DI/PPGI/UFES
$2^{\underline{o}}$ Exercício Computacional de Algoritmos Numéricos I - 19/2
Sistemas Lineares - Métodos Iterativos usando o Octave

Objetivos

- Observar o comportamento dos métodos iterativos quanto as características da matriz dos coeficientes.

Conceitos importantes:

- Os métodos iterativos dependem de critérios de convergência:

$$x^{(k+1)} = M\, x^{(k)} + c \text{ converge } \Leftrightarrow \rho(M) < 1$$

  onde $\rho(M)$ é o maior módulo dos autovalores de M.

- Dado $Ax = b$, os métodos iterativos estacionários convergem se a matriz dos coeficientes A for diagoranl dominante (ou satisfazer o Critério das linhas):

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}| \quad \forall i = 1, \ldots, n$$

Comandos (e/ou funções) do Octave

- `tril(A,-1)` (obtem a submatriz estritamente inferior de `A`)
- `triu(A,1)` (obtem a submatriz estritamente superior de `A`)
- `diag(diag(A))` (obtem a matriz diagonal de `A`)
- `[V lambda]=eig(A)` (obtem os autovetores `V` e os autovalores `lambda` de `A`)
- `max(abs(diag(lambda)))` (obtem o maior valor em módulo dos elementos da diagonal de `lambda`)
- `load <nome>.mat` (carrega dados da matriz associada ao arquivo binário <nome>.mat[1] em uma estrutura auxiliar `A`
- `A = Problem.A;` (Armazena os dados da estrutura `A` na matriz esparsa `A` no formato CCR)
- `n = rows(A);`
- `plot(x,y)` (plota o grafico dos pontos de $y_i = f(x_i)$)
- `x = [1:n]` (gera o vetor $x = [1\ 2\ 3\ \ldots\ n]$)
- `log(x)` (calcula o logarítmo de x)

[1]Arquivo do Repositório SuiteSparse Matrix Collection

Implemente as seguintes funções:

- `[x,er,iter]=jacobi(A,b,tol,nmaxiter)`, sendo `tol` a tolerância pré estabelecida; `nmaxiter` o número máximo de iterações; `er` vetor contendo o erro relativo em cada iteração; `iter` número de iterações necessárias para convergir.
- `[x,er,iter]=sor(A,b,tol,nmaxiter,w)`, sendo `tol` a tolerância pré estabelecida; `nmaxiter` o número máximo de iterações; `er` vetor contendo o erro relativo em cada iteração; `iter` número de iterações necessárias para convergir; `w` parâmetro de relaxação (`w = 1`, método Seidel).
- `[MJ,MS,MSOR]=fatora(A,w)`, sendo `MJ`, `MS`, `MSOR`, matrizes dos métodos Jacobi, Seidel e SOR, respectivamente.
- `diagonal_dominante(A)`, retorna `TRUE` or `FALSE` caso a matriz seja ou não diagonal dominante

O objetivo dos execícios a seguir é observar o comportamento de matrizes esparsas na solução de sistemas lineares via métodos iterativos. Faça download de matrizes esparsas de ordem $n = 10^1; 10^2, 10^3, 10^4, 10^5$ na *SuiteSparse Matrix Collection* que sejam quadradas e inversíveis. Procure obter matrizes capazes de explorar as variadas características de convergência dos métodos iterativos. Para cada uma das matrizes:

1. Recupere a matriz esparsa a partir do arquivo `.mat`
2. Defina o vetor dos termos independentes `b = A*ones(n,1)`;
3. Verifique se a matriz é diagonal dominante;
4. Calcule o raio espectral da matriz de iteração dos métodos Jacobi, Seidel e SOR(w) para cada uma das matrizes obtidas.
5. Obtenha a solução pelos métodos Jacobi, Seidel e SOR se o raio espectral da matriz de iteração do método for menor que 1.0 para um conjunto de parâmetros `w` do método SOR. Estabeleça uma tolerância adequada para cada matriz.
6. Escolha o `w` que obtem o melhor comportamento para o SOR e faça o gráfico $iter \times log(er)$ dos três métodos no mesmo sistema de eixos.
7. O que podemos dizer sobre a convergência dos métodos Jacobi, Seidel e SOR(w) para cada uma das matrizes?

Relatório

Escreva um relatório com suas conclusões sobre os objetivos listados acima. Entregar os arquivos fonte e uma cópia do relatório em pdf (nome do arquivo AN192-EXE2-<nome1><nome2>) via email (luciac@inf.ufes.br) até 17/09/2019. O título do email deve ser AN192-EXE2-<nome-ultimosobrenome>.