}{(2 \cdot l + 1)^3 \cdot \pi^6 \cdot (2 \cdot k + 1)^3} \right) :  
> animate3d(T, x = 0 .. 1, y = 0 .. 1, t = 0 .. 10, frames = 50)
```



## 2.2 - Solution Numérique

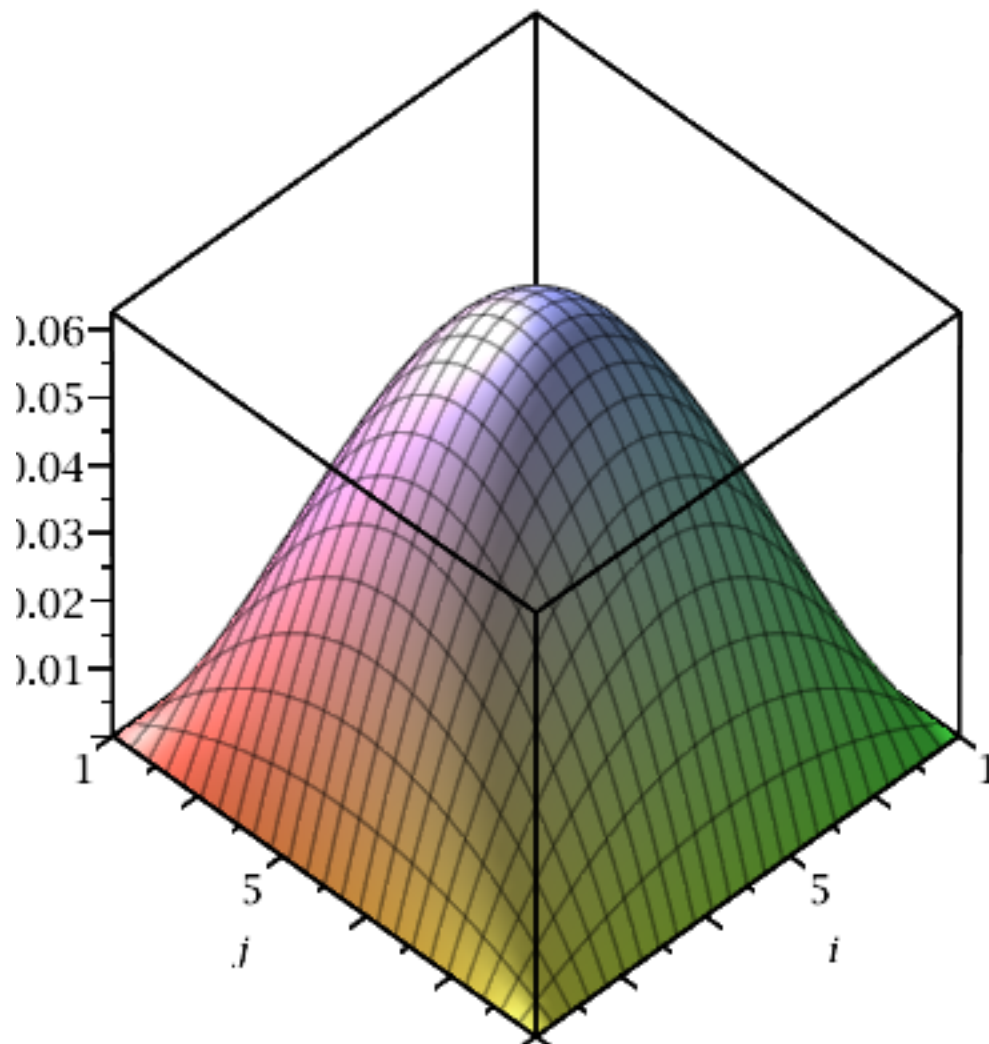
### 2.2.1 - Forme standard

### Initialisation:

```
> restart :  
> with(plots) :
```

### Données:

