

Universidade Federal do Espírito Santo
Departamento de Informática / Programa de Pós Graduação em Informática
1º Exercício Computacional de Algoritmos Numéricos II / Computação Científica - 2021
EARTE
Sistemas Lineares - Métodos Diretos e Métodos Iterativos Estacionários - usando o Octave

Objetivos

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

Conceitos/comandos importantes:

- A coleção de matrizes esparsas *SuiteSparse Matrix Collection*¹ disponibiliza uma variedade de matrizes esparsas oriundas das mais diversas áreas do conhecimento. Um dos formatos disponíveis para as matrizes é `<nome>.mat`. Arquivo binário que armazena as informações para gerar uma matriz esparsa no formato *Compressed Column Sparse*(CCR). Os principais comandos do Octave relativos a obtenção da matriz esparsa **A** são:
 - `load <nome>.mat` (carrega dados da matriz em uma estrutura auxiliar **A**)
 - `AA = Problem.A` (armazena os dados da estrutura **A** na matriz esparsa **AA** no formato CCR)
 - `nnz(A)` (obtem o número de coeficientes não nulo da matriz **A**)
 - `n = rows(A)`; (obtem o número de linhas da matriz **A**)
- Os métodos diretos são exatos a menos de erros de ponto flutuante cometidos no processo de transformar o sistema original em um sistema trivial. A seguir alguns importantes comando do Octave na solução de sistemas lineares via Métodos Diretos:
 - `x = A\b` (resolve o sistema linear por Eliminação de Gauss)
 - `r = b - A * x` (calcula o resíduo da solução aproximada encontrada)
 - `norm(x,*)` (obtem a norma *** do vetor *x* - por exemplo, pode ser a norma euclidiana *** = 2 ou a norma do máximo *** = `inf`) – Métricas importantes para avaliação da solução aproximada: $\|x_e - x\|_*$ e $\|r\|_*$, sendo x_e solução exata.
- Os métodos diretos são bem eficientes para matrizes de pequeno porte. Entretanto, o processo de solução prevê preenchimento de posições originalmente nulas (fill-in), aumentando assim o número de operações de ponto flutuante. Principais comandos do Octave:
 - `[L,U,P] = lu(A)` (obtem os fatores *L*, *U* e *P*)
 - `spy(A)` (obtem a configuração da esparsidade da matriz *A*)
- Uma matriz é dita mal-condicionada se:

$$\text{cond}(A) = \|A\|_* \|A^{-1}\|_* \quad \text{for um valor expressivamente elevado}$$

Comandos do octave:

- `cond(A)`
- `norm(A,*)` (obtem a norma *** da matriz *A* - por exemplo, pode ser a norma por coluna *** = 1 ou a norma por linha *** = `inf`)

¹<https://sparse.tamu.edu/>

- Os métodos iterativos dependem de critérios de convergência (critério necessário e suficiente):

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

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

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

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

- `plot(x,y)` (plota o gráfico dos pontos de $y_i = f(x_i)$)
- `x = [1:n]` (gera o vetor $x = [1 \ 2 \ 3 \ \dots \ n]$)
- `log(x)` (calcula o logaritmo de x)

Exercícios propostos

1. Métodos Diretos:

O objetivo do exercício a seguir é observar o comportamento de matrizes esparsas na solução de sistemas lineares via métodos diretos e iterativos estacionários. Visite o site <https://sparse.tamu.edu/> e faça download de matrizes esparsas quadradas e inversíveis de ordem $n = 10^2, 10^3$. Para cada uma das matrizes:

- Recupere as matrizes esparsas a partir do arquivo `.mat`
- Obtenha os fatores L , U e P utilizando a função `[L,U,P]=lu(A)`;
- Observe a configuração de esparsidade das matrizes A , L e U .
- Calcule a taxa de preenchimento: $100 - \left(\frac{\text{nnz}(A)}{\text{nnz}(L) + \text{nnz}(U)} \right) * 100$.
- Calcule a solução do sistema linear onde $\mathbf{b} = \mathbf{A} * \text{ones}(n, 1)$, através de $\bar{\mathbf{x}} = \mathbf{A} \backslash \mathbf{b}$.
- Calcule a distância relativa entre a solução exata e a solução aproximada $\frac{\|\delta \mathbf{x}\|_{\infty}}{\|\bar{\mathbf{x}}\|_{\infty}}$
- Calcule a distância relativa entre a matriz original e a matriz resultante da decomposição LU: $\frac{\|\delta \mathbf{A}\|_{\infty}}{\|\mathbf{A}\|_{\infty}}$; $\delta \mathbf{A} = \mathbf{A} - \mathbf{P} * \mathbf{L} * \mathbf{U}$.
- Calcule a distância relativa entre o vetor dos termos independentes original e o vetor resultante da decomposição LU: $\frac{\|\delta \mathbf{b}\|_{\infty}}{\|\mathbf{b}\|_{\infty}}$
- Calcule a norma do resíduo: através de $\|\mathbf{r}\|_{\infty} = \text{norm}(\mathbf{b} - \mathbf{A} * \mathbf{x}, \text{inf})$.
- Calcule o número de condicionamento da matriz: $K = \text{cond}(\mathbf{A})$.

Monte uma tabela contendo as informações observadas e calculadas de cada matriz. Faça um estudo sobre o comportamento de cada matriz com relação ao preenchimento no processo de decomposição e seu condicionamento. Podemos afirmar que as matrizes escolhidas são mal-condicionadas?

2. Métodos Iterativos:

O objetivo do exercício a seguir é observar o comportamento de matrizes esparsas na solução de sistemas lineares via métodos iterativos. Faça download de matrizes esparsas de ordem $n = 10^2, 10^3$ em <https://sparse.tamu.edu/> que sejam quadradas e inversíveis. Procure obter matrizes capazes de explorar as variadas características de convergência dos métodos iterativos. Para cada uma das matrizes:

- (a) Recupere a matriz esparsa a partir do arquivo `.mat`
- (b) Defina o vetor dos termos independentes `b = A*ones(n,1);`
- (c) Verifique se a matriz é diagonal dominante;
- (d) Obtenha a solução pelos métodos Jacobi, Seidel e SOR. Estabeleça uma tolerância adequada para cada matriz.
- (e) Escolha o `w` que obtém o melhor comportamento para o SOR e faça o gráfico $iter \times \log(er)$ dos três métodos no mesmo sistema de eixos.
- (f) O que podemos dizer sobre a convergência dos métodos Jacobi, Seidel e SOR(`w`) para cada uma das matrizes?

Considere as seguintes funções já implementadas no material de apoio.

- `[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).
- `diagonal_dominante(A)`, retorna `TRUE` or `FALSE` caso a matriz seja ou não diagonal dominante

Relatório

Escreva um relatório com suas conclusões sobre os objetivos listados acima. Postar uma cópia em pdf no ambiente Classroom até 06/07/2021.