COBENGE 2003

APLICAÇÕES DO MATLAB NO ENSINO DE DISCIPLINAS BÁSICAS NOS CURSOS DE ENGENHARIA

Viviana Cocco Mariani - mariani@rla01.pucpr.br
Pontifícia Universidade Católica do Paraná, Departamento de Engenharia Mecânica
Rua Imaculada Conceição, 1155, Prado Velho
81611-970 - Curitiba, PR, Brasil

Emerson Martim – martim@rla01.pucpr.br
Pontifícia Universidade Católica do Paraná, Departamento de Engenharia Química
Rua Imaculada Conceição, 1155, Prado Velho
81611-970 - Curitiba, PR, Brasil

Resumo: *As aulas de laboratório computacional têm como característica principal à participação efetiva dos alunos na parte prática, o que se reflete no processo de ensino-aprendizagem. Entretanto, deve-se ter cuidado para que nas aulas de laboratório computacional os alunos não tenham a simples incumbência de averiguar a parte teórica que é apresentada em sala de aula, mas sim que participem ativamente construindo novas conjecturas, e explorando sua criatividade no entendimento físico e na resolução de problemas aplicados. Com este objetivo o presente artigo visa relatar as experiências vivenciadas pelos autores nas aulas teóricas e de laboratório computacional, especificamente ao ministrar os conteúdos solução de sistemas de equações lineares, cálculo de autovalores e cálculo de autovetores. Neste trabalho o aplicativo Matlab foi escolhido para este fim porque é um sistema interativo, onde desenvolver algoritmos na sua linguagem de programação é uma tarefa rápida e eficiente.*

Palavras-chave: *Ensino nas engenharias, Matlab, Autovalores, Autovetores, Sistemas de equações lineares.*

1. INTRODUÇÃO

Quando os problemas físicos são modelados matematicamente alguns destes problemas são descritos por sistemas de equações lineares, ou seja, a resolução de sistemas de equações lineares surge nas mais diversas áreas, ou dependem do cálculo de autovalores e autovetores.

O presente artigo relata as experiências vivenciadas pela autora em disciplinas como Cálculo Numérico, Métodos Numéricos e Álgebra Linear nos cursos de Engenharia da Pontifícia Universidade Católica do Paraná, quando os conteúdos relacionados à resolução de equações lineares, autovalores e autovetores foram abordados nestas disciplinas.

Nestas disciplinas, primeiramente a parte teórica é apresentada em sala de aula, e em seguida uma lista de exercícios aplicados as suas áreas de formação são expostos aos alunos. Numa primeira etapa os alunos em sala de aula resolvem alguns dos problemas manualmente e/ou utilizando calculadora programável e em seguida são encaminhados para a aula de laboratório computacional onde resolvem os demais problemas presentes na listagem com o auxílio do Matlab.

Os sistemas de computação algébrica e simbólica estão sendo cada vez mais utilizados para resolver problemas nas áreas de engenharia, devido à característica de combinar diferentes ferramentas em um único *software*. Entre estes aplicativos está o Matlab que é um ambiente computacional integrado de modelagem de sistemas e configuração apropriada e robusta de algoritmos. Além disso, o Matlab pode ser usado eficientemente para a implementação de projetos complexos, e por esta razão é adotado como ferramenta padrão em várias universidades.

Alguns dos problemas propostos durante as aulas são citados e descritos no presente artigo e resolvidos com o auxílio do Matlab. Com o uso deste aplicativo, pode-se evitar o desperdício de tempo com tarefas manuais exaustivas, fazendo com que o aluno se prenda mais ao entendimento físico dos problemas. O registro da aprendizagem se dá através da elaboração de um relatório.

