

Instituto Tecnológico de Aeronáutica – ITA

Matemática Computacional – CCI-22

Laboratório 2 – Métodos Diretos para Solução de Sistemas Lineares de Equações

Professor: Marcos Ricardo Omena de Albuquerque Maximo

14 de março de 2019

1 Implementação:

Implementar as seguintes funções em MATLAB (cada uma em um arquivo .m separado):

1. `x = SolucaoTriangularSuperior(A, b)`: resolve um sistema linear de equações $\mathbf{Ax} = \mathbf{b}$ em que a matriz \mathbf{A} é triangular superior e retorna a solução \mathbf{x} . No caso, considera-se que $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{x} \in \mathbb{R}^{n \times 1}$ e $\mathbf{b} \in \mathbb{R}^{n \times 1}$ (note que \mathbf{x} e \mathbf{b} são vetores coluna).
2. `[Ap, bp] = EliminacaoGauss(A, b)`: executa Eliminação de Gauss **com pivoteamento parcial** no sistema linear de equações $\mathbf{Ax} = \mathbf{b}$ e retorna o sistema resultante $\mathbf{A}_p \mathbf{x} = \mathbf{b}_p$, em que \mathbf{A}_p é triangular superior.
3. `x = SolucaoTriangularInferior(A, b)`: resolve um sistema linear de equações $\mathbf{Ax} = \mathbf{b}$ em que a matriz \mathbf{A} é triangular inferior e retorna a solução \mathbf{x} .
4. `[L, U, P] = DecomposicaoLU(A)`: realiza Decomposição LU **com pivoteamento parcial** da matriz \mathbf{A} e retorna as matrizes resultantes, em que \mathbf{L} é triangular inferior, \mathbf{U} é triangular superior e \mathbf{P} é a matriz de permutações devido ao pivoteamento parcial.
5. `x = SolucaoLU(L, U, P, b)`: soluciona o sistema linear de equações $\mathbf{LU} = \mathbf{Pb}$ através da solução sucessiva dos sistemas $\mathbf{Ly} = \mathbf{Pb}$ e $\mathbf{Ux} = \mathbf{y}$. $\mathbf{LU} = \mathbf{Pb}$ é o sistema resultante da Decomposição LU com pivoteamento parcial aplicada a um sistema $\mathbf{Ax} = \mathbf{b}$. O retorno da função é a solução \mathbf{x} .

2 Análise

Verificar o ganho de desempenho obtido com decomposição LU no caso de solução de vários sistemas com mesma matriz A e com variações em b . Para isso, resolva 100 sistemas lineares de dimensão 100 utilizando ambos os métodos implementados: Eliminação de

Gauss e Decomposição LU. Para cada método, plote graficamente como o tempo total de execução (considerando desde a solução do primeiro sistema) varia com o número de sistemas resolvidos. Compare os resultados obtidos. Não é necessário efetivamente alterar b , dado que estamos interessados apenas em comparar tempos de execução. Lembre de não incluir os tempos necessários para construir A e b e de incluir o tempo necessário para realizar a decomposição na solução do primeiro sistema usando Decomposição LU.

3 Instruções:

- A primeira etapa do processo de correção consistirá em submeter as funções implementadas a vários casos de teste de forma automatizada. Assim, os cabeçalhos das funções devem ser seguidos **rigorosamente**. Arquivos .m com os nomes destas funções e os cabeçalhos já implementados foram fornecidos juntamente com este roteiro. Dê preferência a implementar seu laboratório a partir destes arquivos .m fornecidos para evitar erros.
- **Não** é permitido o uso de funções ou comandos prontos do MATLAB que realizem toda a funcionalidade atribuída a uma certa função: $\mathbf{x} = \mathbf{A} \setminus \mathbf{b}$, $\mathbf{x} = \text{linsolve}(\mathbf{A}, \mathbf{b})$ e $[\mathbf{L}, \mathbf{U}, \mathbf{P}] = \text{lu}(\mathbf{A})$ etc. Entretanto, o uso destas funções para verificação das implementações realizadas é encorajado. Em caso de dúvida quanto à permissão de uso de alguma função ou comando, recomenda-se consultar o professor.
- Não é necessário se preocupar com verificação dos dados de entrada: assuma que as dimensões de A e b são compatíveis, que a matriz A é não singular e que os sistemas usados são bem condicionados.
- Os arquivos .m implementados devem ser entregues juntamente com um relatório.
- No relatório, não é necessário demonstrar que as funções implementadas funcionam corretamente (isto será verificado separadamente). Basta incluir resultados e conclusões da **Análise**.
- Coloque os gráficos diretamente no relatório. Além disso, adicione nomes aos eixos das figuras e legendas para figuras que mostram mais de uma mesma curva em um mesmo gráfico.

4 Dicas:

- Cuidado com o formato requerido para vetores e matrizes (incluindo dimensões) para evitar problemas na correção automática. Pode ser descontado ponto no caso de formatação inadequada.
- Para que seu relatório tenha gráficos de alta qualidade, dê sempre preferência para formatos vetorizados.
- No Microsoft Word, utilize preferencialmente o formato SVG. No MATLAB, crie arquivos SVG com: `print -dsvg grafico.svg`.

- Se utilizar \LaTeX , dê preferência para formato EPS. No Linux, utilizando `pdflatex`, você pode gerar um gráfico em formato EPS usando “`print -depsc2 grafico.eps`” no MATLAB e depois convertê-lo para PDF usando o comando de terminal “`epstopdf grafico.eps`”. O arquivo PDF é aceito pelo `pdflatex`.
- Caso tenha problemas com formatos vetorizados, use PNG com boa resolução: `print -dpng -r300 grafico.png`.
- Utilize `A = rand(N)` e `b = rand(N,1)` para gerar sistemas lineares de equações rapidamente, em que N é a dimensão do sistema. Como o sistema é gerado aleatoriamente, você pode dar azar de obter um sistema mal condicionado. Assim, verifique o número de condição de A com `cond(A,Inf)` antes de usar o sistema.
- As seguintes funções e comandos do MATLAB podem ser úteis para verificar suas implementações:
 - `x = A \ b`: resolve o sistema linear de equações $Ax = b$ e retorna a solução x .
 - `[L, U, P] = lu(A)`: realiza Decomposição LU seguindo o mesmo formato pedido para a função `DecomposicaoLU`.
- Submeta suas funções a vários casos de teste e compare com os resultados obtidos usando funções e comandos prontos do MATLAB. Devido a imprecisões numéricas, os resultados podem diferir um pouco, porém espera-se que as diferenças sejam bem pequenas.
- Os seguintes comandos provêm formas convenientes de comparar matrizes e vetores no MATLAB:
 - `max(max(abs(A2-A1)))`, com $A1$ e $A2$ matrizes.
 - `max(abs(b2-b1))`, com $b1$ e $b2$ vetores.
- A ordem de implementação das funções foi pensada de modo que a implementação de uma certa função possa se beneficiar de funções já implementadas. Aproveite isso para poupar trabalho!