TD/TP 4 de Calcul numérique
Exploitation des structures et calcul creux

T. Dufaud

—M1 CHPS—

Cette fiche de TD/TP est associée au cours de calcul numérique sur l'exploitation des structure de matrice et les matrices creuses.
Les exercices sont à réaliser sur feuille puis à implémenter en Scilab.

▸ **Exercice 1.** *Cours / TD : Factorisation LDL^T pour A symétrique*

À partir de la factorisation LU d'une matrice symétrique, on peut obtenir une factorisation LDL^T.

1. *montrez l'existence et l'unicité de la factorisation $A = LU$ avec L unit lower triangular*
2. *montrez que si A symétrique alors il existe une factorisation $A = LDL^T$ unique*
3. *proposez un algorithme pour la factorisation LDL^T*

▸ **Exercice 2.** *Appliquer la méthode de Gauss à la matrice tridiagonale suivante:*

$$A = \begin{pmatrix} a_1 & c_1 & 0 & \cdots & 0 & 0 \\ b_1 & a_2 & c_2 & \cdots & 0 & 0 \\ & & \ddots & & & \\ & & & \ddots & & \\ 0 & 0 & 0 & \cdots & a_{n-1} & c_{n-1} \\ 0 & 0 & 0 & \cdots & b_{n-1} & a_n \end{pmatrix}.$$

Montrer que, sans pivot, cette méthode réduit la matrice A à sa décomposition LU:

$$A = LU = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ e_1 & 1 & \cdots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & e_{n-1} & 1 \end{pmatrix} \begin{pmatrix} d_1 & c_1 & \cdots & 0 & 0 \\ 0 & d_2 & \cdots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \cdots & d_{n-1} & c_{n-1} \\ 0 & 0 & \cdots & 0 & d_n \end{pmatrix}.$$

où les e_i et d_j (pour $2 \leqq i \leqq n$ et $1 \leqq i \leqq n-1$) sont facilement calculables. Écrire, analyser et implanter l'algorithme de cette décomposition.

▸ **Exercice 3.** *[Stockage 1D]*

1. *Trouvez une formule qui détermine la localisation en mémoire de l'élément $A(i_1, i_2)$ du tableau $A(1, \cdots, u_1; 1, \cdots, u_2)$. On supposera que l'adresse de $A(1,1)$ est α et que les éléments du tableau sont stockés par ligne.*
2. *Trouvez une formule qui détermine la localisation en mémoire de l'élément $A(i_1, i_2, i_3)$ du tableau $A(1, \cdots, u_1; 1, \cdots, u_2; 1, \cdots, u_3)$. On supposera que l'adresse de $A(1,1,1)$ est α et que les éléments du tableau sont stockés par colonne.*

▸ **Exercice 4.** *[Stockage CSR et CSC] Soit la matrice suivante l'algorithme précédent sur la matrice:*

$$A = \begin{pmatrix} 15 & 0 & 0 & 22 & 0 & -15 & 0 & 0 \\ 0 & 11 & 3 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 91 & 0 & 0 & 0 & 0 & 0 & 25 & 7 \\ 0 & 0 & 28 & 0 & 0 & 0 & 0 & -2 \end{pmatrix}$$

1. *proposez un stockage CSR de A*
2. *proposez un stockage CSC de A*
3. *En utilisant une structure de données pour les matrices creuses issue de la compression par ligne ou par colonne de la matrice, écrire un programme qui accepte en entrée une matrice creuse A et donne en sortie sa matrice transposée (avec la même structure de données que A).*
4. *Évaluez les complexités en temps et en espace de cet algorithme.*

▸ **Exercice 5.** *[Produit Matrice Vecteur Creux] Écrire un algorithme qui permet de calculer le produit d'une matrice creuse A de m lignes et de n colonnes et un vecteur de longueur n.*

`(i)-` *Simuler cet algorithme sur le vecteur $x = (1, 1, \cdots, 1)^T$.*

`(ii)-` *Simuler cet algorithme sur le vecteur $x = (1, 0, 0, 1, 0, 0, \cdots, 1, 0, 0)^T$.*