Na próxima seção deste trabalho define-se brevemente um sistema de equações lineares, sua classificação a respeito da solução e alguns dos métodos numéricos que podem ser empregados para a sua resolução e define-se autovalor e autovetor. Logo após são apresentados os principais comandos do Matlab para resolução de matrizes e obtenção de autovalores e autovetores. Os métodos programados no Matlab para a resolução dos sistemas de equações lineares também são descritos. Na seção três, os problemas são resolvidos com o aplicativo Matlab. É relevante mencionar que os procedimentos utilizados do Matlab podem ser aplicados em diferentes problemas de Engenharia, não importando a área específica.

2. FUNDAMENTOS

Um sistema linear com m equações e n incógnitas (ou variáveis) é escrito usualmente na forma,

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n = b_m \end{cases}, \tag{1}$$

onde a_{ij} são os coeficientes ($1 \leq i \leq m$, $1 \leq j \leq n$), x_j são as variáveis (ou incógnitas) ($1 \leq j \leq n$) e bi são as constantes do vetor de termos independentes ($1 \leq i \leq m$).

A resolução de um sistema linear Ax = b consiste em calcular os valores de x_j (j = 1, ..., n), caso eles existam, que satisfaçam as m equações simultaneamente. Usando notação matricial, o sistema linear pode ser representado por Ax = b.

A respeito das soluções de um sistema linear de equações, o sistema pode ser impossível (ou incompatível) se não possui solução, possível (ou compatível) se possui solução. Caso o sistema seja possível então ele pode ser determinado se apresentar solução única ou indeterminado se possuir infinitas soluções (Faires e Burden, 2000).

A resolução de do sistema linear Ax = b nos faz indagar sobre as seguintes questões (Ruggiero e Lopes, 1996):

i) Existe solução $\bar{x} \in IR^n$ tal que $A\bar{x} = b$?
ii) Se existir $\bar{x}$, $\bar{x}$ é único?
iii) Como obter $\bar{x}$.

No presente trabalho estamos preocupados em responder ao item (i) e (ii), não obstante o outro item também foi abordado em sala de aula, contudo é omitido aqui por não fazer parte do objetivo principal deste trabalho.

Para a resolução dos sistemas lineares são apresentados os métodos diretos (Eliminação de Gauss, método de Gauss-Jordan, inversão de matriz, fatoração LU e fatoração de Cholesky) e os métodos iterativos (Jacobi e Gauss-Seidel).

Os métodos diretos são aqueles que, a menos de erros de arredondamento, fornecem a solução exata do sistema linear, caso ela exista, após um número finito de operações. Os métodos iterativos geram uma seqüência de vetores x_k, a partir de uma aproximação inicial x_0. Sob certas condições esta seqüência converge para a solução $\bar{x}$ caso ela exista (Barroso *et al*., 1987).

Autovalores e autovetores estão presentes em diferentes ramos da matemática incluindo formas quadráticas, sistemas diferenciais, problemas de otimização não linear, e podem ser usados para resolver problemas de diversos campos, como economia, teoria da informação, análise estrutural, eletrônica, teoria de controle e muitos outros.

Uma transformação linear T de um espaço vetorial V em um espaço vetorial U, T: V $\rightarrow$ U, é uma correspondência que associa a cada vetor $\vec{v}$ de V um vetor $T(\vec{v})$ em U de modo que:

$$T(a\vec{u} + b\vec{v}) = aT(\vec{u}) + bT(\vec{v}) \text{ para } \forall\ \vec{u}, \vec{v} \in V \text{ e } \forall\ a, b \in IR. \qquad (2)$$

Em particular se T: V $\rightarrow$ V então dizemos que T é um operador linear num espaço vetorial V. Seja T: V $\rightarrow$ V um operador linear, um escalar $\lambda \in IR$ é um autovalor (ou valor próprio) de T se existe um vetor não nulo $\vec{v}\,(\vec{v} \neq 0) \in V$ tal que: $T(\vec{v}) = \lambda\vec{v}$. Todo vetor $\vec{v}$ satisfazendo essa relação é um autovetor (ou vetor próprio) de T correspondente ao autovalor λ. Se A é uma matriz quadrada, de ordem n, então $\lambda \in IR$ é um autovalor de A se existe um vetor não nulo $\vec{v}\,(\vec{v} \neq 0) \in V$ tal que: $A\vec{v} = \lambda\vec{v}$. Considere uma matriz quadrada A_{nxn}, sobre um corpo K, isto é:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \qquad (3)$$

