© J.-B. A. K. <jean-baptiste.apoung@math.u-psud.fr>

Devoir surveillé : 07 avril 2016 (Durée 1h15)

IMPORTANT- Consignes

— Les notes de cours et les scripts de TP sont autorisés. La consultation des pages Internet, en particulier de votre message électronique, est interdite. Le non respect de cette consigne entraînera l'annulation de votre note.

— Commencez par créer un répertoire **`M325_DS1_###`** où **`###`** est votre NOM. Travailler dans ce répertoire. Il devra contenir tous les fichiers dont vous aurez eu besoin.

— Lorsque la réponse à une question nécessite des commentaires rédigés, vous devez mettre ces commentaires dans vos fichiers en utilisant les commentaires du langage C.

— A la fin de l'examen, vous devez envoyer votre répertoire **`M325_DS1_###`** zippé, par mail à l' adresse suivante : **`jean-baptiste.apoung@math.u-psud.fr`**

Thème - 1 *Problème*

On considère l'équation aux dérivées partielles suivante : $\Omega \subset \mathbb{R}^2$.

$$\begin{cases} -\Delta u(x,y) + \alpha \frac{\partial u}{\partial x}(x,y) = f(x,y), \quad \forall (x,y) \in \Omega =]a,b[\times]c,d[. \\ u(x,y) = g(x,y), \quad \forall (x,y) \in \partial\Omega, \end{cases} \tag{1}$$

où, f et g sont des fonctions données continues, et $\alpha \geq 0$ un réel donné.

Nous nous intéressons à sa discrétisation par différences finies sur une grille de $\bar{\Omega}$ de pas h_x et h_y dans les directions x et y, et de sommets $(x_i, y_j), 0 \leq i \leq n_x + 1, 0 \leq j \leq n_y + 1$ avec $x_0 = a, x_{n_x+1} = b, y_0 = c, y_{n_y+1} = d$.

En séance de TDM (voir TP1) nous avons traité le cas $\alpha = 0$ et avons construit le schéma à 5 points suivant :

$$\begin{cases} -\frac{1}{h_x^2}u_{i-1,j} - \frac{1}{h_x^2}u_{i+1,j} - \frac{1}{h_y^2}u_{i,j+1} - \frac{1}{h_y^2}u_{i,j-1} + (\frac{2}{h_x^2} + \frac{2}{h_y^2})u_{i,j} &= f(x_i,y_j), \quad 1 \leq i \leq n_x \quad 1 \leq j \leq n_y \\ u_{i,j} &= g(x_i,y_j) \quad i,j \text{ tels que } (x_i,y_j) \in \partial\Omega \end{cases} \tag{2}$$

où $u_{i,j}$ est une valeur approchée de $u(x_i, y_j)$ pour tout $0 \leq i \leq n_x + 1, 0 \leq j \leq n_y + 1$.

Ce schéma est connu pour être consistant d'ordre 2 en x et 2 en y.

On souhaite à présent proposer un schéma aux différences finies pour le problème (1) lorsque $\alpha > 0$.

Note 1 (Données numériques et guide pour mise en oeuvre).

Dans tout ce qui suit, on prendra :
$f(x,y) = -4 + 2\alpha x, \quad g(x,y) = x^2 + y^2, \quad \Omega =]0,1[\times]0,1[$,
de sorte que la **solution exacte** soit $u(x,y) = x^2 + y^2$.
On pourra utiliser et faire de simples modifications ou rajouts dans les fichiers :
`Laplacien2D.h, Laplacien2D.c, test_Laplacien2D.c` fournis lors dans le TP1 et rappelés en partie dans le **Thème-2** ci-dessous.

Un schéma aux différences finies potentiellement oscillant

Dans un premier temps on considère le schéma suivant :

$$\left\{\begin{array}{rcl} -\dfrac{1}{h_x^2}u_{i-1,j}-\dfrac{1}{h_x^2}u_{i+1,j}-\dfrac{1}{h_y^2}u_{i,j+1}-\dfrac{1}{h_y^2}u_{i,j-1}+(\dfrac{2}{h_x^2}+\dfrac{2}{h_y^2})u_{i,j}+\alpha\,\dfrac{u_{i+1,j}-u_{i,j}}{h_x} &=& f(x_i,y_j),\ 1\le i\le n_x,\ 1\le j\le n_y \\ u_{i,j} &=& g(x_i,y_j),\ i,j \text{ tels que } (x_i,y_j)\in\partial\Omega \end{array}\right. \quad (3)$$

Q-1 : **Consistance.** Sachant que $\frac{g(x+h,y)-g(x,y)}{h} = \frac{\partial g}{\partial x}(x,y)+\mathcal{O}(h)$, sans faire de calcul, donner l'ordre de consistance de ce schéma en x et en y.

