

Conduction thermique



Enseignant responsable
Bernard GLEYSE

Étudiants :
Pierre CHARDONNAL
Floriane GLOZ
Mathieu MEDICI

Quentin EGLY
Katie LEBLOND

Date de remise du rapport : 15/06/2014

Référence du projet : Rapport P6 2014 3

Intitulé du projet : Conduction thermique

Type de projet :

Modélisation

Expérience

Objectifs du projet :

L'objectif de ce projet est de mettre en œuvre la modélisation d'un problème, la résolution de l'équation de la chaleur, de manière analytique et implémenter ce modèle grâce à l'informatique. De plus, nous devons réaliser une expérience dans le but de comparer les résultats théoriques et expérimentaux. Ainsi, ce projet permet d'allier les mathématiques, la physique avec l'informatique mais aussi la théorie avec la pratique.

Mots clés du projet :

Equation de la chaleur

Modélisation et implémentation

Expérience

Table des matières

Introduction	4
1 Méthodologie, organisation du travail	5
2 Travail réalisé et résultats	6
2.1 Etablissement de l'équation de la chaleur	6
2.2 Résolution de l'équation de la chaleur en 1D	7
2.2.1 Méthode explicite	7
2.2.2 Méthode implicite	8
2.3 Implémentation informatique	10
2.4 Expérience	11
Conclusion et perspectives	12
Expérience	12
Bibliographie	13
A Code du programme de résolution de l'équation de la chaleur en explicite et en implicite	14
B Graphique	18

Introduction

Dans le cadre de notre 2^{ème} année de cycle STPI au sein de l'INSA de Rouen, nous avons effectué un projet physique qui s'est déroulé sur le temps du semestre par groupe de 5 personnes. Notre sujet est la conduction thermique.

Le transfert d'énergie thermique ou transfert de chaleur est une transmission de cette énergie d'une région à une autre, sous l'influence d'une différence de température. Dans le cas contraire, le milieu est en équilibre thermique et aucun transfert de chaleur ne peut se produire. Il existe 3 modes classiques de transmission : la conduction, le rayonnement et la convection. Ici, nous étudions la conduction.

Dans le mode conductif, la chaleur se diffuse de proche en proche d'une particule à l'autre par chocs ; ce mode nécessite donc la présence de matière, mais sans qu'il y ait de déplacement apparent de cette dernière. Dans les corps solides totalement opaques, c'est le seul mode de transmission. Ainsi dans l'étude de la diffusion de la chaleur à travers une barre, seule la conduction thermique intervient. À l'échelle microscopique, le problème conductif est très complexe et nous ne l'envisagerons pas ici. Nous nous plaçons dans l'hypothèse des milieux continus.

Son étude fait partie d'une discipline appelée thermique, qui étudie tous ces transferts. Elle intervient lors de la conception de systèmes de chauffage, d'isolement ou de refroidissement. Cependant cette étude est très vaste, ainsi nous n'en avons traité qu'une partie.

Nous avons, à travers ce rapport, combiné à la fois la physique aux mathématiques et à l'informatique. En effet, nous avons établi l'équation de la chaleur d'un point de vue de la physique, puis résolu cette dernière en 1D dans les schémas explicite et implicite analytiquement et informatiquement. La partie analytique permet de justifier les calculs effectués par l'ordinateur. Et enfin, nous avons aussi réalisé une expérience en laboratoire afin de pouvoir comparer les valeurs théoriques et les valeurs expérimentales.

Ainsi, ce projet nous incite à travailler en équipe sur un sujet qui nous est à la base inconnu et sur une période relativement importante. En plus de l'acquisition de nouvelles connaissances sur le sujet et d'un esprit d'équipe d'important, ce projet nécessite l'utilisation de Latex pour la rédaction du rapport. L'apprentissage de l'utilisation de ce logiciel est pour nous un bagage supplémentaire en tant que futur ingénieur.