A matriz A - λI, ou seja,

$$A - \lambda I = \begin{pmatrix} a_{11} - \lambda & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} - \lambda & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} - \lambda \end{pmatrix} \quad (4)$$

onde I é a matriz identidade de ordem n e λ é um parâmetro, é chamada matriz característica de A. Seu determinante, |A-λI|, que é um polinômio de grau n em λ é chamado polinômio característico de A. Encontrando as raízes deste polinômio, isto é, fazendo det(A- λI) = 0 obtemos os autovalores da matriz A.

3. PROCEDIMENTOS NO MATLAB

O Matlab tem encontrado vasta aplicabilidade na Engenharia, tanto do ponto de vista técnico como educacional. Os diversos recursos disponíveis no Matlab podem ser utilizados na resolução de problemas em diferentes áreas, além de permitir ao professor transmitir aos alunos tópicos da disciplina complexos de serem ministrados em sala de aula com quadro e giz (Mariani, 2002).

A seguir são apresentados alguns comandos básicos do Matlab para trabalhar com matrizes. Os alunos no laboratório computacional são instigados primeiramente a explorar estes comandos. Depois desta etapa os alunos empregam os métodos programados no Matlab para a resolução dos sistemas de equações lineares. Os métodos diretos fatoração LU, fatoração de Cholesky e inversão de matrizes já estão pré-definidos no Matlab, enquanto os métodos Eliminação de Gauss, Jacobi e Gauss-Seidel são programados no Matlab e utilizados pelos alunos.

3.1. Alguns Comandos Básicos no Matlab

São mostrados a seguir alguns comandos básicos para se trabalhar com matrizes no Matlab, estes comandos foram executados no laboratório computacional na seqüência apresentada (The Math Works Inc., 1997).

a) Atribuir valores para uma matriz A e um vetor b: Ex: A=[2 3 4;1 2 3;5 -4 3], b=[1;-2;3]
b) Calcular a transposta B de uma matriz A: B = A'
c) Calcular o determinante de uma matriz A: det(A)
d) Criar uma matriz identidade, por exemplo, de ordem 3: eye (3)
e) Calcular a inversa de uma matriz A: inv(A)
f) Calcular o produto entre duas matrizes: A*B
g) Resolver o sistema de equações lineares de forma direta pelo Matlab:
    g.1) x = A\b   (divisão esquerda, equivale a $x = A^{-1}$*b
    g.2) x = inv(A)*b
    g.3) x = linsolve(A,b)

h) A fatoração LU já é uma função pré-definida no Matlab para tal basta digitar a matriz (A) representante do sistema e fazer [L U] = lu(A). Após obter a fatoração resolve-se os sistemas Ly = b ⇒ y = L\b e Ux = y ⇒ x = U\y, por qualquer uma das formas apresentadas no item g.

i) A fatoração de Cholesky é obtida por U = chol(A), isto é, obtém-se a matriz triangular superior através deste comando. Utilizar o seguinte sistema na forma matricial para obter a fatoração de Cholesky. Vale ressaltar que a fatoração de Cholesky só é possível para matriz simétrica positiva definida. Obtida a matriz U resolver $U^T y = b$ (substituição progressiva) e depois $Ux = y$ (por substituição retroativa) de acordo com o item g.
j) Para calcular os autovalores da matriz quadrada A, basta de dentro da área de trabalho do Matlab, fazer a chamada à rotina eig do Matlab: b = eig(A)
k) O comando que segue forma duas matrizes, C uma matriz cujas colunas são os autovetores de A, e D é uma matriz diagonal contendo os autovalores na diagonal: [C,D]=eig(A)
l) A função poly calcula diretamente o polinômio característico de uma matriz A, isto é, retorna um vetor que contém os coeficientes do polinômio característico de A: e = poly(A)
m) As raízes do polinômio são os autovalores, para obtê-los basta usar o comando roots que retorna um vetor coluna contendo os autovalores: roots(e).

