Projet n° 2
Méthode du gradient conjugué / Application à l'équation de la chaleur

Groupe n° 4 - Equipe n° 1
Responsable : asehoubo
Secrétaire : cfarcy001
Codeurs : edao,thparpai

Résumé : Le but de ce projet consiste à implémenter des algorithmes de résolution de systèmes linéaires de grande taille, et à les appliquer à un problème de résolution d'équation aux dérivées partielles. Dans ce devoir, on considère uniquement des systèmes linéaires symétriques, définis positifs et creux (ne comportant que relativement peu d'éléments non nuls), et on exploite ces trois propriétés pour obtenir des algorithmes plus efficaces.

Présentation du travail

Répartition des tâches

Conclusion, axes d'améliorations

1 Décompositon de Cholesky

La factorisation de Cholesky consiste, pour une matrice symétrique définie positive A, à déterminer une matrice triangulaire inférieure T telle que $A = T^tT$. Ceci permet d'obtenir la matrice inverse A^{-1}. Il suffit que les relations suivantes soient vérifiées :

$$\begin{cases} t_{i,i}^2 = a_{i,i} - \Sigma_{k=1}^{i-1} t_{i,k}^2 \\ t_{i,j} = \frac{a_{i,j} - \Sigma_{k=1}^{i1} t_{i,k} t_{j,k}}{t_{i,i}} \; j \leq i \end{cases}$$

Dans l'algorithme de Cholesky implémenté, on a une imbrication de boucles for. Le premier indice j va de 1 à n, et le second i va de 1 à j. Une fois arrivé à cette deuxième boucle for, on peut rencontrer plusieurs cas, se séparant en également en deux cas. Si $i == j$, alors 2 cas se présentent : si $i < 0$, alors on n'effectue qu'une affectation et l'opération de la racine carrée ; sinon, une troisème boucle for est appelée d'indice k allant de 1 à $(i-1)$. Dans cette boucle, on n'effectue qu'une opération élémentaire.

Ainsi, dans le pire des cas, on a 3 boucles for imbriquées. N'appelant que des opérations élémentaires, on peut calculer sa complexité temporelle de la façon suivante :

$$\sum_{j=1}^{n}(\sum_{i=1}^{j}(\sum_{k=1}^{i-1} 1)) = \sum_{j=1}^{n}(\sum_{i=1}^{j}(i-1)) = \sum_{j=1}^{n}(\frac{j(j+1)}{2}) - \sum_{j=1}^{n}(j)$$
$$= \frac{1}{2} \times (\frac{n(n+1)(2n+1)}{6} - \frac{n(n+1)}{2}) = \frac{n(n+1)(2n-2)}{12}$$
$$\simeq O(n^3)$$

La complexité temporelle est donc cubique. L'algorithme ne fait que modifier la matrice qui lui est donné en argument. Elle ne prend donc pas de place en mémoire.

Lors de la résolution de $Ax = b$, on peut donc écrire $T^tTx = b$. On peut voir cette équation sous la forme $Ty = b$ où ${}^tTx = y$. On a donc 2 équations à résoudre. Pour obtenir ces 2 équations, il faut utiliser l'algorithme de Cholesky, de complexité cubique. Une fois la matrice triangulaire T trouvée, la résolution de $Ty = b$ se fait de façon carré. La deuxième équation se fait de manière similaire, tout en transposant la matrice T, ce faisant cette fois ci de façon linéaire. La complexité totale est donc de $O(n^3)$.

Pour diminuer cette complexité, il existe la factorisation de Cholesky incomplète. Elle consiste à ne pas calculer les éléments $t_{i,j}$ lorsque $a_{i,j} = 0$. Ceci n'influence pas sur le reste du calcul. Avec ceci, on réduit le nombre d'itérations.

2 Méthode du gradient conjugué

La méthode du gradient conjugué est une méthode qui permet de résoudre un système d'équations linéaires $Ax = b$ dans le cas où A est une matrice symétrique définie positive.

Cette méthode est souvent implémentée sous la forme d'un algorithme itératif. Dans ce cas, les résultats obtenus ne sont pas exacts comme dans le cas d'une méthode directe, mais convergeront vers une vecteur solution X. L'intérêt de cette approche est de pouvoir traiter des matrices creuses trop grandes pour être gérées par les méthodes directes, comme la décomposition de Cholesky.

Wikipedia propose une implémentation de cette méthode, mais qui ne respecte pas des standards de codage très sains. En effet, elle ne comporte pas de commentaires, est peu lisible à cause de l'absence d'espaces entre les opérateurs et utilise un *break* pour sortir d'une boucle au lieu d'en sortir par la vérification de la condition de sortie.

Nous avons donc implémenté la méthode du gradient conjugué en nous inspirant de l'algorithme proposé, puis testé sur ...

L'utilisation d'un préconditionneur permet d'assurer la convergence rapide de cette méthode. Nous avons donc modifié l'algorithme en conséquence, en utilisant le préconditionneur fourni par la décomposition de Cholesky, d'abord celle fournie par la bibliothèque Numpy de Python, puis par celle que nous avons implémentée.

3 Application à l'équation de la chaleur

3.1 Discrétisation de l'équation de la chaleur

Pour étudier l'équation de la chaleur en machine, il est nécessaire de discrétiser l'espace (i.e. la température) en un nombre fini de points. Pour cela on utiliser une matrice de taille $N^2 * N^2$ divisée en N blocs de taille $N * N$.

