

---

Dans ce devoir, nous nous intéressons à l'exploitation de la faible densité des matrices éléments finis dans le cadre d'une résolution au moyen de la méthode directe LU.

### Format creux CSR

Soit une matrice carrée  $A \in \mathbb{C}^{n \times n}$  de faible densité produite par le modèle `ndt.py`. Implémentez une fonction `PYTHON3 CSRformat(A)` effectuant la conversion en format CSR de la matrice  $A$ . Cette fonction reçoit un `numpy.array`  $A$  et retourne le tuple  $(sA, iA, jA)$  composé de trois `numpy.array`. Ce tuple contient respectivement les entrées non nulles de  $A$ , les débuts (et fins) de chaque ligne et les indices des colonnes correspondantes. En d'autres termes,

- $sA[s]$  donne le  $s$ -ème élément non nul de  $A$
- $iA[i]$  donne l'indice dans  $sA$  du premier élément non nul sur la ligne  $i$
- $jA[j]$  donne l'indice de la colonne contenant le  $j$ -ème élément non nul de  $A$

Évaluez brièvement les performances de votre convertisseur.

### LU creux

Écrivez une fonction `PYTHON3 LUcsr(sA, iA, jA)` effectuant la décomposition LU d'une matrice sous format CSR. À aucun moment une matrice de format plein ne doit être construite. Cette fonction retourne le tuple  $(sLU, iLU, jLU)$  correspondant à la décomposition LU sous format CSR. Vous pouvez travailler sans pivotage.

### Reverse Cuthill-McKee

Par ailleurs, nous vous demandons d'étudier l'influence de l'algorithme `reverse Cuthill McKee` sur la densité de la décomposition LU et sur la largeur de bande de la matrice. Il vous faut donc écrire une fonction `PYTHON3 RCMK(iA, jA)` qui retourne  $r$ , un `numpy.array` représentant le vecteur de permutation.

Évaluez la complexité temporelle de la résolution du système linéaire  $Ax = b$  obtenu dans le cas du régime harmonique ( $freq = 50\text{Hz}$ ,  $vel = 0\text{m/s}$ ) par la décomposition LU *creuse*, avec et sans l'algorithme RCMK. Discutez ses performances vis-à-vis de la LU pleine.

Bonus : Discutez la pertinence de recourir au pivotage dans le cas de l'algorithme LU creux.

## Consignes

Ce devoir est absolument **individuel**.

Un rapport de quatre pages (càd deux rectos et deux versos) est à remettre pour le vendredi 6 décembre à l'Euler A.108 (Astrid Leduc) avant 16h. En cas d'absence, vous pouvez simplement glisser votre devoir sous la porte.

Le `.pdf` de ce rapport, ainsi que le `.tex` (figures) sont à soumettre sur Moodle, ainsi que les différentes implémentations PYTHON demandées dans ce devoir. Le tout est compressé dans un dossier `.zip` (~~`.rar`~~). Vos travaux seront soumis à un logiciel anti-plagiat.

Le rapport ne doit pas contenir de page de garde, seulement une entête reprenant au moins le nom de l'auteur. Le rapport ne doit pas spécialement contenir de code source. La langue de rédaction est le français. Le rapport doit être réalisé avec L<sup>A</sup>T<sub>E</sub>X, avec la `documentclass article [11pt]` en `pagestyle plain`. Veuillez à fournir des implémentations lisibles, dûment commentées, avec des noms de variables explicites.

Les implémentations sont à réaliser en PYTHON 3. Les librairies admises sont :

- NUMPY
- MATPLOTLIB
- TIME

Toute autre librairie ne sera pas acceptée ; SCIPY est donc prohibée.