TD/TP 5 de Calcul numérique Méthodes itératives de base

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

UNIVERSITÉ PARIS SACLAY
M1 CHPS

Rédigé par
Ahmed Taleb BECHIR

2023/2024
Encadrant : T. Dufaud J. Gurhem

Introduction

1 Travail préliminaire : Etablissement d'un cas de test

1.1 Exercice 1. Fait en TD

Introduction

L'étude de la conduction thermique dans des matériaux linéaires homogènes unidimensionnels joue un rôle crucial dans la compréhension du comportement thermique dans diverses applications scientifiques et d'ingénierie. Dans ce contexte, l'équation de conduction thermique, une équation aux dérivées partielles fondamentale, fournit une description mathématique de l'évolution de la température dans un domaine spatial donné. L'objectif principal de ce rapport est de résoudre numériquement l'équation de conduction thermique unidimensionnelle avec des conditions aux limites prescrites en utilisant une méthode de différences finies.

Considérons un milieu linéaire homogène avec un coefficient de conductivité thermique constant $k > 0$, subissant une conduction thermique le long d'un domaine unidimensionnel $[0, 1]$. La distribution de la température dans ce domaine est régie par l'équation de conduction thermique :

$$-k\frac{\partial^2 T}{\partial x^2} = 0, \quad x \in [0, 1]$$

où T représente la température et x la coordonnée spatiale. Les conditions aux limites sont données par $T(0) = T_0$ et $T(1) = T_1$, où T_0 et T_1 sont les températures aux frontières gauche et droite, respectivement. Pour résoudre ce problème numériquement, nous utilisons un schéma de différences finies centrées du second ordre pour approximer la dérivée seconde de la température par rapport à x. La discrétisation du domaine conduit à un système d'équations linéaires, nous permettant de représenter le problème sous la forme $Au = f$, où A est une matrice tridiagonale symétrique, u est le vecteur des températures inconnues, et f est le côté droit. La solution numérique de ce système fournit une approximation de la distribution de la température dans le domaine.

Dans ce rapport, nous présentons les détails de la stratégie de résolution numérique, la formulation du système linéaire et les résultats obtenus grâce à l'application des méthodes de différences finies. Cette étude sert de base pour comprendre et appliquer des techniques numériques pour résoudre des problèmes de conduction thermique dans des scénarios plus complexes.

1.2 Approximation de la dérivée seconde de T au moyen d'un schéma centré d'ordre 2

Pour approximer la dérivée seconde T au moyen d'un schéma centré d'ordre 2, on peut utiliser la formule de différences finies suivante :

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{T_{i-1} - 2T_i + T_{i+1}}{(\Delta x)^2}$$

1.3 le système linéaire de dimension n correspondant au probléme 1.

$$\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ 0 & -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix} = \begin{bmatrix} T_0 \\ 0 \\ \vdots \\ 0 \\ T_1 \end{bmatrix}$$

2 Méthode directe et stockage bande

2.1 Référence et utilisation de BLAS/LAPACK

1. Pour déclarer et allouer une matrice en C pour utiliser BLAS et LAPACK, vous pouvez utiliser un tableau à une dimension (vecteur) pour stocker les éléments de la matrice.

2. La constante `LAPACK COL MAJOR` signifie que les matrices sont stockées en mode "column-major order" (stockage par colonne). En LAPACK, les éléments d'une matrice sont stockés colonne par colonne, ce qui signifie que les éléments d'une colonne consécutive sont contigus en mémoire. Cela diffère du mode "row-major order" (stockage par ligne), où les éléments d'une ligne consécutive sont contigus en mémoire. `COL MAJOR` est souvent utilisé pour interagir avec BLAS et LAPACK en C.

3. La dimension principale (*leading dimension*), généralement notée `ld`, correspond au nombre d'éléments entre deux éléments consécutifs de la même colonne dans la mémoire. Cela dépend de la convention de stockage de la matrice. En mode LAPACK COL MAJOR, `ld` serait le nombre de lignes de la matrice. Il est utilisé pour accéder efficacement aux éléments de la matrice en mémoire. Lorsque vous appelez une fonction LAPACK ou BLAS, vous devez spécifier la dimension principale pour assurer un accès correct aux données stockées.

