Trabalho de Algoritmos Numéricos DI/CT/UFES/2018-2 Ciencia da Computação, Engenharia da Computação e Engenharia Elétrica.

Comparação das soluções e do esforço computacional na resolução de um sistema de equações lineares via método de eliminação de Gauss e via método iterativo de Gauss Seidel para o caso de um problema "pentadiagonal".

Datas de entrega:
do código: até 4/11/2018; do relatório até dia 6/11/2018.

1 Objetivos

Implementar o método de eliminação de Gauss e o método iterativo de Gauss Seidel para resolver um sistema linear pentadiagonal levando em consideração a particularidade do sistema. Comparar os solucões obtidas e o esforço (quantidade de operações efetuadas) das duas versões.

2 Sistemas lineares do tipo pentadiagonal

Matrizes pentadigonais são aquelas cujos elementos se concentram em uma faixa central da matriz, tendo elementos não nulos nas cinco diagonais centrais e todo o restante nulo. Dito de outra forma, matrizes pentadigonais são matrizes tais que os únicos elementos não nulos são:

$$a_{i,j} \neq 0 \;\text{ se }\; |i-j| \leq 2$$

Um sistema linear pentadiagonal que aparece, algumas vezes, em aplicações numéricas, é aquele onde os elementos das diagonais, tanto na diagonal principal, como nas demais subdiagonais adjacentes (superiores e inferiores à diagonal principal) são constantes. Este caso é descrito por,

$$\begin{array}{l}
a_{i,j} = d_A \;\text{ se }\; (i-j) = 2 \\
a_{i,j} = d_B \;\text{ se }\; (i-j) = 1 \\
a_{i,j} = D_P \;\text{ se }\; i = j \\
a_{i,j} = d_C \;\text{ se }\; (i-j) = (-2) \\
a_{i,j} = d_D \;\text{ se }\; (i-j) = (-1)
\end{array}$$

Um exemplo, de matriz pentadiagonal, de dimensão $n = 8$, que satisfaz às condições descritas acima, é mostrada abaixo. Neste caso, $d_A = 0.5$, $d_B = -2.0$, $D_P = 9.5$, $d_C = -2.4$ e $d_D = 0.8$.

$$A = \begin{bmatrix}
9.5 & -2.4 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\
-2.0 & 9.5 & -2.4 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 \\
0.5 & -2.0 & 9.5 & -2.4 & 0.8 & 0.0 & 0.0 & 0.0 \\
0.0 & 0.5 & -2.0 & 9.5 & -2.4 & 0.8 & 0.0 & 0.0 \\
0.0 & 0.0 & 0.5 & -2.0 & 9.5 & -2.4 & 0.8 & 0.0 \\
0.0 & 0.0 & 0.0 & 0.5 & -2.0 & 9.5 & -2.4 & 0.8 \\
0.0 & 0.0 & 0.0 & 0.0 & 0.5 & -2.0 & 9.5 & -2.4 \\
0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.5 & -2.0 & 9.5
\end{bmatrix}$$

A aplicação do método de eliminação de Gauss em sistemas lineares cuja estrutura da matriz é particular pode ser feita levando-se em consideração esta especificidade.

No caso pentadiagonal, à medida que as etapas vão sendo realizadas, há posições, inicialmente nulas, que se tornam não nulas. Ou seja, à medida que se efetua a triangularização, algumas "novas diagonais" (adjacentes às já existentes) passam a ter elementos não nulos. Este preenchimento é conhecido como *fill-in*. É possível prever que posições são preenchidas.

Assim, a implementação do método para o caso pentadiagonal deve ser construída para operar/acessar **apenas** os elementos que são **não nulos** ou que permanecem não nulos ao longo da triangularização. Um algoritmo bem projetado implicará em uma redução de operações em relação ao caso onde a matriz é cheia. Quando n é grande, a redução no número de operações, em relação ao caso geral (matriz cheia) pode ser bastante significativa.

A implementação do método iterativo de Seidel em sistemas lineares cuja estrutura da matriz é pentadiagonal deve também ser feita levando-se em consideração a especificidade.

3 Implementações (valor 6,0 pts)

(A) Implementar o método de eliminação de Gauss com pivoteamento para resolver um sistema linear tal como descrito na sessão anterior. A implementação deve ser tal que leve em consideração a especificidade do problema, isto é, que manipule/acesse **somente** as posições onde há elementos não nulos.

(B) Implementar o método iterativo de Gauss Seidel levando em consideração que a matriz emvolvida é pentadiagonal. Tome como vetor inicial o vetor $x^{(0)}(i) = b(i)/a(i,i)$ e tolerância $tol = 10^{(-10)}$.

Incluir, em cada código, um contador que grave o número de operações realizadas. Por exemplo, no código abaixo, que obtem o produto interno de dois vetores X*Y, há uma váriavel do tipo inteira, chamada contador, que registra quantas operações (neste caso, somas e produtos) o código faz.

```
Calcula o produto interno de X com Y
INICIO
     Ler(n, X, Y)
     contador=0
     soma=0
     Para i de 1 até n, (passo 1)
         soma = soma + X(i)*Y(i)
         contador = contador + 2
     Fim {para i}
FIM
```

Os códigos (A) e (B) devem estar em um mesmo programa, na sequência um do outro. Como se quer comparar a quantidade de operações realizadas por cada método, é necessário gravar o esforço dispendido (a quantidade de operações) para cada caso.