Chapitre 1

Méthodologie, organisation du travail

Nous avons débuté le projet par des recherches globales sur le sujet. Il est vrai qu'il nous a fallu un certain temps pour cerner le sujet et comprendre ce que l'on attendait de nous. Ainsi nous avons longuement étudié les rapports des années précédentes. Ensuite nous nous sommes répartis le travail entre la partie théorique et l'implémentation informatique. Nous avons réalisé un point tous ensemble à chaque cours sur les avancées dans ces deux parties et vérifier qu'elles concordent. Finalement, nous avons mis en œuvre une expérience en laboratoire dans le but de comparer ces résultats avec ceux donnés par le programme informatique.

Voici un tableau récapitulatif de la distribution des tâches tout au long du projet :

tâche à effectuer	membres en charge	semaine
familiarisation avec le sujet	tout le monde	semaine 1 à 5
établissement de l'équation de la chaleur	tout le monde	semaine 5
méthode de résolution explicite en 1D		semaine 6 à 8
Partie mathématiques	Pierre Chardonnal Floriane Gloz Katie Leblond	
Partie informatique	Quentin Egly Mathieu Medici	
méthode de résolution implicite en 1D		semaine 9 à 12
Partie mathématiques	Pierre Chardonnal Floriane Gloz Katie Leblond	
Partie informatique	Quentin Egly Mathieu Medici	
expérience en laboratoire	tout le monde	semaine 13
comparaison des résultats	tout le monde	semaine 14
rédaction du rapport	tout le monde	au long du projet
préparation de l'oral	tout le monde	semaine 14

Chapitre 2

Travail réalisé et résultats

2.1 Etablissement de l'équation de la chaleur

On part des formules vues en P1 / T1 qui peuvent être appliquées pour un solide.

On applique le premier principe de la thermodynamique à notre barre qui est considérée comme un volume solide de conducteur, homogène.

$dU = \delta Q + \delta W$ (variation d'énergie interne entre t et $t+dt$, exprimé en J).

Or notre barre est un système isochore donc $dV = 0 \implies \delta W = -PdV = 0 \implies dU = \delta Q$ (1)

On va chercher à exprimer dU et δQ separemment.

Expression de dU :

On pose : $dU(t) = \iiint \rho c dT d^3V$

Avec :

ρ Masse volumique ($Kg.m^{-3}$)

c Chaleur spécifique ($J.K^{-1}.Kg^{-1}$)

Alors qu'en thermodynamique on étudiait la variation de U en fonction de la variation de température T , on cherche ici à l'exprimer en fonction du temps. T est (entre autre) fonction du temps ce qui donne :

$$dT = \frac{\partial T}{\partial t} dt \text{ donc en remplaçant dans } U \text{ on obtient : } dU(t) = \iiint \rho c \frac{\partial T}{\partial t} dt d^3V$$

Calcul de δQ (chaleur reçue élémentaire= chaleur reçue pendant dt) :

La chaleur reçue est égale à la chaleur produite dans la barre (source de chaleur) et celle perdue par échange avec l'extérieur (flux de chaleur). On peut donc écrire :

$$\delta Q = \Phi dt + F(M)dt$$

Avec :

Φ Flux de chaleur ($J.s^{-1}$)

$F(M)$ Source de chaleur ($J.s^{-1}$)

Expression de Φ :

D'après la loi de Fourier on sait que :

$$\Phi = - \iint \varphi d^2 S = - \iint -k_{tr} \overrightarrow{\text{grad}}(T) d^2 S$$

où φ est la densité de flux en $J.m^{-2}.s^{-1}$ (le signe - devant l'intégrale vient du fait qu'il s'agit d'un flux sortant donc négatif).

k_{tr} est la conductivité thermique du milieu en $J.m^{-1}.K^{-1}.s^{-1}$

Pour obtenir une expression avec une triple intégrale de volume (comme pour dU) on peut utiliser la formule de Green- Ostrogradsky pour Φ :

$$\Phi = \iint k_{tr} \overrightarrow{\text{grad}}(T) d^2 S = \iiint \text{div}(k_{tr} \overrightarrow{\text{grad}}(T)) d^3 V$$

On pose également :

$$F(M) = \iiint f(M) d^3 V$$

Avec $f(M)$ "Densité de source de chaleur" ($J.s^{-1}.m^{-3}$)