3.2. Métodos Programados no Matlab para Resolver Sistemas Lineares

As "Figuras 1 e 2" ilustram o código computacional de três programas em Matlab, de acordo com os algoritmos dos métodos de Eliminação de Gauss, Jacobi e Gauss-Seidel, respectivamente.

O método de Eliminação de Gauss, apresentado na "Figura 1", pode ser utilizado digitando na linha de comando do Matlab a função EliGauss(Ab), desde que a matriz Ab (matriz estendida, formada pelos coeficientes da matriz A mais o termo independente b) esteja disponível no ambiente do Matlab e $\det(A) \neq 0$. No método de Jacobi escreve-se a função Jacobi(A,b,x0,tol) na linha de comando do Matlab, desde que se tenha valores para A, b, x0 e tol. O script é executado pelo comando.

```
function [x,iter] = EliGauss (Ab)
n_l = size(Ab,1);     % no. de linhas da matriz
estendida Ab
n_c = size(Ab,2)     % no. de colunas da
matriz estendida Ab
for k=1:(n_l-1)
  for i=(k+1):n_l
    m = -Ab(i,k)/Ab(k,k)
    Ab(i,k) = 0
    for j=(k+1):n_c
      Ab(i,j) = Ab(i,j) + m*Ab(k,j)
    end
  end
end
x(n_l) = Ab(n_l,n_c)/Ab(n_l,n_l);
for i=(n_l-1):-1:1
   s = 0
for j=i+1:n_l
      s = s + Ab(i,j)*x(j)
   end
   x(i) = (Ab(i,n_c)-s)/Ab(i,i)
End
```

```
function [x,iter]=Jacobi(A,b,x0,tol)
L=-tril(A,-1); % extrai a parte triangular
inferior da matriz
U=-triu(A,1);  % extrai a parte triangular
superior da matriz
D=diag(diag(A));  % extrai os elementos da
diagonal principal da matriz
N=L+U;
C=D\N;
G=D\b;

iter=1;

x = C*x0 + G;
while norm(x-x0)>tol*norm(x0)
   x0 = x;
    x = C*x0 + G;
 iter = iter + 1;
 End
```

Figura 1 - Método de Eliminação de Gauss e método de Jacobi.

Para usar o método de Gauss-Seidel, ilustrado na "Figura 2", de forma análoga ao método de Jacobi digita-se a função Gauss(A,b,x0,tol) desde que se tenha valores para A, b, x0 e tol. Vale ressaltar que para usar estes métodos a matriz deve ser diagonalmente dominante (matriz cujos coeficientes da diagonal principal em módulo devem ser maior que a soma dos módulos dos coeficientes não pertencentes a diagonal principal), caso contrário os métodos falham (Ruggiero e Lopes, 1996).

```
function [x,iter]=gauss(A,b,x0,tol)
L = tril(A,-1); % extrai a parte triangular
inferior da matriz
U = -triu(A,1);  % extrai a parte triangular
superior da matriz
D = diag(diag(A));  % extrai os elementos da
diagonal principal da matriz
M = D+L;
C = M\U;
G = M\b;
iter = 1;
x = C*x0 + G;
while norm(x-x0)>tol*norm(x0)
   x0   = x;
   x    = C*x0 + G;
   iter = iter+1; end
```

Figura 2 - Método de Gauss-Seidel.

4. APLICAÇÕES

Em sala de aula são resolvidos vários problemas aplicados, a seguir três destes são descritos. Os alunos ficam motivados e engajados no processo de ensino-aprendizagem quando situações problemas reais envolvendo os tópicos das disciplinas são apresentados, é notável que o rendimento dos alunos aumenta. Os alunos também são instigados a buscar novos problemas para serem resolvidos em sala de aula utilizando os conteúdos abordados.