```
>  $\Delta t := 0.05$       # Pas temporel en s  
                                 $\Delta t := 0.05$   
=>  
>  $\Delta x := 0.1$       # Pas spacial selon x en m  
                                 $\Delta x := 0.1$   
=>  
>  $\Delta y := 0.1$       # Pas spacial selon y en m  
                                 $\Delta y := 0.1$   
=>  
>  $nmax := 160$       # Nombre de pas temporels  
                                 $nmax := 160$   
=>  
>  $imax := 11$       # Nombre de noeuds selon x  
                                 $imax := 11$   
=>  
>  $jmax := 11$       # Nombre de noeuds selon y  
                                 $jmax := 11$   
=>  
>  $Tb := 0$           # Condition limite en bas  
                                 $Tb := 0$   
=>  
>  $Th := 0$           # Condition à limite en haut  
                                 $Th := 0$   
=>  
>  $Tg := 0$           # Condition à limite à gauche  
                                 $Tg := 0$   
=>  
>  $Td := 0$           # Condition à limite à droite  
                                 $Td := 0$   
=>  
>  $\sigma := (\Delta x \cdot (i - 1) - 1) \cdot (\Delta x \cdot (i - 1)) \cdot (\Delta y \cdot (j - 1) - 1) \cdot (\Delta y \cdot (j - 1))$   
                                # Condition initiale  
                                 $\sigma := 0.01 (0.1 i - 1.1) (i - 1) (0.1 j - 1.1) (j - 1)$   
=>  
> plot3d( $\sigma$ ,  $i = 1 .. imax$ ,  $j = 1 .. jmax$ , axes = boxed)
```



Calculs:

>  $\lambda 1 := \frac{\Delta t}{\Delta x^2}$

$\lambda 1 := 5.000000000$

>  $\lambda 2 := \frac{\Delta t}{\Delta y^2}$

$\lambda 2 := 5.000000000$

>  $\alpha := \frac{1}{80}$

$\alpha := \frac{1}{80}$

>  $(\lambda 1 + \lambda 2) \cdot \alpha$  # *Condition de stabilité*  
0.1250000000

### Conditins aux limites:

```
> for n from 0 to nmax do;  
    for i from 1 to imax do;  
         $T[i, 1, n] := Tb$   
    end do;  
end do;  
[  
    > for n from 0 to nmax do  
        for i from 1 to imax do;  
             $T[i, jmax, n] := Th$   
        end do;  
    end do;  
    > for n from 0 to nmax do  
        for j from 1 to jmax do;  
             $T[1, j, n] := Tg$   
        end do;  
    end do;  
    > for n from 0 to nmax do;  
        for j from 1 to jmax do;  
             $T[imax, j, n] := Td$   
        end do;  
    end do;  
]
```

### Conditions initiales:

```
> for i from 2 to imax - 1 do;  
    for j from 2 to jmax - 1 do;  
         $T[i, j, 0] := \sigma$   
    end do;  
end do;
```

### Boucle principale:

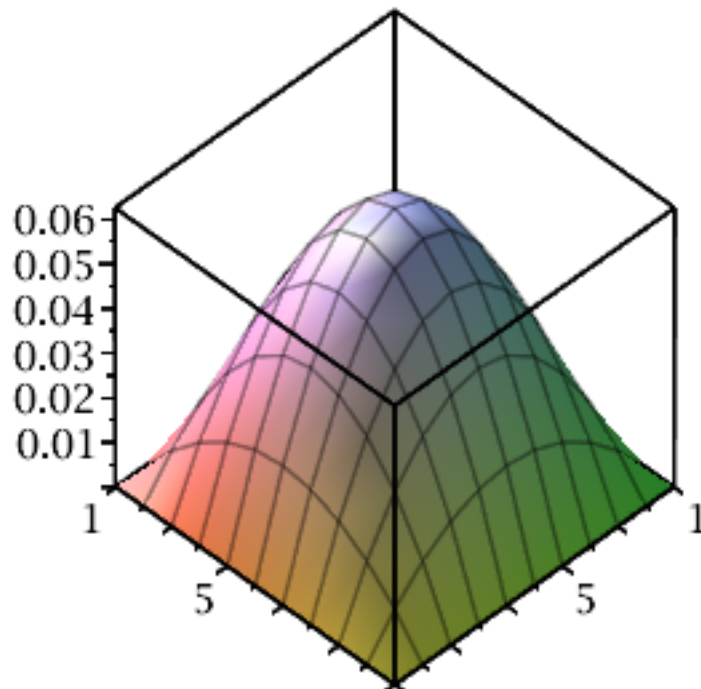
```
> for n from 0 to nmax - 1 do;  
    for i from 2 to imax - 1 do;  
        for j from 2 to jmax - 1 do;  
             $T[i, j, n + 1] := \lambda 1 \cdot \alpha \cdot (T[i - 1, j, n] + T[i + 