4. La fonction `dgbmv` effectue une multiplication d'une matrice générale stockée sous forme de bandes par un vecteur. Elle implémente la multiplication d'une matrice générale par un vecteur, et elle peut être utilisée pour réaliser des opérations de produit matrice-vecteur.

5. La fonction `dgbtrf` réalise la factorisation LU d'une matrice générale stockée sous forme de bandes. Elle implémente la méthode de factorisation LU pour une matrice générale et peut être utilisée pour résoudre des systèmes linéaires et calculer le déterminant d'une matrice.

6. La fonction `dgbtrs` résout un système linéaire avec une matrice générale stockée sous forme de bandes, en utilisant les résultats de la factorisation LU préalablement effectuée par `dgbtrf`. Elle implémente la méthode de substitution arrière après la factorisation LU.

7. La fonction `dgbsv` résout un système linéaire avec une matrice générale stockée sous forme de bandes en effectuant la factorisation LU et la substitution arrière en une seule étape. Elle implémente la méthode de factorisation LU avec substitution arrière.

8. Pour calculer la norme du résidu relatif avec des appels BLAS, vous pouvez utiliser la fonction `dnrm2`, qui calcule la norme euclidienne d'un vecteur. Vous pouvez l'appliquer au vecteur du résidu après avoir effectué une opération matrice-vecteur avec `dgbmv` et obtenu la solution du système linéaire.

2.2 Stockage GB et appel à DGBMV

2.2.1 Proposez une méthode de validation

Une méthode de validation pourrait consister à comparer les résultats de la multiplication matrice-vecteur obtenus avec la fonction `dgbmv` avec ceux obtenus en utilisant une autre méthode de multiplication matrice-vecteur connue. Vous pourriez également valider en comparant les résultats avec une solution analytique ou en vérifiant la conformité aux propriétés attendues de la matrice de Poisson 1D.

2.3 DGBTRF, DGBTRS, DGBSV

La complexité temporelle de la factorisation LU est généralement en $O(n^3)$, où n est la taille de la matrice. Il s'agit d'une complexité cubique par rapport à la taille de la matrice.
La complexité temporelle pour résoudre un système d'équations linéaires en utilisant la factorisation LU est généralement en $O(n^2)$, où n est la taille de la matrice.

2.4 LU pour les matrices tridiagonales

Une méthode de validation consiste à multiplier la matrice LU obtenue pour vérifier si elle est égale à la matrice d'origine (avec une certaine tolérance numérique).

3 Méthode de résolution itératives

3.1 Implémentation C - Richardson

1. l'erreur par rapport à la solution analytique : 8.027617e-03
2. l'historique de convergence :