On peut alors remplacer dans (1) ce qui nous donne :

$$\iiint \rho c \frac{\partial T}{\partial t} d^3 V = \iiint k_{tr} \Delta(T) d^3 V dt + \iiint f(M) d^3 V dt$$

Donc :

$$\rho c \frac{\partial T}{\partial t} = k_{tr} \Delta(T) + f(M)$$

Finalement on obtient l'équation de la chaleur :

$$f(M) = \rho c \frac{\partial T}{\partial t} - k_{tr} \Delta(T)$$

2.2 Résolution de l'équation de la chaleur en 1D

Pour toutes nos modélisations, on pose comme conditions aux limites :

$$T_0^p = 0$$

$$T_N^{p+1} = T_{N-1}^{p+1} \text{ (condition de Neumann sur le flux)}$$

et comme condition initiale : $T_n^0 = T_0(x_n)$.

2.2.1 Méthode explicite

On utilise la formule de Taylor-Young à l'ordre 2.

$$T(x_n+h, t_p-\tau) = T(x_n, t_p) + h \frac{\partial T}{\partial x_n}(x_n, t_p) - \tau \frac{\partial T}{\partial t_p}(x_n, t_p) + \frac{1}{2} \left(\frac{\partial^2 T}{\partial x_n^2}(x_n, t_p) h^2 - h \tau \frac{\partial^2 T}{\partial x_n \partial t_p}(x_n, t_p) + \tau^2 \frac{\partial^2 T}{\partial t_p^2}(x_n, t_p) \right) - \tau h \varepsilon(\tau h)$$

Or $\frac{1}{2}(-h\tau \frac{\partial^2 T}{\partial x_n \partial t_p}(x_n, t_p) + \tau^2 \frac{\partial^2 T}{\partial t_p^2}(x_n, t_p)) - \tau h \varepsilon(\tau h)$ sont négligeables devant $T(x_n, t_p)$ et on cherche à se rapprocher le plus possible de la forme de l'équation de la chaleur.

$$\text{On a donc : } T(x_n + h, t_p - \tau) = T(x_n, t_p) + h \frac{\partial T}{\partial x_n}(x_n, t_p) - \tau \frac{\partial T}{\partial t_p}(x_n, t_p) + \frac{h^2}{2} \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

De même, on trouve que :

$$T(x_n - h, t_p - \tau) = T(x_n, t_p) - h \frac{\partial T}{\partial x_n}(x_n, t_p) - \tau \frac{\partial T}{\partial t_p}(x_n, t_p) + \frac{h^2}{2} \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

On additionne les 2 approximations :

$$T(x_n + h, t_p - \tau) + T(x_n - h, t_p - \tau) = 2T(x_n, t_p) - 2\tau \frac{\partial T}{\partial t_p}(x_n, t_p) + h^2 \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

On prend $h=0$ pour avoir la différentielle d'ordre 1 :

$$2T(x_n, t_p - \tau) = 2T(x_n, t_p) - 2\tau \frac{\partial T}{\partial t_p}(x_n, t_p)$$

$$\Leftrightarrow \frac{\partial T}{\partial t_p}(x_n, t_p) = \frac{T(x_n, t_p) - T(x_n, t_p - \tau)}{\tau} \quad (*)$$

On remplace la différentielle d'ordre 1 trouvée précédemment dans l'addition des 2 approximations, on trouve donc :

$$\begin{aligned} T(x_n + h, t_p - \tau) + T(x_n - h, t_p - \tau) &= 2T(x_n, t_p) - 2(T(x_n, t_p) - T(x_n, t_p - \tau)) + h^2 \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p) \\ \Leftrightarrow \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p) &= \frac{T(x_n + h, t_p - \tau) + T(x_n - h, t_p - \tau) - 2T(x_n, t_p - \tau)}{h^2} \quad (*) \end{aligned}$$

L'équation de la chaleur est de la forme :

$$\frac{\partial T}{\partial t_p} - k \frac{\partial^2 T}{\partial x_n^2} = f(x_n, t_p)$$

On remplace avec les approximations (*) :

$$\begin{aligned} \frac{T(x_n, t_p) - T(x_n, t_p - \tau)}{\tau} - k \frac{T(x_n + h, t_p - \tau) + T(x_n - h, t_p - \tau) - 2T(x_n, t_p - \tau)}{h^2} &= f(x_n, t_p) \\ \Leftrightarrow T_n^p &= \tau f_n^p + \frac{\tau k}{h^2} T_{n-1}^{p-1} + (1 - 2 \frac{\tau k}{h^2}) T_n^{p-1} + \frac{\tau k}{h^2} T_{n+1}^{p-1} \end{aligned}$$

Cette formule permet de calculer la température à un temps p à partir de la température à la date $p-1$ (formule de récurrence 1 du programme informatique).

