Corso di Laurea Magistrale in Ingegneria Aerospaziale
A.A. 2012/2013 – II Ciclo

Esercitazione 1

1. Realizzare una function MATLAB ***g_seidel.m*** per la risoluzione di sistemi lineari con il relativo metodo iterativo. Sintassi: [x, iter] = gauss_seidel(A, b, x0, tol)

2. Nello script ***ex2.m*** utilizzare, quando possibile, i metodi iterativi di Gauss-Seidel, Gradienti coniugati e SOR implementati nei rispettivi m-file: ***g_seidel.m, conjgrad.m*** e ***SOR.m,*** per la risoluzione dei seguenti sistemi lineari Ax=b, ove

   (a) A=[5 0 -1 2; -2 4 1 0; 0 -1 4 -1; 2 0 0 3], b=$[15\ -4\ -8\ 3]^T$;
   prima con precisione tol=1e-3 e poi tol=1e-5. Ci si aspetta che i metodi convergano?
   Soluzione esatta x=$[3\ 1\ –2\ -1]^T$.

   (b) A ha dimensione n, **pentadiagonale**. Le uniche diagonali non nulle di A, oltre a quella principale che ha tutti elementi pari a 4, sono la prima e la terza sopra e sotto la diagonale principale che hanno elementi pari a -1. Come sempre b è scelto in modo tale che la soluzione esatta del sistema sia x=**ones**(n,1), x_0=zeros(n,1), n=10,100,1000, con precisione tol=1e-5 e Nmax=1000, (per avere il grafico di sparsità **spy(A)**)

   (c) Sistema a blocchi

$$A = \begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ 1 \\ 2 \\ 1 \\ 0 \\ 1 \\ 2 \\ 1 \\ 2 \end{bmatrix}$$

   che ha soluzione esatta x=$[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]^T$, prima con precisione tol=1e-3 e poi tol=1e-5.

Stampare in una tabella le componenti delle tre soluzioni e il numero di iterazioni utilizzate da ciascun metodo per ottenere la soluzione a meno di una tollleranza tol=10^{-3}. Valutare la velocità di convergenza dei metodi nella risoluzione dei sistemi lineari dati. Calcolare inoltre la norma del vettore errore tra soluzione esatta e soluzione calcolata.

3. **Metodi diretti vs metodi iterativi.** Nello script ***ex3.m*** utilizzare il metodo iterativo dei Gradienti coniugati ***conjgrad.m*** ed il metodo diretto di eliminazione di Gauss (‘\’ di MATLAB) per la risoluzione dei seguenti sistemi lineari Ax=b, ove A=hilb(n), n=8,12,15 con precisione tol=1e-6, x_0=zeros(n,1). Visualizzare gli errori ottenuti e il condizionamento dei sistemi lineari.

4. **Precondizionatori.** Nello script ***ex4.m*** Consideriamo la matrice A di dimensione m x m ottenuta dalla discretizzazione alle differenze finite dell'operatore di Laplace sul quadrato unitario [-1,1]x[-1,1]. Questa matrice è stata generata dal comando

G=numgrid('S',n); A=delsq(G);

dove $m=(n-2)^2$. A è simmetrica def. pos. e diventa mal condizionata per valori grandi di m.

- Calcolare il suo numero di condizionamento per m=400 e m=1600.

Costruire un precondizionatore mediante fattorizzazione incompleta di Cholesky $M=R^TR$ definito da IC(0) generato dal comando [R]= `ichol(A,struct('type','nofill'));`

- Calcolare il numero di condizionamento di $M^{-1}A$.
- Risolvere il sistema lineare Ax=b dove **b=A*ones(m,1)** con il metodo dei gradienti coniugati con e senza condizionatore confrontando il numero di iterazioni richieste.

  [X,FLAG,RELRES,ITER] = pcg(A,b,1e-10,200,M)