De plus, il est nécessaire de discrétiser les opérateurs pour travailler sur l'espace que l'on vient de créer (car les dérivées partielles ne sont plus applicables étant donné qu'on a rompu la continuité des valeurs). On applique alors la méthode des différences finies, c'est-à-dire qu'on les approche à l'aide des équations linéaires suivantes où H est la distance entre deux points consécutifs sur la même ligne ou colonne :

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_{i,j} \approx \frac{T(x_i + h, y_j) + T(x_i - h, y_j) - 2T(x_i, y_j)}{h^2} = \frac{t_{i+1,j} + t_{i-1,j} - 2t_{i,j}}{h^2}$$

$$\left(\frac{\partial^2 T}{\partial y^2}\right)_{i,j} \approx \frac{T(x_i, y_j + h) + T(x_i, y_j - h) - 2T(x_i, y_j)}{h^2} = \frac{t_{i,j+1} + t_{i,j-1} - 2t_{i,j}}{h^2}$$

On obtient ainsi un système linéaire de la forme $Ax = b$ qu'il est possible de résoudre grâce aux méthodes présentées dans les deux premières parties.

3.2 Applications

Nous avons résolu l'équation de la chaleur dans deux cas de figure : un premier où on considère que la source de chaleur provient d'un radiateur situé au centre de la pièce, puis un autre où on considère qu'il y a un mur chaud au Nord de celle-ci. Nous avons réalisé nos tests avec un paramètre de $N = 3$ et nous avons mis à 1 la coordonnée du vecteur b correspondant à la (ou les localisations) des cases souhaitées. Les résultats sont concluants : on observe en effet un pic de chaleur depuis la source puis une dissipation progressive de celle-ci. On constate également que les murs ont un rôle de "conservation" de la chaleur.

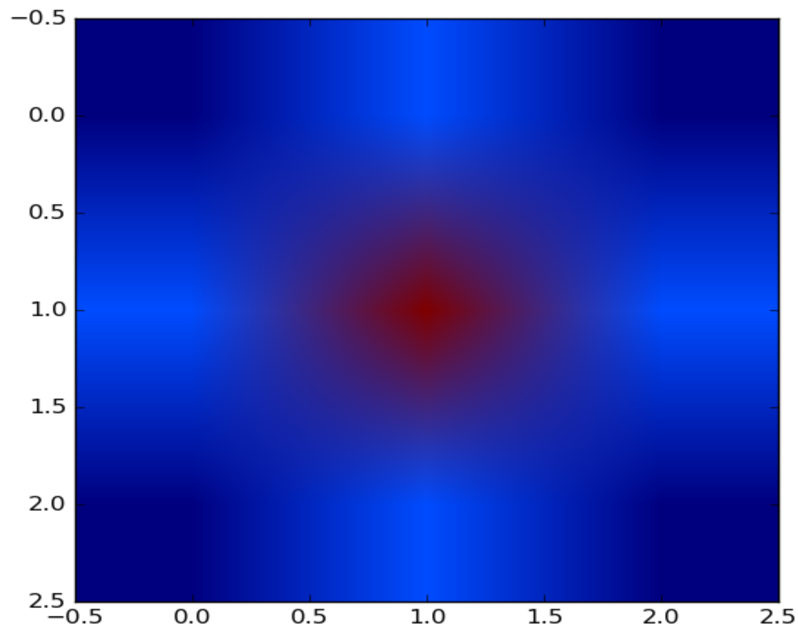

Figure 1 – Radiateur placé au centre du carré

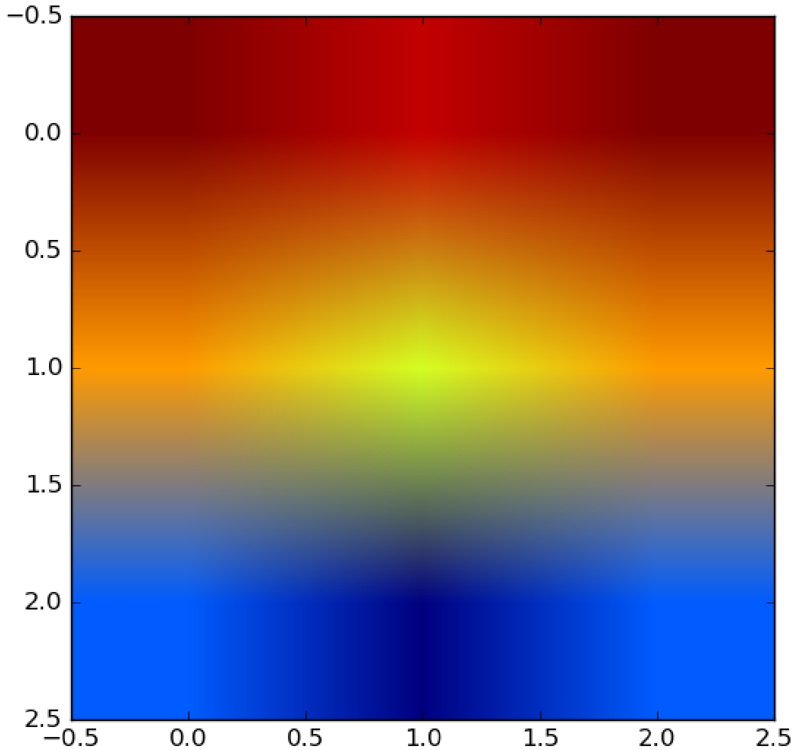

FIGURE 2 – Mur chaud placé au Nord du carré