2.2.2 Méthode implicite

On utilise la même méthode de calcul que pour la méthode explicite sauf que l'on se place au temps suivant à la place du temps précédent. Ce qui donne :

Formule de Taylor-Young à l'ordre 2.

$$T(x_n + h, t_p + \tau) = T(x_n, t_p) + h \frac{\partial T}{\partial x_n}(x_n, t_p) + \tau \frac{\partial T}{\partial t_p}(x_n, t_p) + \frac{h^2}{2} \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

$$T(x_n - h, t_p + \tau) = T(x_n, t_p) - h \frac{\partial T}{\partial x_n}(x_n, t_p) + \tau \frac{\partial T}{\partial t_p}(x_n, t_p) + \frac{h^2}{2} \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

On additionne les 2 approximations :

$$T(x_n + h, t_p + \tau) + T(x_n - h, t_p + \tau) = 2T(x_n, t_p) + 2\tau \frac{\partial T}{\partial t_p}(x_n, t_p) + h^2 \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

On prend $h=0$ pour avoir la différentielle d'ordre 1 :

$$2T(x_n, t_p + \tau) = 2T(x_n, t_p) + 2\tau \frac{\partial T}{\partial t_p}(x_n, t_p)$$

$$\Leftrightarrow \frac{\partial T}{\partial t_p}(x_n, t_p) = \frac{T(x_n, t_p + \tau) - T(x_n, t_p)}{\tau} \quad (*)$$

On remplace la différentielle d'ordre 1 trouvée précédemment dans l'addition des 2 approximations, on trouve donc :

$$T(x_n + h, t_p + \tau) + T(x_n - h, t_p + \tau) = 2T(x_n, t_p) + 2(T(x_n, t_p + \tau) - T(x_n, t_p)) + h^2 \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p)$$

$$\Leftrightarrow \frac{\partial^2 T}{\partial x_n^2}(x_n, t_p) = \frac{T(x_n + h, t_p + \tau) + T(x_n - h, t_p + \tau) - 2T(x_n, t_p + \tau)}{h^2} \quad (*)$$

L'équation de la chaleur est de la forme :

$$\frac{\partial T}{\partial t_p} - k \frac{\partial^2 T}{\partial x_n^2} = f(x_n, t_p + \tau)$$

On remplace avec les approximations (*) :

$$\frac{T(x_n, t_p + \tau) - T(x_n, t_p)}{\tau} - k \frac{T(x_n + h, t_p + \tau) + T(x_n - h, t_p + \tau) - 2T(x_n, t_p + \tau)}{h^2} = f(x_n, t_p + \tau)$$

$$\Leftrightarrow \tau f_n^{p+1} = \frac{-k\tau(T_{n+1}^{p+1} + T_{n-1}^{p+1})}{h^2} + T_n^{p+1} \left(1 + \frac{2k\tau}{h^2}\right) - T_n^p$$

On prend $n = 1; \dots; N - 1$:

$$f_1^{p+1} + \frac{T_1^p}{\tau} = \frac{T_1^{p+1}}{\tau} - \frac{k}{h^2} T_2^{p+1} + 2 \frac{k}{h^2} T_1^{p+1} - \frac{k}{h^2} T_0^{p+1}$$

Or $T_0^{p+1} = 0$ d'après les conditions initiales. Donc $\tau f_1^{p+1} + T_1^p = T_1^{p+1} - \frac{\tau k}{h^2} T_2^{p+1} + 2 \frac{\tau k}{h^2} T_1^{p+1}$

$$\tau f_2^{p+1} + T_2^p = T_2^{p+1} - \frac{\tau k}{h^2} T_3^{p+1} + 2 \frac{\tau k}{h^2} T_2^{p+1} - \frac{\tau k}{h^2} T_1^{p+1}$$