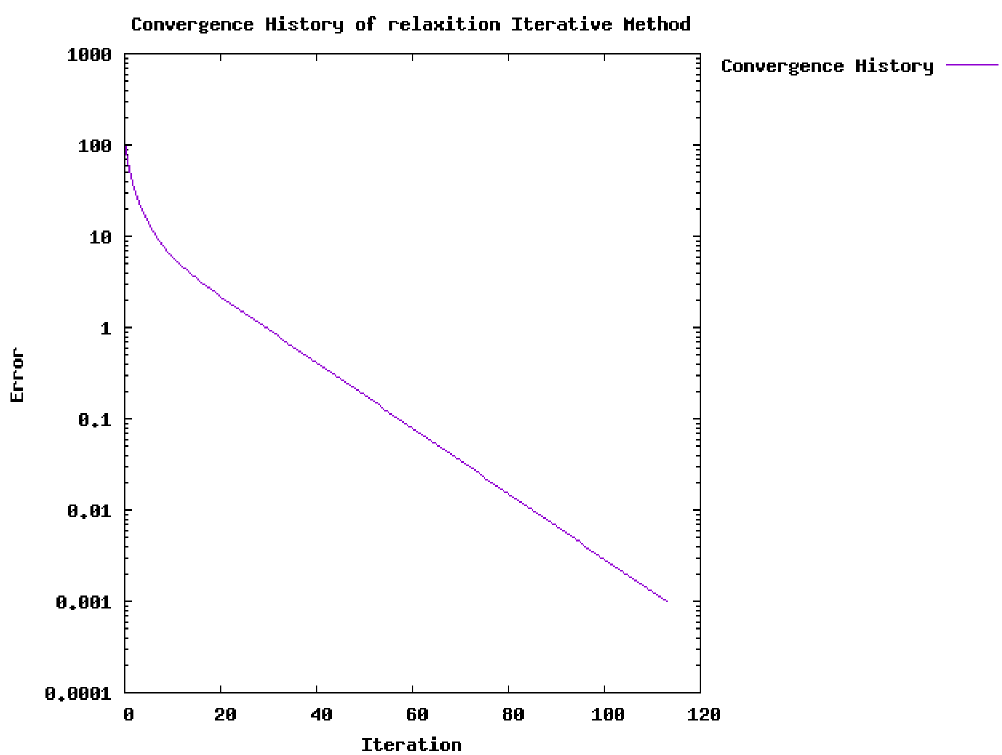

FIGURE 1 – historique de convergence de la méthode de Richardson

3.2 Implémentation C - Jacobi

1. l'erreur par rapport à la solution analytique : 2.489418e-04
2. l'historique de convergence :

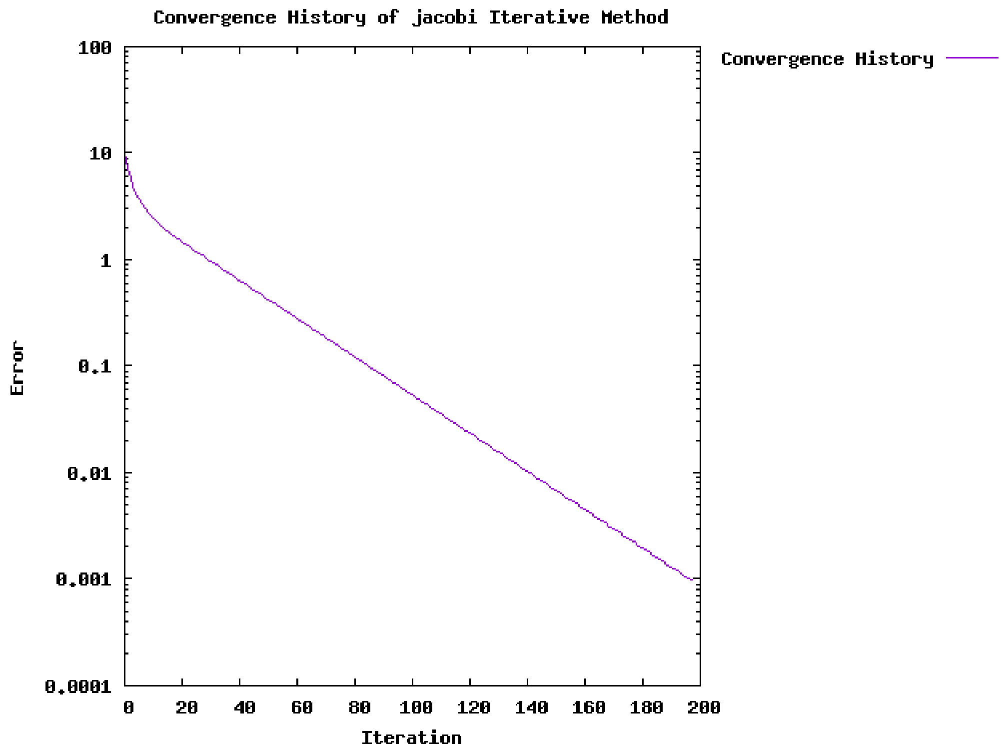

FIGURE 2 – historique de convergence de la méthode de Jacobi

3.3 Implémentation C -Gauss-Seidel

1. l'erreur par rapport à la solution analytique : 8.027617e-03
2. l'historique de convergence :

