EMMN (4)

Mise en oeuvre numérique d'un schéma explicite pour l'équation de la chaleur

• Discrétisation

* Nous nous intéressons au modèle mono-dimensionnel pour l'évolution d'un champ $u(x,t)$ [la température typiquement] :

(1) $$\frac{\partial u}{\partial t} - \mu \frac{\partial^2 u}{\partial x^2} = 0, \quad 0 < x < L, \quad t > 0 \, .$$

L'équation aux dérivées partielles (1) est associée à une condition limite homogène de Dirichlet en $x=0$ et $x=L$:

(2) $$u(0,t) = u(L,t) = 0 \, , \quad t > 0$$

ainsi qu'à une condition initiale $(t=0)$:

(3) $$u(x,0) = u_0(x) \, , \quad 0 \leq x \leq L \, .$$

* Nous avons vu qu'après discrétisation en espace à l'aide d'un pas d'espace $\Delta x > 0$ et d'une variable discrète u_j telle que

(4) $$u_j \simeq u(j\Delta x) \, , \quad 0 \leq j \leq J$$

où $J \Delta x = L$

la dérivée seconde en espace est bien approchée à l'aide d'un schéma dit "à trois points" :

(5) $$\left(\frac{\partial^2 u}{\partial x^2}\right)(j\Delta x) \simeq \frac{1}{\Delta x^2}\left(u_{j+1} - 2u_j + u_{j-1}\right).$$

On peut donc remplacer l'équation (1) par un système de $(J-1)$ équations différentielles couplées, d'inconnues $u_j(t)$, $0 \leq j \leq J$ et $t > 0$ temps continu :

(6) $$\frac{d}{dt}u_j(t) - \frac{\mu}{\Delta x^2}\left(u_{j+1}(t) - 2u_j(t) + u_{j-1}(t)\right) = 0, \quad 1 \leq j \leq J-1, \; t > 0$$

avec les conditions aux limites en $x = 0$ $(j = 0)$ et $x = L$ $(j = J)$:

(7) $$u_0(t) = u_J(t) = 0, \quad t \geq 0,$$

et les conditions initiales

(8) $$u_j(t=0) = u_j^0, \quad 0 \leq j \leq J$$

* Pour résoudre l'équation différentielle (1), on procède comme au chapitre 1. On introduit un pas de temps $\Delta t > 0$, une valeur approchée u_j^n à l'instant $n\Delta t$:

(9) $$u_j^n \simeq u_j(n\Delta t), \quad n \text{ entier} \geq 0,$$

et on remplace d'une part $\frac{du_j}{dt}$ par l'approximation $(u_j^{n+1} - u_j^n)/\Delta t$ et d'autre part $(u_{j+1} - 2u_j + u_{j-1})$ par son approximation $(u_{j+1}^n - 2u_j^n + u_{j-1}^n)$ à l'instant $n\Delta t$; on obtient ainsi :

$$(10) \quad \left\{ \begin{array}{l} \frac{1}{\Delta t}(u_j^{n+1} - u_j^n) - \frac{\mu}{\Delta x^2}(u_{j+1}^n - 2u_j^n + u_{j-1}^n) = 0 \,, \\ \qquad\qquad\qquad\qquad 1 \leq j \leq J-1 \,, \; n \geq 0 \end{array} \right.$$

qu'il convient de joindre à la condition limite

$$(11) \quad u_0^n = u_J^n = 0 \,, \quad n \geq 0$$

et la condition initiale (8).

* Si on connait la famille $(u_j^n)_{0 \leq j \leq J}$ à l'instant $n\Delta t$, on peut sans difficulter la déterminer à l'instant ultérieur $(n+1)\Delta t$. On a d'une part la condition limite en $j=0$ et $j=J$:

$$(12) \quad u_0^{n+1} = u_J^{n+1} = 0$$

et d'autre part la résolution de l'équation (10) avec pour inconnue u_j^{n+1}. On pose

$$(13) \quad \sigma = \mu \frac{\Delta t}{\Delta x^2}$$

et on a

$$(14)\quad u_j^{n+1} = u_j^n + \rho\left(u_{j+1}^n - 2u_j^n + u_{j-1}^n\right), \quad 1 \leq j \leq J-1$$

- Mise en oeuvre informatique.

La programmation, la mise en oeuvre, l'implémentation du schéma (13)(14) est facile car la relation (14) est une "formule explicite" de calcul des nouvelles valeurs à l'instant $(n+1)$ en fonction des anciennes à l'instant n. Afin de maîtriser la place mémoire, on utilise deux vecteurs U et V de longueur $J+1$, avec des indices entre 0 et J dans ces notes de cours, même si "octave" impose une numérotation qui démarre à $j=1$.

* On doit d'abord initialiser U qui contient alors le vecteur $u_0(j\Delta x)$ discrétisé en espace. On a donc

boucle sur j : 0 à J

$U(j) = u_0(j\Delta x)$

fin de boucle.

* Une itération de l'algorithme permet d'abord de calculer les nouvelles valeurs à l'instant $(n+1)$, qu'on stocke dans un vecteur V :

```
V(0) = 0
V(J) = 0
boucle sur j : 1 à J-1
V(j) = U(j) + S(U(j+1) - 2*U(j)
                + U(j-1))
fin boucle.
```

(Attention aux indices !)

On procède alors à un "passage du temps discret" en remplaçant le contenu du vecteur U par celui du vecteur V qu'on vient de calculer :

```
boucle sur j : 0 à J
U(j) = V(j)
fin de boucle.
```

A l'issue de cette étape le vecteur U contient les variables $(u_j^{n+1})_{0 \leq j \leq J}$ et le temps s'est incrémenté de la valeur n à la valeur $(n+1)$.

* Il ne reste plus qu'à dessiner le nouveau champ à l'écran et mettre une boucle en temps autour des deux étapes précédentes.

On obtient ainsi l'algorithme global qui suit

% Initialisation

boucle sur j : 0 à J
$U(j) = u_0(j\Delta x)$
fin de boucle

% Boucle en temps
boucle sur n : 1 à N $\left\{ N : \text{nombre de pas de temps} \right.$

$V(0) = 0$, $V(J) = 0$

% Boucle en espace
boucle sur j : 1 à $J-1$
$V(j) = U(j) + \rho\,(U(j+1) - 2U(j) + U(j-1))$
fin boucle
boucle sur j : 0 à J
$U(j) = V(j)$
fin de boucle
dessin du graphe $(X(j), U(j))$
fin de boucle % en temps.

Le programme en "Octave" est voisin de ce qui est écrit sur cette page, mais pas identique. attention !

Dubois mars 07.