Département Génie Électrique – 3ème année

TP 4 d'Analyse Numérique : Résolution de système linéaire

Exercice 1. (Décomposition de Cholesky)
Nous nous intéressons dans cet exercice à la décomposition de Cholesky d'une matrice A symétrique définie positive :

$$A = LL^t.$$

(a) Écrire une fonction `A = matriceA(n)` qui prend en argument un entier n et qui renvoie la matrice A de taille n définie par

$$A_{i,j} = \min(i,j).$$

(b) Écrire une fonction `L = macholesky(A)` qui prend en argument une matrice A et qui retourne la matrice L issue de sa décomposition de Cholesky. (On testera l'algorithme sur la matrice A définie précédemment)

(c) Tracer en échelle logarithmique le temps de calcul numérique de la factorisation de Cholesky d'une matrice A en fonction sa taille n pour $n = 50, 100, 200, 400$. On pourra utiliser la commande `tic` et `toc` de `Matlab` pour estimer ce temps de calcul. Remarques ?

(d) Écrire une fonction `H = matriceH(n)` qui prend en argument un entier n et qui renvoie la matrice de Hilbert H de taille n définie par

$$H_{i,j} = \frac{1}{i+j-1}.$$

Utiliser une dernière fois la fonction `macholesky` sur les matrices de Hilbert H avec $n = 25, 50, 100$ et estimer l'erreur relative

$$\max_{i,j=1\cdots n} \left| \frac{H_{i,j} - (L\ L^t)_{i,j}}{H_{i,j}} \right|.$$

Remarques ?

Exercice 2. (Algorithme de Gauss Seidel)
Nous nous intéressons dans ce deuxième exercice à la résolution d'un système linéaire

$$Ax = b$$

à l'aide de la méthode de Gauss-Seidel. Nous supposons par la suite que A est une matrice symétrique définie positive.

(a) Écrire une fonction `[x_eps,N_iter] = maGaussSeidel(A,b,x0,eps)` qui prend en argument une matrice `A`, un vecteur `b`, un vecteur `x0`, un scalaire `eps` et qui renvoie une approximation `x_eps` de la solution du système linéaire $Ax = b$ et le nombre d'itérations `N_iter` effectuées par l'algorithme de Gauss-Seidel en partant de la condition initiale $x_0 =$`x0` et en utilisant le critère d'arrêt

$$\|Ax_k - b\| \leq eps.$$

(b) Tester la fonction `maGaussSeidel` avec
- `A = matriceA(10)`,
- $b = Ax$ où x est un vecteur aléatoire de taille 10,
- $x_0 = 0$,
- $eps = 10^{-4}, 10^{-5}, 10^{-6}$.

Remarques ?

On pourra utiliser la commande `rand` de MATLAB pour générer un vecteur aléatoire

(c) Déterminer numériquement la vitesse de convergence de l'algorithme dans le cas des matrice `A = matriceA(10)` et `A = matriceA(20)`. On essaiera en particulier d'estimer numériquement le coefficient λ tel que

$$\|x_k - x^*\| \simeq C(x_0)\lambda^k,$$

où x^* est la solution du système linéaire $Ax = b$ et $C(x_0)$ est une constante qui dépend de x_0.