Analyse Numérique Licence L3 de Mathématique 2015-2016, TP6

Soit $A \in \mathcal{M}_n(\mathbb{R})$ et $b \in \mathbb{R}^n$. On va programmer l'algorithme de factorisation LU avec et sans pivotage ainsi que la descente remontée pour résoudre le système linéaire $Ax = b$.

(1) Programmer dans scilab l'algorithme de factorisation LU sans pivotage (avec stockage dans la matrice A). On écrira une fonction *A=LU(A,n)*. Faire un test d'arret sur la valeur du pivot avec un seuil égal à 10^{-16}.

For $k = 1, \cdots, n-1$ (boucle sur les pivots) tester $|A_{k,k}| > 10^{-16}$ sinon abort

For $i = k+1, \cdots, n$

$$A_{i,k} \leftarrow \frac{A_{i,k}}{A_{k,k}}$$

End For

For $i, j = k+1, \cdots, n$

$$A_{i,j} \leftarrow A_{i,j} - A_{i,k}A_{k,j}$$

End For

End For

(2) Programmer dans scilab la descente et la remontée pour résoudre $Ax = b$ en créant une fonction *x=SolLU(A,n,b)*.

Descente : $Lz = b$

$z = 0$

For $i = 1, \cdots, n$

$$z_i = b_i - \sum_{j=1}^{i-1} A_{i,j}z_j$$

End For

Remontée : $Ux = z$

$x = 0$

For $i = n, \cdots, 1$

$$x_i = \frac{z_i - \sum_{j=i+1}^{n} A_{i,j}x_j}{A_{i,i}}$$

End For

(3) Tester l'algorithme précédent de résolution de $Ax = b$ sur les différentes matrices de l'exercice. Que constatez vous ? Quelles solutions proposez vous ?