Travaux pratiques Graphe II

M1 CSMI - Décembre 2017

Le but de ce TP est d'utiliser des algorithmes de théorie des graphes pour optimiser la résolution d'un système linéaire issu de la discrétisation d'une EDP.

Vous devez modifier le programme en C fourni afin de répondre aux questions suivantes.

Il faut rendre à la fin de la séance un compte-rendu par personne (le travail à plusieurs est autorisé ainsi que les recherches sur internet).

Le programme fourni est constitué de 4 fichiers :

— skyline.h et skyline.c : une structure C et des fonctions pour afficher et résoudre un système linéaire stocké sous forme ligne de ciel ("skyline").

— cutmckee.c et cutmckee.h : une structure C et des fonctions pour résoudre le problème de Poisson sur un rectangle avec conditions de Dirichlet.

Ce programme résout par différence finies le problème

$$-\Delta u = 1 \text{ dans } \Omega =]0, L_x[\times]0, L_y[,$$

$$u = 0 \text{ sur } \partial\Omega.$$

Le carré est discrétisé sur les points

$$(x_i, y_j) = (i\Delta x, j\Delta y), \quad 0 \le i \le N_x, \quad 0 \le j \le N_y.$$

avec

$$\Delta x = \frac{L_x}{N_x}, \quad \Delta y = \frac{L_y}{N_y}.$$

Il y a donc neq=$(N_x + 1)(N_y + 1)$ points. Au point (i, j) on associe l'indice

$$k = i + j(N_x + 1). \tag{1}$$

On a donc $0 \le k <$ neq. La fonction u est approchée aux points (i, j) de la grille par une quantité notée indifféremment $u_{i,j}$ ou u_k. Pour des raisons pratiques, les valeurs de u sur les points du bord sont considérées comme des inconnues fictives. Les inconnues sont rangées dans un vecteur

$$U = (u_0, u_1, \ldots, u_{\text{neq}-1})^T.$$

La compilation et l'exécution du code se font avec les commandes :

gcc cutmckee.c skyline.c -lm -ligraph -std=c99 -I/usr/include/igraph/

./a.out

Partie I (questions de compréhension)

1. Comment retrouve-t-on les indices i et j à partir de k (voir formule (1)) ? Expliciter les formules en C99 donnant i et j.
2. Écrire les équations vérifiées par les $u_{i,j}$ en précisant soigneusement dans quelles bornes varient les indices i et j.
3. Pour les points du bord, expliquer comment on utilise la méthode du grand pivot.
4. On pose $F = (1, \ldots, 1)^T \in \mathbb{R}^{\text{neq}}$. Pour $N_x = 3$ et $N_y = 2$, expliciter la structure de la matrice du système linéaire $AU = F$ (on indiquera par un symbole la position des termes non nuls de cette matrice).
5. Pourquoi est-il intéressant de stocker A sous forme ligne de ciel ? Expliquer rapidement comment on assemble cette matrice. Pourquoi doit-on procéder en 2 passes ?

Partie II (programmation)

1. Programmer la renumérotation de Cutill-McKee au moyen de la bibliothèque igraph. Expliquer. Comment stockez-vous la permutation associée ?
2. Testez votre programme pour le cas $N_x = 3$ et $N_y = 2$. Expliquer votre test.
3. Que devez-vous faire pour que le résultat affiché soit exactement le même que sans la renumérotation ?
4. On prend d'abord $N_x = 10$, $N_y = 1000$. Mesurer le temps de la résolution du système linéaire avec ou sans renumérotation. Pour mesurer le temps, on utilisera la fonction "time" du C. On ne mesurera que le temps de résolution du système linéaire. Pour accélérer la construction du graphe on pourra remplacer avantageusement la fonction igraph_add_edge par igraph_add_edges (qui permet d'ajouter tous les arcs en même temps).
5. Même question avec $N_x = 1000$, $N_y = 10$. Conclusion ?

Partie III (méthode de bissection)

Soit $G = (V, E)$ le graphe construit à la question 1 de la partie II. On suppose que les noeuds du graphe sont partitionnés en deux ensembles W_1 et W_2. On note $V_1 \subset W_1$ l'ensemble des noeuds de W_1 qui ne sont reliés qu'à des noeuds de W_1. On note $V_2 \subset W_2$ l'ensemble des noeuds de W_2 qui ne sont reliés qu'à des noeuds de W_2. On note $V_3 = V \backslash (V_1 \cup V_2)$.

1. Pour $N_x = 3$ et $N_y = 2$, on choisit $W_1 = \{0, 1, 4, 5, 8, 9\}$. Faire un dessin du graphe G et déterminer V_1, W_2, V_2 et V_3.
2. On renumérote en premier les n_1 sommets de V_1 puis les n_2 sommets de V_2 puis les n_3 sommets de V_3. Donc neq $= n_1 + n_2 + n_3$. Quelle est la structure de la matrice A après cette renumérotation ? Quel est l'avantage de cette nouvelle numérotation si la matrice est stockée sous forme ligne de ciel ? (on pourra comparer avec une numérotation où les sommets de V_3 sont mis en premier dans la liste).
3. Pour que la bissection soit efficace, il faut que $n_1 \simeq n_2$ et que n_3 soit le plus petit possible. Le problème de trouver la meilleure bissection est un problème difficile. Pour simplifier, on peut commencer par renuméroter les noeuds par l'algorithme de Cutill-McKee. Ensuite, on répartit équitablement les noeuds, en mettant le début de la liste obtenue dans W_1, puis la fin dans W_2. Programmer cet algorithme. Vérifier que la matrice a bien la structure attendue après renumérotation.
4. On note $E_i \subset E$ l'ensemble des arcs dont les deux extrémités sont dans V_i. La méthode des bissections emboîtées consiste à répéter récursivement le procédé précédent sur les sous-graphes (V_1, E_1) et (V_2, E_2) jusqu'à ce qu'on ne puisse plus partitionner les sous-graphes. Programmer l'algorithme de bissection emboîtée. Mesurer le gain obtenu avec $N_x = 63$ et $N_y = 63$.