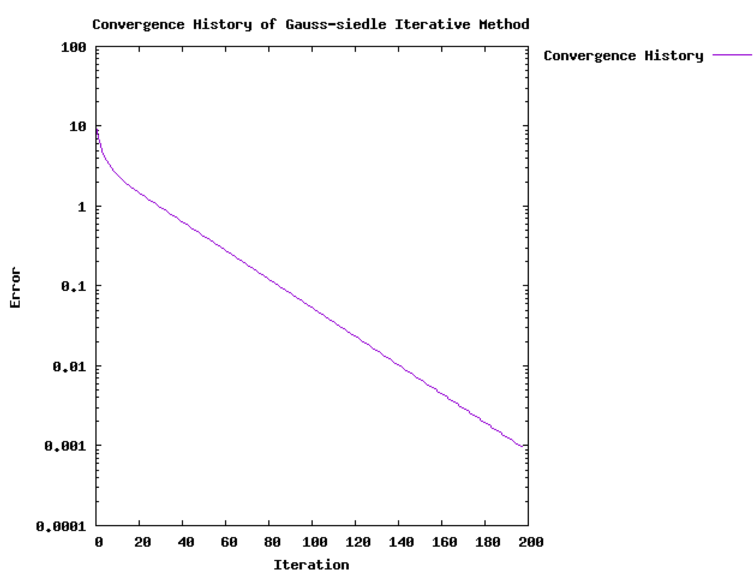

FIGURE 3 – historique de convergence de la méthode de Gauss-Seidel

3.4 comparaison entre les différents méthodes

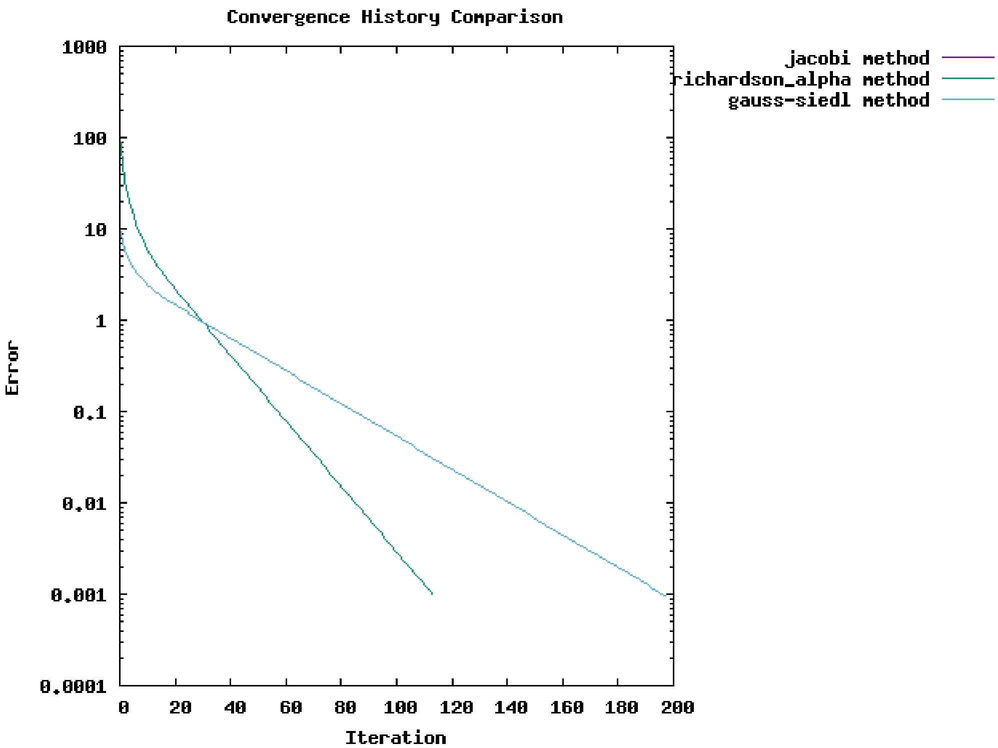

FIGURE 4 – comparaison entres les différentes convergence

4 Autres formats de stockage

4.1 Format CSR / CSC

Implémentation de CSR et CSC