Problema 1 - Considere a reação química:

$$x_1PbN_6 + x_2CrMn_2O_8 \rightarrow x_3Pb_3O_4 + x_4Cr_2O_3 + x_5MnO_2 + x_6NO \qquad (5)$$

para cada um dos reagentes e cada um dos produtos, obtenha um vetor do R^5 que contenha o número de átomos por moléculas, para chumbo, nitrogênio, cromo manganês, e oxigênio. Por exemplo, o vetor para o permanganato de cromo ($CrMn_2O_8$) é (0,0,1,2,8). Seja $x_1,....,x_6$ o número de moléculas de cada tipo que são necessárias para balancear a "equação (5)". Escreva uma equação vetorial a ser satisfeita por essas variáveis. Passando todas as variáveis para a esquerda, reescreva a equação vetorial da forma Ax = 0 e resolva-a. Existem infinitas soluções matemáticas, encontre aquela que tenha mais sentido quimicamente. Se possível, use uma representação racional ou aritmética exata para os cálculos.

Solução: O sistema resultante da reação química é:

$$\begin{bmatrix} 1 & 0 & -3 & 0 & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & -2 & 0 & 0 \\ 0 & 2 & 0 & 0 & -1 & 0 \\ 0 & 8 & -4 & -3 & -2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6}$$

A matriz não é quadrada logo o sistema apresenta uma variável livre. Neste caso, escolheu-se x_1 como esta variável, atribuindo o valor unitário obtém-se usando a decomposição LU as seguintes matrizes [L U] = lu(A).

$$L = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1/8 & -1/6 & 1 & 0 & 0 \\ 1/4 & -1/3 & -6/13 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \text{ e}$$

$$U = \begin{bmatrix} 8 & 4 & -3 & -2 & -1 \\ 0 & -3 & 0 & 0 & 0 \\ 0 & 0 & 13/8 & 1/4 & 1/8 \\ 0 & 0 & 0 & -5/13 & 4/13 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{7}$$

Resolvendo o sistema triangular inferior (y=L\b) e a seguir o sistema triangular superior (x=U\y) obtém-se o vetor intermediário y e o vetor solução x,

$$y = \begin{bmatrix} 0 \\ -1 \\ -1/6 \\ -16/39 \\ -6 \end{bmatrix} \text{ e}$$

$$x = \begin{bmatrix} 44/15 \\ 1/3 \\ 22/15 \\ 88/15 \\ 6 \end{bmatrix} \tag{8}$$

Logo o vetor solução, incluindo a variável x_1 é $[1 \;\; 44/15 \;\; 1/3 \;\; 22/15 \;\; 88/15 \;\; 6]^T$. Transformando as dízimas periódicas em números inteiros obtém-se a seguinte solução para o sistema linear: $[15 \; 44 \; 5 \; 22 \; 88 \; 90]^T$. Portanto a equação balanceada fica:

$15PbN_6 + 44CrMn_2O_8 \rightarrow 5Pb_3O_4 + 22Cr_2O_3 + 88MnO_2 + 90NO.$

Problema 2 - Uma consideração importante no estudo de transferência de calor é a de se determinar a distribuição de temperatura assintótica de uma placa fina quando a temperatura em graus Celsius em suas bordas é conhecida. Considere que a "Figura 3" representa uma seção transversal de uma barra de metal, com fluxo de calor desprezível na direção perpendicular à placa. Sejam $T_1.....T_6$ as temperaturas em graus Celsius nos seis vértices interiores do reticulado desta figura. A temperatura num vértice é aproximadamente igual à média dos quatro vértices vizinhos mais próximos: à esquerda; à direita; acima e abaixo.

Por exemplo: $T_1 = (10 + 20 + T_2 + T_4)/4$. Escreva um sistema de equações cuja solução fornece as estimativas para as temperaturas de $T_1...T_6$. Resolva o sistema de equações resultante.