1, j, n]) + \lambda 2 \cdot \alpha \cdot (T[i, j - 1, n] + T[i, j + 1, n]) + (1 - 2 \cdot \alpha \cdot (\lambda 1 + \lambda 2)) \cdot T[i, j, n]$   
        end do;  
    end do;  
end do;
```

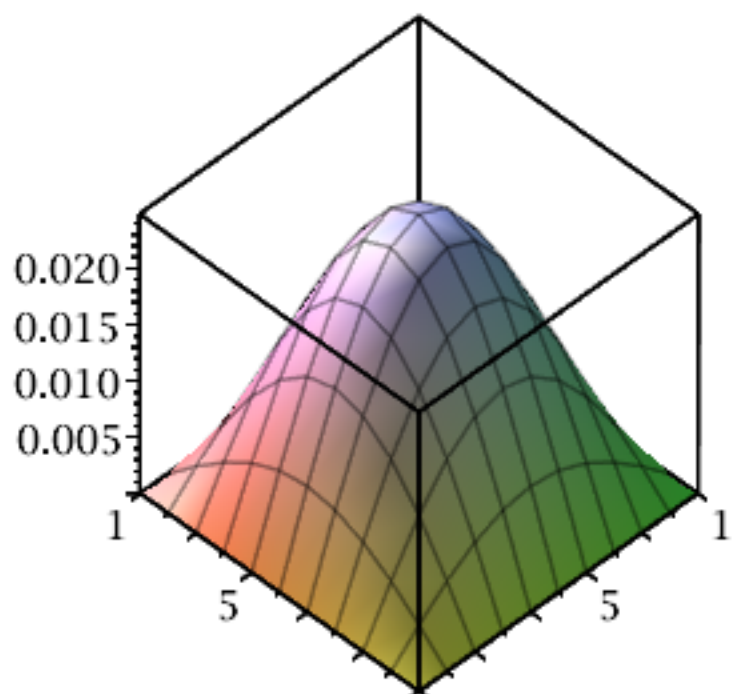
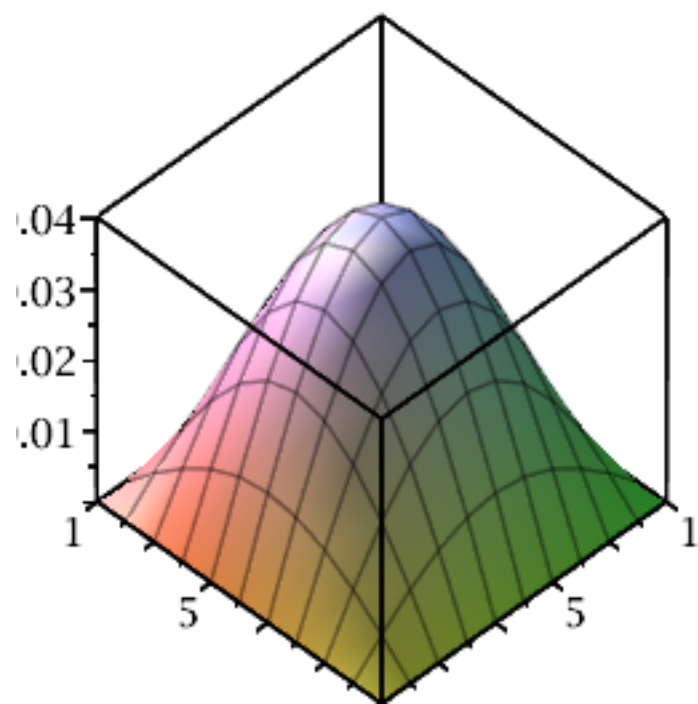
### Création des listes pour le tracé graphique:

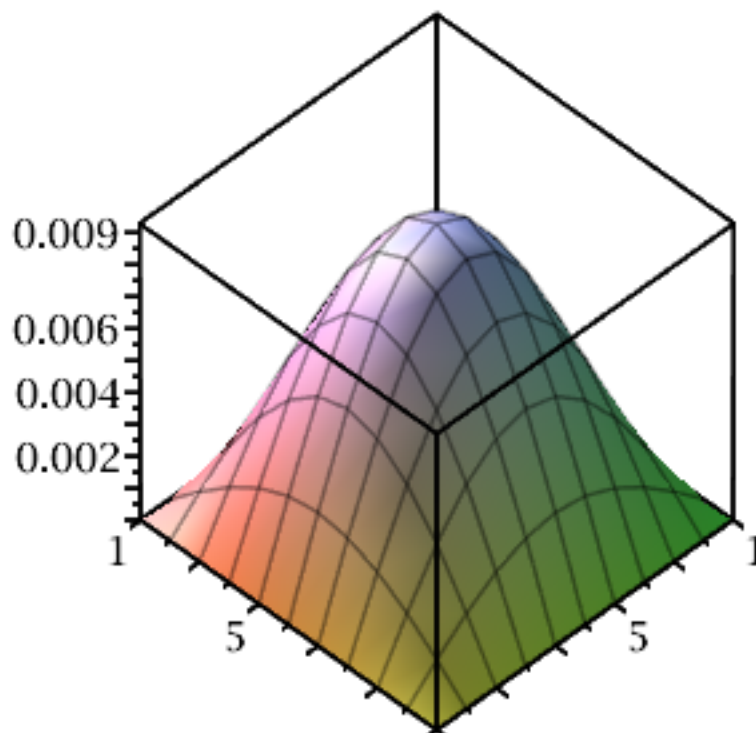
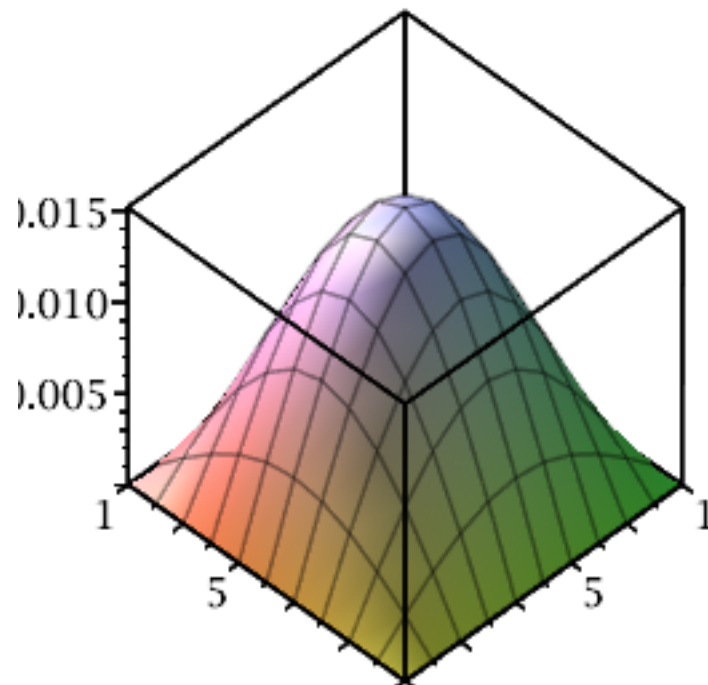
```