⋮

$$\tau f_{N-1}^{p+1} + T_{N-1}^p = T_{N-1}^{p+1} - \frac{\tau k}{h^2} T_N^{p+1} + 2 \frac{\tau k}{h^2} T_{N-1}^{p+1} - \frac{\tau k}{h^2} T_{N-2}^{p+1}$$

Or d'après la condition de Neumann $\frac{\partial T}{\partial x}(l, t) = 0$ (c'est-à-dire que le flux est nul en l)

Cela implique $T_N^{p+1} = T_{N-1}^{p+1}$

$$\Leftrightarrow \tau f_{N-1}^{p+1} + T_{N-1}^p = T_{N-1}^{p+1} \left(1 + \frac{\tau k}{h^2}\right) - \frac{\tau k}{h^2} T_{N-2}^{p+1}$$

$$\begin{pmatrix} \tau f_1^{p+1} + T_1^p \\ \tau f_2^{p+1} + T_2^p \\ \vdots \\ \tau f_{N-1}^{p+1} + T_{N-1}^p \end{pmatrix} = \begin{pmatrix} 1 + \frac{2k\tau}{h^2} & \frac{-k\tau}{h^2} & 0 & \cdots & 0 \\ \frac{-k\tau}{h^2} & \ddots & \ddots & 0 & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{-k\tau}{h^2} & 1 + \frac{k\tau}{h^2} \end{pmatrix} \begin{pmatrix} T_1^{p+1} \\ T_2^{p+1} \\ \vdots \\ T_{N-1}^{p+1} \end{pmatrix}$$

On a choisi d'utiliser la condition de Neumann car c'est plus simple de considérer le flux thermique nul. Si on avait décidé d'utiliser la condition de Dirichlet, la dernière ligne serait différente.

Afin de résoudre cette matrice, on demande au programme d'effectuer l'algorithme de Thomas qui est une adaptation de la méthode de Gauss pour ces matrices. Le calcul est réalisé par le programme en deux étapes. Tout d'abord, il triangularise la matrice, puis il détermine les inconnues.

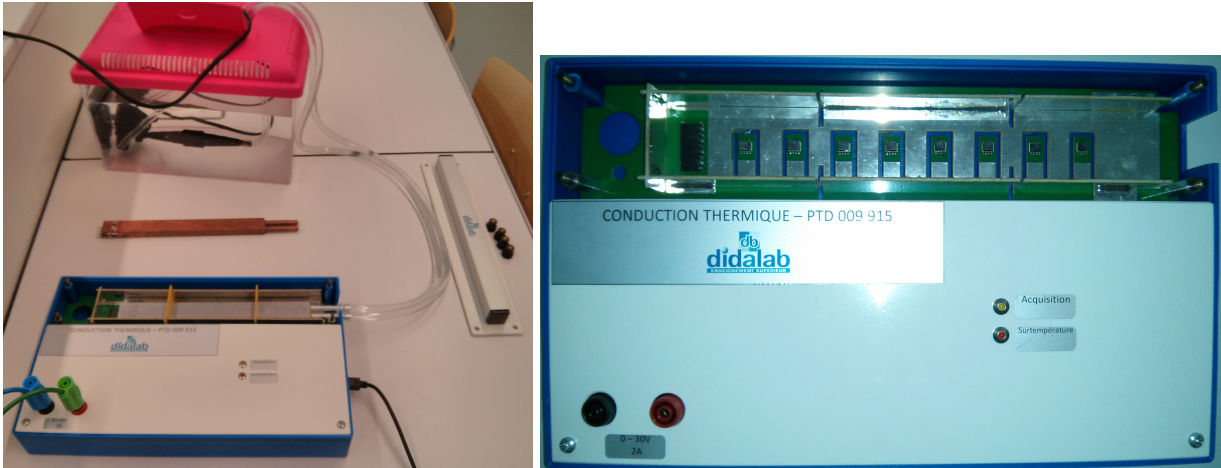
2.3 Implémentation informatique

Résoudre l'équation de la chaleur d'un système "à la main" peut être très laborieux et gourmand en temps. Heureusement, nous pouvons résoudre ces équations grâce à l'outil informatique. Grâce à de la documentation, nous avons pu réaliser un programme permettant de résoudre l'équation d'un système suivant la méthode explicite ou implicite.