Q-2 : **Mise en oeuvre.** Implémenter le schéma à travers une fonction :

`void Laplacien2D_construit_systeme_DCAV(Laplacien2D* pb)`

Q-3 : **Validation** On prendra $\alpha = 125$, $n_y = 10$ et deux valeurs de n_x

Q-3-1 : Pour $n_x = 40$, écrire dans un fichier **`solution_dcav_40.txt`**, la solution du problème.

Q-3-2 : Faire de même pour $n_x = 125$ et **`solution_dcav_124.txt`**

Q-4 : Que remarquez-vous ? On pourra représenter chaque solution accompagnée de la solution exacte.

Un schéma aux différences finies non oscillant d'ordre 1

On considère à présent le schéma suivant :

$$\left\{\begin{array}{rcl} -\dfrac{1}{h_x^2}u_{i-1,j}-\dfrac{1}{h_x^2}u_{i+1,j}-\dfrac{1}{h_y^2}u_{i,j+1}-\dfrac{1}{h_y^2}u_{i,j-1}+(\dfrac{2}{h_x^2}+\dfrac{2}{h_y^2})u_{i,j}+\alpha\,\dfrac{u_{i,j}-u_{i-1,j}}{h_x} &=& f(x_i,y_j),\ 1\le i\le n_x,\ 1\le j\le n_y \\ u_{i,j} &=& g(x_i,y_j),\ i,j \text{ tels que } (x_i,y_j)\in\partial\Omega \end{array}\right. \quad (4)$$

Q-5 : **Consistance.** Sans faire de calcul, donner l'ordre de consistance de ce schéma en x et en y.

Q-6 : **Mise en oeuvre.** Implémenter ce schéma à travers une fonction

`void Laplacien2D_construit_systeme_DCAM1(Laplacien2D* pb);`

Q-7 : **Validation.** Pour $\alpha = 125$, $n_y = 10$, $n_x = 40$. Ecrire la solution dans un fichier **`solution_dcam1_40.txt`**

Q-8 : Que remarquez-vous ? On comparera au même cas test pour le schéma (3).

Un schéma aux différences finies d'ordre 2

On considère à présent le schéma suivant :

$$\left\{\begin{array}{rcl} -(\dfrac{1}{h_x^2}+\dfrac{\alpha}{h_x})u_{0,j}-\dfrac{1}{h_x^2}u_{2,j}-\dfrac{1}{h_y^2}u_{1,j+1}-\dfrac{1}{h_y^2}u_{1,j-1}+(\dfrac{2}{h_x^2}+\dfrac{2}{h_y^2}+\dfrac{\alpha}{h_x})u_{1,j} &=& f(x_1,y_j),\ 1\le j\le n_y \\ -\dfrac{u_{i-1,j}}{h_x^2}-\dfrac{u_{i+1,j}}{h_x^2}-\dfrac{u_{i,j+1}}{h_y^2}-\dfrac{u_{i,j-1}}{h_y^2}+(\dfrac{2}{h_x^2}+\dfrac{2}{h_y^2})u_{i,j}+\alpha\,\dfrac{3u_{i,j}-4u_{i-1,j}+u_{i-2,j}}{2h_x} &=& f(x_i,y_j),\ 2\le i\le n_x,\ 1\le j\le n_y \\ u_{i,j} &=& g(x_i,y_j),\ i,j \text{ tels que } (x_i,y_j)\in\partial\Omega \end{array}\right. \quad (5)$$

Q-9 : **Consistance.** Sachant que $\dfrac{3g(x,y)-4g(x-h,y)+g(x-2h,y)}{2h} = \dfrac{\partial g}{\partial x}(x,y)+\mathcal{O}(h^2)$, sans faire de calcul, donner l'ordre de consistance de ce schéma en x et en y.

Q-10 : **Mise en oeuvre.** Implémenter ce schéma à travers une fonction

`void Laplacien2D_construit_systeme_DCAM2(Laplacien2D* pb);`

Q-11 : **Validation.** Pour $\alpha = 125$, $n_y = 10$, $n_x = 40$. Ecrire la solution dans un fichier **`solution_dcam2_40.txt`**

Q-12 : Comparer ce schéma au précédent (4). *On renseignera sur le schéma qui semble plus précis, plus stable.*

Il vous a été remis en TDM les fichiers **Laplacien2D.h, Laplacien2D.c, test_Laplacien2D.c**. vous devez vous en servir pour cette épreuve. Seuls les contenus de **Laplacien2D.h** et **test_Laplacien2D.c** sont fournis ici.