> for n from 0 to nmax do;
    L[n] := [ (seq( [ seq( T[i, j, n], i = 1 ..imax) ], j = 1 ..jmax) ) ]
end do:
> for n from 0 to nmax do
    Q[n] := listplot3d(L[n], axes = boxed)
end do:
> for n from 0 to nmax by  $\frac{nmax}{4}$  do
    Q[n]
end do

```







### 2.2.2 - Forme procédurale

```
lc:=proc( $\Delta t, \Delta x, \Delta y, nmax, imax, jmax, \alpha, Tb, Th, Tg, Td$ )
  local  $\lambda 1, \lambda 2, n, i, T, j, L, Q$ ;
```



```

 $\lambda 1 := \frac{\Delta t}{\Delta x^2};$ 

 $\lambda 2 := \frac{\Delta t}{\Delta y^2};$ 

for  $n$  from 0 to  $nmax$  do
  for  $i$  to  $imax$  do
     $T[i, 1, n] := Tb$ 
  end do
end do;
for  $n$  from 0 to  $nmax$  do
  for  $i$  to  $imax$  do
     $T[i, jmax, n] := Th$ 
  end do
end do;
for  $n$  from 0 to  $nmax$  do
  for  $j$  to  $jmax$  do
     $T[1, j, n] := Tg$ 
  end do
end do;
for  $n$  from 0 to  $nmax$  do
  for  $j$  to  $jmax$  do
     $T[imax, j, n] := Td$ 
  end do
end do;
for  $i$  from 2 to  $imax - 1$  do
  for  $j$  from 2 to  $jmax - 1$  do
     $T[i, j, 0] := (\Delta x \cdot (i - 1) - 1) \cdot \Delta x \cdot (i - 1) \cdot (\Delta y \cdot (j - 1) - 1) \cdot \Delta y \cdot (j - 1)$ 
  end do
end do;
for  $n$  from 0 to  $nmax - 1$  do
  for  $i$  from 2 to  $imax - 1$  do
    for  $j$  from 2 to  $jmax - 1$  do
 $T[i, j, n + 1] := \lambda 1 \cdot \alpha \cdot (T[i - 1, j, n] + T[i + 1, j, n]) + \lambda 2 \cdot \alpha \cdot (T[i, j - 1, n]$ 
     $+ T[i, j + 1, n]) + (1 - 2 \cdot \alpha \cdot (\lambda 1 + \lambda 2)) \cdot T[i, j, n]$ 
    end do
  end do
end do;
for  $n$  from 0 to  $nmax$  do
   $L[n] := [seq([seq(T[i, j, n], i = 1 .. imax)], j = 1 .. jmax)]$ 

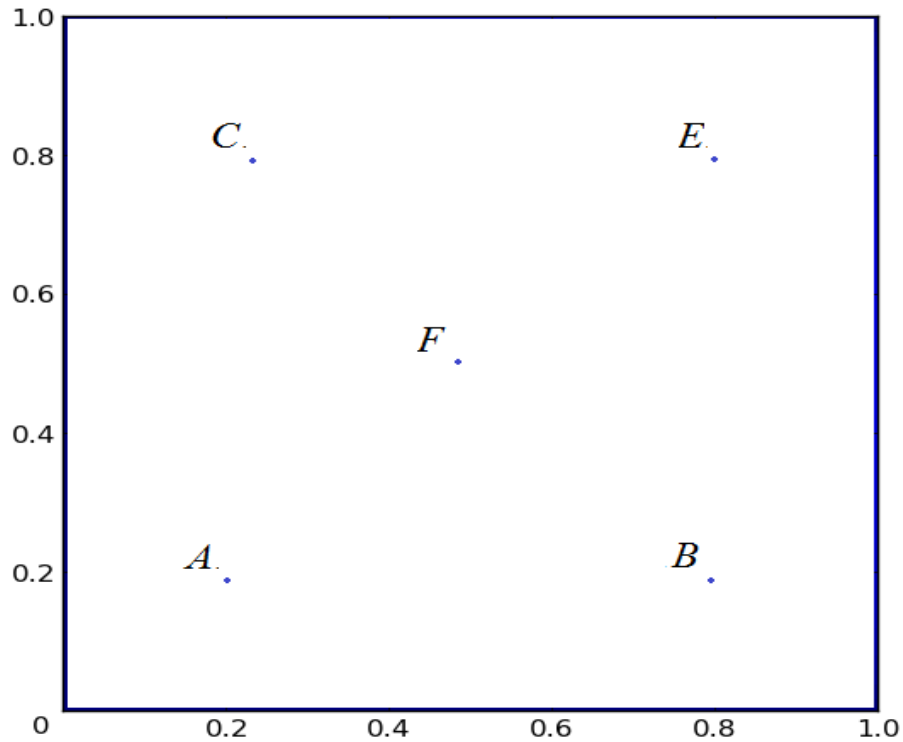
```

```
end do;  
  for  $n$  from 0 to  $nmax$  do  
     $Q[n] := listplot3d(L[n])$   
  end do;  
  for  $n$  from 0 to  $nmax$  do  
     $Q[n]$   
  end do;  
end proc
```

### 3 - Comparaison

- Nous avons choisi arbitrairement cinq points pour voir la différence entre la solution analytique et la solution numérique en cinq temps différents.

La position de ces cinq points est indiquée dans la figure ci-dessous:



```
> with(geometry) : with(plots) :
  t := n * Δt :
  p := trunc( (nmax) / 4 ) :
  h := trunc( (nmax * Δt) / 4 ) :
```

Nous utilisons des boucles variant avec le temps pour calculer

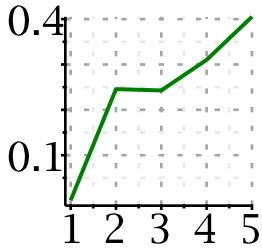
$T_{AN}[t]$ ,  $T_{Num}[n]$ ,  $\epsilon \left[ \text{trunc} \left( \frac{t}{\Delta t} \right) \right]$  en utilisant les équations suivantes :

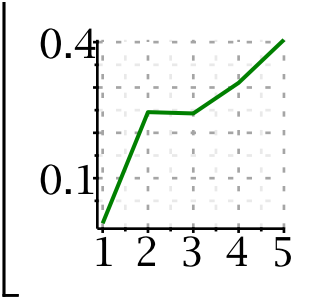
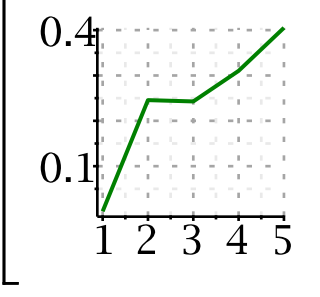
•  $T_{AN}[t] :=$

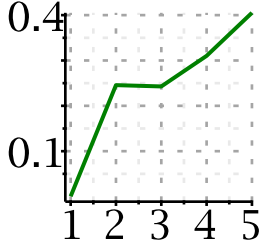
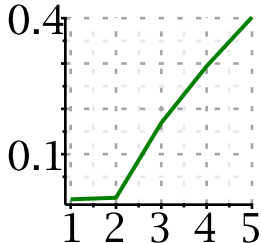
$$\sum_{k=0}^{30} \left( \sum_{l=0}^{30} \frac{64 \cdot e^{-\frac{1}{80}((2 \cdot k + 1)^2 \cdot \pi^2 + (2 \cdot l + 1)^2 \cdot \pi^2)t}} \cdot \sin((2 \cdot k + 1) \cdot \pi \cdot x) \cdot \sin((2 \cdot l + 1) \cdot \pi \cdot y)}{(2 \cdot l + 1)^3 \cdot \pi^6 \cdot (2 \cdot k + 1)^3} \right)$$

•  $T_{Num}[n] := T[i, j, n]$

$$\bullet \, \boldsymbol{\varepsilon} \left[ \text{trunc} \left( \frac{t}{\Delta t} \right) \right] := \left| \frac{T_{AN}[t] - T_{Num} \left[ \text{trunc} \left( \frac{t}{\Delta t} \right) \right]}{T_{AN}[t]} \right| \cdot 100$$

Point de comparaison	Solution Analytique	Solution Numérique	Erreur Relative (%)
$A_{analytique} := [0.2, 0.2]$ $A_{numerique} := [3, 3]$	$T_{AN_0} := 0.02559989119$ $T_{AN_2} := 0.01427588457$ $T_{AN_4} := 0.0085919248 \backslash 08$ $T_{AN_6} := 0.0052349243 \backslash 64$ $T_{AN_8} := 0.0031950270 \backslash 78$	$T_{Num_0} := 0.0256$ $T_{Num_{40}} := 0.01431095522$ $T_{Num_{80}} := 0.0086127818 \backslash 85$ $T_{Num_{120}} := 0.0052511636 \backslash 52$ $T_{Num_{160}} := 0.0032079715 \backslash 94$	$\epsilon_0 := 0.0004250408 \backslash 691$ $\epsilon_{40} := 0.2456635862$ $\epsilon_{80} := 0.2427520895$ $\epsilon_{120} := 0.3102105565$ $\epsilon_{160} := 0.4051457369$ 
$B_{analytique} := [0.8, 0.2]$ $B_{numerique} := [9, 3]$	$T_{AN_0} := 0.02559989116$ $T_{AN_2} := 0.01427588456$ $T_{AN_4} := 0.0085919248 \backslash 08$ $T_{AN_6} := 0.0052349243 \backslash 64$ $T_{AN_8} := 0.0031950270 \backslash$	$T_{Num_0} := 0.0256$ $T_{Num_{40}} := 0.01431095522$ $T_{Num_{80}} := 0.0086127818 \backslash 85$ $T_{Num_{120}} := 0.0052511636 \backslash 52$ $T_{Num_{160}} := 0.0032079715 \backslash 94$	$\epsilon_0 := 0.0004251580 \backslash 576$ $\epsilon_{40} := 0.2456636564$ $\epsilon_{80} := 0.2427520895$ $\epsilon_{120} := 0.3102105565$ $\epsilon_{160} := 0.4051457998$

	76		