Dans le programme, nous pouvons rentrer les différents paramètres pour définir le système étudié. On peut choisir entre autre, la longueur de la barre, la longueur de l'échantillon de temps, le nombre de points de la subdivision du temps et de l'espace ainsi que les pas de temps et d'espace.

Comme expliqué précédemment, la méthode explicite repose sur une relation de récurrence. La méthode implicite quant à elle utilise l'algorithme de Thomas pour une matrice tridimensionnelle. (cf annexe)

2.4 Expérience



Lors de notre dernière séance de p6, nous avons eu l'opportunité de réaliser une expérience en rapport avec notre projet.

Grâce à un dispositif décrit ci-dessous et un logiciel sur l'ordinateur CONDUCTOO, nous avons obtenu, via une acquisition du logiciel, des courbes et valeurs de la température à l'intérieur de la barre (aux 8 capteurs présents sur cette barre) en fonction du temps.

Description du dispositif :

Il est composé d'un boîtier de 30 cm sur 17 cm dans lequel on a inséré une barre d'aluminium de 17,4cm de long. La barre est reliée à un dispositif de chauffage par une extrémité et à un système de refroidissement à eau par l'autre extrémité, et possède 8 capteurs disposés tous les 2,2cm. Le dispositif de refroidissement est un circuit à eau fermé. L'eau est pompée dans un « aquarium » et envoyée dans l'ailette via un tuyau. Elle ressort de l'ailette par un deuxième tuyau qui se déverse dans ce même aquarium.

Remarque sur les graphiques et valeurs :

- En annexe se trouvent les courbes de températures aux 8 capteurs en fonction du temps. Il s'agit du 3ème essai car celui-ci fut le plus concluant.
 - On remarque qu'un régime permanent s'établit au bout d'environ 800secondes.
- L'expérience diffère de l'étude théorique de par les conditions initiales qui sont différentes :
- Au 1er capteur en x proche de 0 (arrivée d'eau), la température reste approximativement constante (température ambiante).
 - De plus, dans l'étude théorique nous avons considéré que le flux était nul au bout 1 de la barre alors que dans l'expérience le flux est nul au bout 0 de la barre.

Conclusion et perspectives

Malgré un début de projet où le sujet nous semblait confus et les objectifs vagues, nous avons réussi à préciser cela tout au long des séances et comprendre progressivement la modélisation et l'implémentation de l'équation de la chaleur.

Nous avons utilisé à travers ce projet des connaissances acquises pendant les premiers semestres de l'INSA, connaissances que l'on imaginait pas pouvoir relier. Mais aussi, cela nous a permis à tous d'approfondir ces connaissances et de découvrir de nouvelles choses.

De plus, nous avons grâce à ce projet eu l'occasion de mettre en place une véritable démarche scientifique afin de mieux cerner l'étude d'un phénomène physique, la conduction thermique. Cet apprentissage pour suivre une démarche scientifique au sein d'une équipe est un réel atout pour notre future carrière d'ingénieur.

L'expérience nous a donné l'opportunité de tester le modèle. Mais malheureusement, le TP a été réalisé en "bonus" en dernière séance, ainsi nous n'avons pas pu l'exploiter suffisamment pour valider de manière expérimentale notre modèle.

Finalement, nous avons trouvé ce projet complet car chaque étape réalisée est justifiée à la fois de manière analytique, à l'aide de l'implémentation informatique. Nous pourrions aller plus loin en faisant la même démarche dans le cas de la méthode 2D et en approfondissant les résultats de l'expérience.