Para o método iterativo de Gauss Seidel, além da quantidade de operações realizadas, grave também o número de iterações necessárias para atingir a dada tolerância.

Para que se possa de calcular o erro da solução obtida, um sistema com solução conhecida será gerado. Neste trabalho o vetor b deve ser gerado imaginando que a solução seja o vetor $x = [1.0, 1.0, ..., 1.0]$, ou seja, basta fazer o produto Ax para obter um vetor b.

4 Menus de entrada e valores de saída

4.1 Entrada

Os valores de entrada são: n, d_A, d_B, D_P, d_C e d_D e o vetor b (este gerado no código).

Seu programa deve exibir, na tela, um menu para que o usuário forneça os dados de entrada, que devem ser fornecidos na seguinte ordem:

1. inicialmente, deve ser solicitado/lido a dimensão do sistema (n).
2. Em seguida, os coeficientes da matriz, adotando a seguinte ordem: d_A, d_B, D_P, d_C e d_D.

4.2 Saída

Os valores de saída, para cada um dos métodos de resolução, devem ser:

1. o número de operações total
2. os valores de:
$$||e||_{max} = e_{Max} = max(|e(i)|)$$
e
$$e_{Medio} = (\sum_{1}^{n} |e(i)|)/n$$
onde e é o vetor erro. Lembrando que a solução exata x_{ex} esperada é o vetor unitário (isto é: $x_{ex}(i) = 1.0$) Assim, o erro de cada solução é dado por :
$$e(i) = |x_{obtido}(i) - 1.0|$$

5 Execuções

1. Rode o seu código para seguintes valores: $n = 20$, $d_A = 2.0$, $d_B = 1.0$, $D_P = 2.2$, $d_C = 0.1$ e $d_D = 0.2$, ou seja, para um dado problema linear $Ax = b$, de dimensão $n = 20$. Grave a solução obtida e o número de operações realizadas por cada método.
2. Rode o seu código para seguintes valores: $d_A = 2.0$, $d_B = 1.0$, $D_P = 2.2$, $d_C = 0.1$ e $d_D = 0.2$, com dimensões variadas, cada vez maiores (com $n = 40$, $n = 80$, $n = 160$, $n = 320$, $n = 640$, $n = 1280$ e $n = 2560$). Grave os dados de saída para cada dimensão.

3. Refaça o que foi descrito acima para os valores $d_A = 1.0$, $d_B = 1.0$, $D_P = 4.5$, $d_C = 1.0$ e $d_B = 1.0$, com dimensões variadas, cada vez maiores (com $n = 40$, $n = 80$, $n = 160$, $n = 320$, $n = 640$, $n = 1280$ e $n = 2560$). Grave os dados de saída para cada dimensão.

4. Refaça o mesmo que foi descrito acima para valores de d_A, d_B, D_P, d_C e d_D de sua escolha.

6 Relatório (valor 4,0 pts)

1. Introdução: apresentar uma síntese do trabalho (o que se propõe, o que foi feito, objetivos gerais).

2. Método numérico 1: explicar a ideia da implementação específica do método de eliminação de Gauss projetada para tratar o sistema particular (pentadiagonal).

3. Método numérico 2: explicar a estratégia da implementação do método de Gauss Seidel para tratar o sistema pentadiagonal.

4. Apresentar os resultados para o problema linear descrito no item 1 da seção de execuções. Mostrar também a matriz A e o vetor b.

5. Apresentar os resultados (através de uma tabela) para os problemas lineares descritos no item 2 da seção de execuções. A tabela deve exibir, para cada dimensão (n): o esforço total (quantidade de operações), os valores e_{Max} e $e_{Medio} = (\sum_1^n |e(i)|)/n$, para cada método.

6. Apresentar os resultados (através de uma tabela) para os problemas lineares descritos nos ítens 3 e 4 da seção de execuções. A tabela deve exibir, para cada dimensão (n): o esforço total (quantidade de operações), os valores e_{Max} e $e_{Medio} = (\sum_1^n |e(i)|)/n$, para cada método.

7. Para os problemas lineares descritos no item 2 da seção de execuções, traçar em um par de eixos cartesianos, para cada método, o esforço total (a quantidade de operações) versus a dimensão.

8. Código: imprimir, no final do relatório, o código.

7 Condições de entrega

7.1 Grupo:

Este trabalho deverá ser realizado em grupos de no máximo **2 alunos**. Este trabalho deve ser feito pelo grupo. Por favor, não solicitar que outros o façam. Pode ser feito em C ou em python.

7.2 Como entregar:

- Envie o código até dia **04/11/2018, às 23:59h**.
- Envie o código fonte do seu trabalho por e-mail para *galarda@inf.ufes.br*.
- O assunto do e-mail deverá ser o seguinte (somente o que está entre aspas duplas): "an:trab1:nome1:nome2". Substitua nome1 e nome2 pelo primeiro nome e o último sobrenome de cada integrante do grupo, separados por espaços.
- Envie o arquivo o código fonte (em C ou py) anexado. NÃO coloque o seu código no corpo do e-mail.
- O relatório deverá ser entregue impresso. Deve ser entregue até terça feira, dia, 06/11/2018, até às 17:00h.

Um exemplo: envio do trabalho, com um grupo com 2 integrantes:

```
Para: galarda@inf.ufes.br
De: Ana Silva
Assunto: an:trab1:Ana Silva:Antonio Santos:
Anexo: AnaAntonio.c
```