Listing 1 – fichier test_Laplacien2D.c

```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>
#include <mcheck.h>
#include <assert.h>
#include <string.h>

#include <gsl/gsl_linalg.h>
#include <gsl/gsl_rng.h>
#include <gsl/gsl_blas.h>
#include "Laplacien2D.h"

//compilation  gcc -I. Laplacien2D.c test_Laplacien2D.c -o test_Laplacien2D -lgsl -lgslcblas -lm

double  f(double x, double y){return -4.;}
double  g(double x, double y){return x*x + y*y;}

void TestLaplacien2D(){
 Laplacien2D pb;
 pb.a = 0; pb.b = 1;
 pb.c = 0; pb.d = 1;
 pb.f = f; pb.g = g;

int n0 = 4;
for(int i = 1; i < 4; i++){
n0 *= 2;
 pb.n = n0; pb.m = n0;
 Laplacien2D_construit_systeme(&pb);
 Laplacien2D_resout_systeme(&pb);
 Laplacien2D_affiche_solution_gnuplot(&pb,"solution.txt");
 double h = pb.x[1] -  pb.x[0];
 printf(" Erreur commise %f  %f\n", Laplacien2D_calcule_erreur(&pb,g) ,  h );
 Laplacien2D_libere(&pb);
 }
}

//
int main (int argc, char **argv)
{
  mtrace ();
  TestLaplacien2D ();
  muntrace ();
  return 0;
}
```

Listing 2 – fichier Laplacien2D.h

```c
 /*
   M325 : Calcul Scientifique II
   L3 MINT  Univ. Paris Sud ORSAY
   Copyright (C) 2016 APOUNG KAMGA Jean-Baptiste
   This is part of S.A.F.E.M.I.R.A TOOLS
*/
#ifndef LAPLACIEN2D_H
#define LAPLACIEN2D_H
#include <stdio.h>
#include <stdlib.h>
#include <gsl/gsl_linalg.h>
/*=================================================================
//
//    Auteur: J.-B. APOUNG KAMGA
//    Date:  28 / 03 /2015
//    Time:  17:14
//=================================================================*/
/**
   discretisation de
   -u''(x,y) = f          sur ]a,b[ x ]c,d[
    u(a,y) = u(b,y) = g  sur [c,d]
    u(x,c) = u(x,d) = g  sur [a,b]
    sur une subdivision de [a,b] x [c,d] generant
```

```
    n points internes suivant l'axe des x
    m points internes suivant l'axe des y
    La matrice sera une matrice carree formee
    de n * m lignes et n* m colonnes.
*/
typedef double (*PFunc)(double x, double y);

typedef struct Laplacien2D{
// description du èproblme
double a, b; // x est dans [a,b]
double c, d; // y  est dans [c,d]
PFunc f; // terme source
PFunc g; // condition Dirichlet

// description de la grille
int n;    // nombre de points internes suivant  l'axe x
int m;    // nombre de points internes suivant  l'axe y

// TOUT CE QUI SUIT EST GENERE POUR VOUS
// VOUS NE DEVEZ FOURNIR QUE LES DONNEES CI-DESSUS
// VOUS NE DEVEZ PAS EFFACER LES MEMOIRES ALLOUEES
// UTILISER POUR CELA  LA FONCTION FOURNIE
// Laplacien2D_libere(void* pb);

double* x; // maillage de [a,b] uniforme de pas h = (b-a)/(n+1)
double* y; // maillage de [c,d] uniforme de pas h = (d-c)/(m+1)
int ** C2I; // table ne numerotation lexichographique des sommets
            // interne du maillage
            // la numerotation globale des sommets est dans un ordre
            // lexicographique: de bas en haut et de gauche a droite.
// probleme discret
gsl_matrix* A;   //matrice obtenue en stockage pleine
gsl_vector* F; //second membre obtenu
gsl_vector* U;//solution du systeme lineaire
}Laplacien2D;

void Laplacien2D_bien_decrit(Laplacien2D* pb);
//
void Laplacien2D_construit_systeme(Laplacien2D* pb);
//
void Laplacien2D_resout_systeme(Laplacien2D* pb);
//
void Laplacien2D_affiche_solution_gnuplot(Laplacien2D* pb, const char* nomFichier);
//
double Laplacien2D_calcule_erreur(Laplacien2D* pb, PFunc solutionExacte);
//
void Laplacien2D_libere(void* pb);
#endif
```