Bibliographie

- [1] <http://www.iut-acy.univ-savoie.fr/fileadmin/import/files/2-conduction.pdf> (Valide à la date du 04/04/2014)
- [2] http://www.sciences.univ-nantes.fr/sites/claude_saintblanquet/conducti/cddex.htm (Valide à la date du 12/04/2014)
- [3] <http://docinsa.insa-lyon.fr/polycop/download.php?id=160504&id2=3> (Valide à la date du 12/04/2014)
- [4] <http://apacini.free.fr/physique/diffusion.pdf> (Valide à la date du 16/05/2014)
- [5] *Conduction thermique - cours de MRI3*, 2012.
- [6] *Rapport conduction thermique*, 2011-2012-2013.

Annexe A

Code du programme de résolution de l'équation de la chaleur en explicite et en implicite

```
program EquationDeLaChaleur;

uses crt;

5
const TabMax=50;
    Nt=9; (*nombre de points de la subdivision de l'espace temps*)
    N=3;  (*nombre de points de la subdivision de l'espace des 'x'*)
    TMax=0.5; (*longueur de l'espace des temps*)
10    l=1; (*longueur de l'espace des x*)
    Dt=TMax/Nt; (*Deltat = pas selon l'espace temps*)
    h=1/N; (*Pas selon l'espace des x*)

Type Tabulation=Array[0..TabMax] of real;

15
function f(i,j:Real):real; (*fonction donne qui est utilise pour calculer Up[
    i] dans la mthode explicite*)
var x,y:real;
begin
    x := i*h;
20    y := j*Dt;
    f := (x*sqr(1-x)*(3*x*(4*sqr(y)-1)-8*y)/(1+4*sqr(y))-2*(3*x-2-3*y*(1+
        x*(6*x-6)))/(1+4*sqr(y)));
end;

function exact(i,j:real):real; (*fonction utilise dans la procedure supdif
    pour calculer la solution exacte au point (i,j)*)
25 begin
    exact:=(sqr(1-i*h)*i*h*(1-3*j*Dt*i*h))/(1+4*sqr(j*Dt))
end;

procedure supdif(i:integer;j:real;U:Tabulation;var sup,jo:real; var io:
    integer);
30 (*cette procdure calcule la difference entre la fonction solution et la
    solution exacte,
```

```

au point (i,j), et enregistre le maximum de cette difference sur la grille
*)
var a:real;
begin
  a:=abs(U[i]-exact(i,j));
35  if a >= sup then
      begin
          sup := a;
          io :=i;
          jo:=j
40      end
end;

procedure Recurrence1(U:tabulation;T,Tp:Real; var Up:tabulation; var sup,
  jo:real; var io:integer); (*Methode explicite*)
  var i:integer;
45  L:Real;
begin
  L:=Dt*SQR(N); (*r=t/h coefficient de stabilit; L=r*a o a est le
    coefficient de diffusion, a=1: r=L. Dt=pas de temps*)
  for i:=1 to N-1 do
    begin
50      Up[i]:=U[i]+L*(U[i-1]-2*U[i]+U[i+1])+(Tp-T)*f(i,(Nt*T)/TMax); (*
        calcul de rcurrence*)
        supdif(i,(Nt*T)/TMax,U,sup,jo,io); (*calcul de la difference*)
    end;
    Up[0]:=0; (*condition aux limites en 0: temprature nulle*)
    Up[N]:=Up[N-1]; (*condition aux limites au bout de la barre: gradient
      de temprature nul = pas de flux*)
55  supdif(N,(Nt*T)/TMax,U,sup,jo,io);
end;

procedure TriDiag(Dim:Integer; T:real;Sous,Diag,Sur,Donnee:Tabulation; var
  Result:Tabulation; var sup,jo:real; var io:integer);
  var i:integer;
60 begin (*algorithmme de Thomas*)
  for i:=2 to Dim do Diag[i]:=Diag[i]-Sous[i]/Diag[i-1]*Sur[i-1];
  for i:=2 to Dim do Donnee[i]:=Donnee[i]-Sous[i]/Diag[i-1]*Donnee[i-1];
  Result[Dim]:=Donnee[Dim]/Diag[Dim];
  for i:=Dim-1 downto 1 do
65  begin
      Result[i]:=(Donnee[i]-Sur[i]*Result[i+1])/Diag[i];
      supdif(i,(Nt*T)/TMax,Donnee,sup,jo,io); (*calcul de la difference*)
    end;
end;

70 procedure Recurrence2(U:tabulation;T,Tp:Real; var Up:tabulation; var sup,
  jo:real; var io:integer); (*mthode implicite*)
  Var i: integer;
  Sous,D,Sur:Tabulation;
  L:REAL;
75 begin
  L:=Dt*sqr(N);