$C_{analytique} := [0.2, 0.8]$ $C_{numerique} := [3, 9]$	$T_{AN_0} := 0.02559989119$ $T_{AN_2} := 0.01427588457$ $T_{AN_4} := 0.0085919248 \backslash 08$ $T_{AN_6} := 0.0052349243 \backslash 64$ $T_{AN_8} := 0.0031950270 \backslash 78$	$T_{Num_0} := 0.0256$ $T_{Num_{40}} := 0.01431095522$ $T_{Num_{80}} := 0.0086127818 \backslash 85$ $T_{Num_{120}} := 0.0052511636 \backslash 52$ $T_{Num_{160}} := 0.0032079715 \backslash 94$	$\epsilon_0 := 0.0004250408 \backslash 691$ $\epsilon_{40} := 0.2456635862$ $\epsilon_{80} := 0.2427520895$ $\epsilon_{120} := 0.3102105565$ $\epsilon_{160} := 0.4051457369$ 
$E_{analytique} := [0.8, 0.8]$ $E_{numerique} := [9, 9]$	$T_{AN_0} := 0.02559989116$ $T_{AN_2} := 0.01427588456$ $T_{AN_4} := 0.0085919248 \backslash 08$ $T_{AN_6} := 0.0052349243 \backslash 64$	$T_{Num_0} := 0.0256$ $T_{Num_{40}} := 0.01431095522$ $T_{Num_{80}} := 0.0086127818 \backslash 85$ $T_{Num_{120}} := 0.0052511636 \backslash 52$	$\epsilon_0 := 0.0004251580 \backslash 576$ $\epsilon_{40} := 0.2456636564$ $\epsilon_{80} := 0.2427520895$ $\epsilon_{120} := 0.3102105565$ $\epsilon_{160} :=$

	$T_{AN_8} :=$ $0.0031950270 \backslash$ $76$	$T_{Num_{160}} :=$ $0.0032079715 \backslash$ $94$	$0.4051457998$ 
$Z_{analytique} := [0.5,$ $0.5]$ $Z_{numerique} := [6, 6]$	$T_{AN_0} :=$ $0.06250027017$ $T_{AN_2} :=$ $0.04022568084$ $T_{AN_4} :=$ $0.02477583675$ $T_{AN_6} :=$ $0.01514423333$ $T_{AN_8} :=$ $0.0092471054 \backslash$ $97$	$T_{Num_0} := 0.0625$ $T_{Num_{40}} :=$ $0.04022725210$ $T_{Num_{80}} :=$ $0.02481774772$ $T_{Num_{120}} :=$ $0.01518875618$ $T_{Num_{160}} :=$ $0.0092842770 \backslash$ $16$	$\varepsilon_0 :=$ $0.0004322701 \backslash$ $314$ $\varepsilon_{40} :=$ $0.0039061116 \backslash$ $36$ $\varepsilon_{80} :=$ $0.1691606642$ $\varepsilon_{120} :=$ $0.2939921027$ $\varepsilon_{160} :=$ $0.4019800468$ 

## **4 - Conclusion**

Durant notre cursus, nous avons appris à discrétiser l'équation de la chaleur unidimensionnelle et instationnaire. Ce projet nous a donné l'opportunité d'attaquer le problème bidimensionnel. Après avoir discrétisé les équations du problème continu, nous avons étudié théoriquement sa consistance et sa stabilité avant de passer à sa programmation.

Nous avons utilisé le logiciel Maple pour programmer ce problème de deux manières: classique et procédurale. Ce logiciel nous a aussi permis de faire tous les tracés graphiques nécessaires.

En fin, nous avons comparé nos résultats avec ceux données par la solution analytique issue de la bibliographie. Cette comparaison est faite en utilisant des procédures de calcul de l'erreur relative intégrées directement dans un tableau afin de mieux voir les différences. D'une manière générale, les résultats numériques sont très proches de ceux calculés analytiquement vu que l'erreur relative maximale est très faible ( $< 0.5 \%$ ).