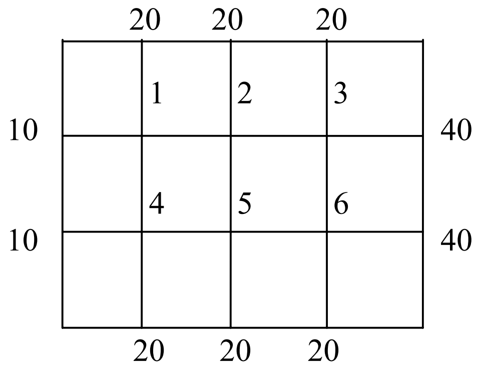

Figura 3 - Seção transversal de uma placa de metal

$$\begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 \\ -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & -1 & 0 & -1 & 4 \end{bmatrix} \tag{9}$$

A solução pelo Matlab com 3 dígitos: T_1 = 17,143ºC; T_2 = 21,429ºC ; T_3 = 27,143ºC ; T_4 = 17,143ºC; T_5 = 21,429ºC e T_6 = 27,143ºC.

5. CONCLUSÕES

Com o aumento das facilidades computacionais, os docentes não podem ignorar as influências e o impacto das ferramentas tecnológicas no processo de ensino-aprendizagem. Contudo a parcela de docentes que utilizam estas ferramentas ainda está longe de ser a adequada. De um modo geral, as facilidades e vantagens trazidas pelos aplicativos computacionais no ensino aumentam a motivação e o aprendizado, e conseqüentemente resultam num maior rendimento dos alunos.

Deve-se lembrar que os exemplos apresentados, neste artigo, utilizaram apenas uma pequena parcela dos recursos oferecidos pelo Matlab. Muitas outras ferramentas contidas neste aplicativo, diferentes das exploradas aqui, podem ser utilizadas em outras situações. Os procedimentos aqui apresentados e os códigos computacionais em Matlab podem ser utilizados nos mais diversos cursos, tais como Matemática, Engenharias e Ciência da Computação, para resolução de sistemas de equações lineares.

REFERÊNCIAS BIBLIOGRÁFICAS

BARROSO, C. L., BARROSO, M. M., FILHO, C. F. F., CARVALHO, M. L. B., **Cálculo Numérico - com Aplicações**, São Paulo, Harbra, 2ª. ed., 1987.

FAIRES, J. D.; BURDEN, R. L., **Numerical Analysis**, 7ª ed., Brooks/Cole Pub Co, 2000.

MARIANI, V. C., Laboratório Computacional na Disciplina de Cálculo Numérico - um Relato, In: XXX CONGRESSO BRASILEIRO DE ENSINO DE ENGENHARIA, **Anais**, Piracicaba, 2002.

RUGGIERO, M. A. G. e LOPES, V. L. R., **Cálculo Numérico - Aspectos Teóricos e Computacionais**, Rio de Janeiro, Makron Books, 2ª. ed., 1996.

MATLAB APPLICATIONS IN THE EDUCATION OF BASIC DISCIPLINES FOR ENGINEERING COURSES

Abstract: *Computational laboratory lessons have as main characteristic the participation of the students in the practical part of course. This important feature influences positively the teaching-learning procedure. However, care must be had so that the students do not have the simple incumbency to inquire the theoretical concepts presented in classroom, but yes that they participate actively constructing new conjectures, and exploring its creativity in the physical agreement and the resolution of applied problems. The present paper aims to comment the experiences lived for the authors in the computational laboratory and theoretical lessons, specifically when giving the contents solution of linear equations systems and eigenvectors and eigenvalues calculation. In this work the software Matlab was chosen for this purpose because it is an interactive system, where to develop algorithms in its programming language is a fast and efficient task.*

Key-words: *Engineering education, Matlab, Eigenvalues, Eigenvectors, Linear equations systems.*