```

```

for i:=1 to N-1 do (*On initialise les 3 diagonales de la matrice*)
begin
    Sous[i]:=-L;
80     D[i]:=1+2*L;
    Sur[i]:=-L;
end;
    D[N]:=1+L;
    TriDiag(N,T,Sous,D,Sur,U,Up,sup,jo,io); (*on determine Up grace l'
        algorithme de Thomas*)
85     Up[0]:=0; (*condition aux limites en 0: temprature nulle*)
    Up[N]:=Up[N-1]; (*condition aux limites au bout de la barre: gradient
        de temprature nul = pas de flux*)
    supdif(N, (Nt*T)/TMax,U,sup,jo,io);
end;

90 function U0(x:real):real; (*fonction utilise dans la procedure Init*)
begin
    U0:=x*sqr(1-x);
end;

95 procedure Init(var U:Tabulation;N:Integer); (*La procedure Init calcule
    les tempratures initiales dans la barre, i.e. t=0*)
    var i:integer;
begin
    for i:=0 to N do U[i]:=U0(I/N);
end;

100 (*Dbut du programme principal*)
var U,Up:tabulation;
    i,io:Integer;
    jo,sup,T,Tp,Lambda:Real;
105    car:Char;
begin
    clrscr;
    writeln('EQUATION DE LA CHALEUR SUR [0,1]');
    writeln('N=',N,' l'intervalle sur x est ',h:7:5);
110    writeln('Resolution de t=0 a t=',TMax:7:4);
    repeat
        writeln('Methode (E)xplícite (I)mplicite (F)in');
        Car:=UpCase(ReadKey);
115        case Car of
            'E': writeln('Choix de la methode explicite');
            'I': writeln('Choix de la methode implicite');
        end;
        writeln;
        If Car<>'F' then
120        begin
            io:=0;
            jo:=0;
            sup:=0; (*les 3 variables io, jo et sup seront utilises pour
                le calcul de la difference*)
            Init(U,N); (*on initialise U l'instant t=0*)

```

```

125     Lambda:=Dt/SQR(h); (*Lambda vrifie si la condition de stabilit
        est respecte. On doit trouver Lambda <= 0.5*)
        writeln(Lambda);
        if Lambda > 0.5 then
            writeln('La condition de stabilit n''est pas vrifie')
        else
130     begin
            writeln('Lambda = ',Lambda:7:5,' ',Nt,' pas en t');
            T:=0;
            for i:=0 to N do writeln('U[' ,i,' ',', (Nt*T)/TMax:1:0, ']=',U[
                i]);
            while T<=TMax do
135     begin
                Tp:=T+Dt; (*On fait varier l'espace temps de Dt*)
                if Car='E' then Recurrence1(U,T,Tp,Up,sup,jo,io) (*On
                    lance la mthode explicite*)
                else Recurrence2(U,T,Tp,Up,sup,jo,io); (*On lance la
                    mthode implicite*)
140     U:=Up; (*U contient les donnes l'instant t, Up l'
                    instant t+1. Ici on se dcale pour calculer les
                    donnes l'instant t+2 en utilisant les donnes t
                    +1*)
                    T:=Tp;
                    if (Nt*T)/TMax < Nt+1 then for i:=0 to N do writeln('U['
                        ,i,' ',', (Nt*T)/TMax:1:0, ']=',U[i]);
                    end;
                    writeln('la difference maximum entre la solution approchee
                        et la solution exacte est ',sup,' en',io:4,' ',jo:4:0)
145     end;
                end;
            end;
            until Car='F';
        end.

```

Annexe B

Graphique

Graphique des courbes de la température à l'intérieur de la barre (aux 8 capteurs présents sur cette barre) en fonction du temps.

