MÉTODOS NUMÉRICOS - EXERCÍCIO V

SÉRGIO CORDEIRO

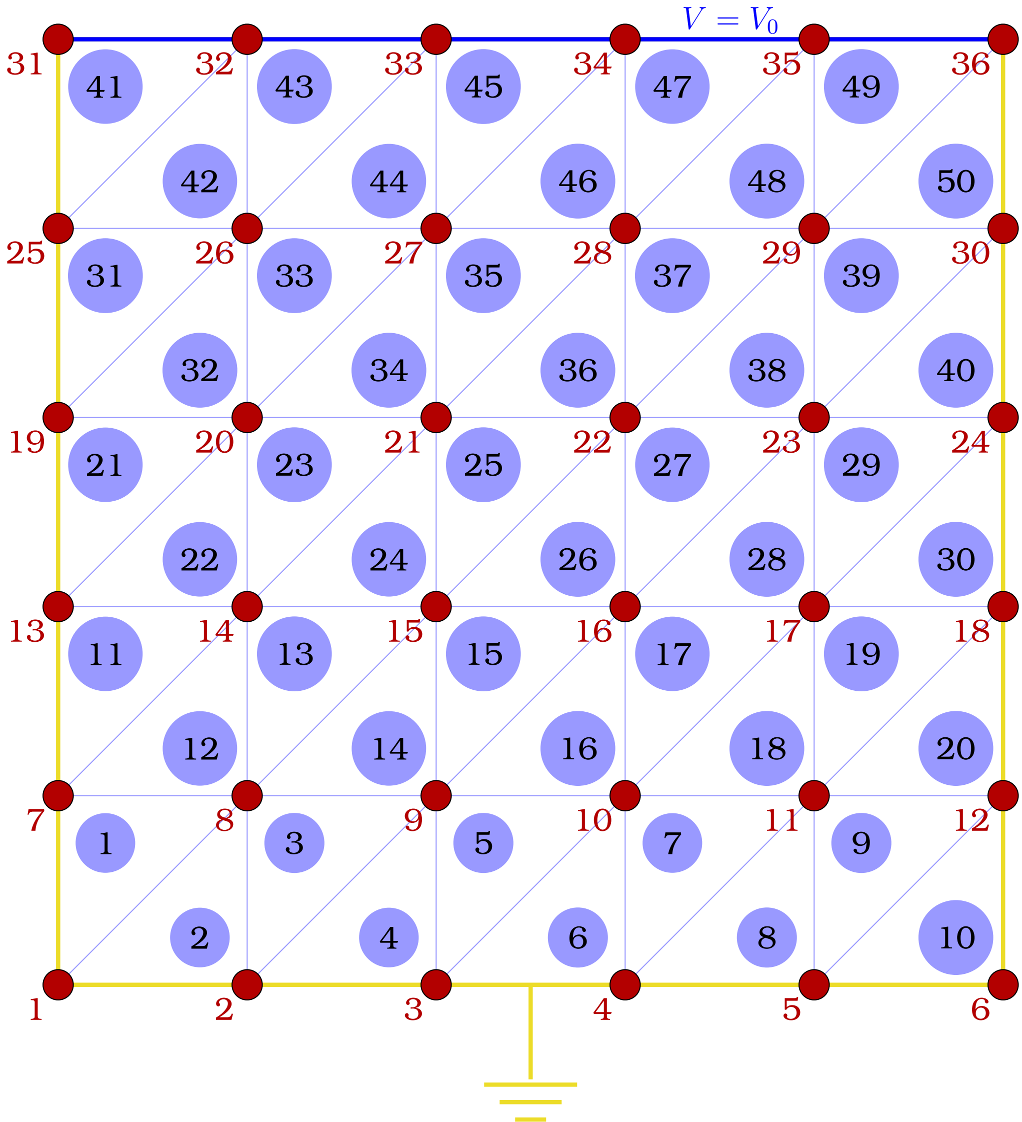
1. Resolver os problemas seguintes problemas do livro: "Elementos de eletromagnetismo-Sadiku":
a. FEM – 15.29 (Desenvolva seu próprio código). (métodos iterativos e matriz de banda).
b. FEM – 15.30
c. FEM – 15.31

Observações: - Todos os desenvolvimento numéricos podem ser feitos no MATLAB
- Apresente uma descrição teórica do problema
- Apresente o código
- Apresente comparações aumentando a malha de discretização
- Apresente conclusões detalhadas

a) O problema consiste em resolver a equação de Laplace em um espaço quadrado, cujos lados medem 1.0 m, e com um potencial V_0 de 100 V aplicado ao lado superior e os demais aterrados. Deve ser aplicado o método dos elementos finitos, com elementos triangulares. A solução exata é:

$$V(x,y) = \frac{4V_0}{\pi}\sum_{k=0}^{\infty}\frac{\sin(n\pi x)\sinh(n\pi y)}{n\sinh(n\pi)} \qquad n = 2k+1 \tag{1}$$

Um programa para solução por sistema linear foi fornecido, bem como a grade numerada abaixo:



No programa fornecido pela bibliografia, os dados devem ser lidos a partir de um arquivo. No nosso caso, para facilitar o recálculo com várias divisões da grade, tais dados são calculados a partir da geometria do problema por uma função de inicialização especial, aproveitando a simplicidade da geometria. A função que resolve o problema permaneceu completamente genérica. Construiu-se também uma função para saída de dados, que também é específica para a geometria dada. O programa de cálculo foi adaptado da bibliografia e poderia ainda ser melhorado. O código está listado abaixo.

LISTING 1. FEM_Tri_SolveG1.m

```
function [Vmap] = FEM_Tri_SolveG1(n, l)
% Resolve a equação de Laplace pelo método dos elementos finitos em um quadrado de lado
    'l', usando 'n' elementos triangulares.
% Os elementos são triângulos retângulos e formam uma grade retangular.
% Problema 15.29 do livro “Elementos de eletromagnetismo", de Matthew Sadiku
% Retorna um mapa do potencial em cada ponto da grade retangular.
  % Inicializa os dados para o método, de acordo com a geometria dada
  [map, XY] = FEM_Tri_SolveI(n, l);
  % Inicializa os pontos fixos
  nd = size(XY,1);
  Vp = zeros(nd, 1);        % V = 0 na base e nos lados
  Vp((nd-n):nd) = 100;      % V = 100 no topo
  % Calcula o potencial em cada nó
  V = FEM_Tri_SolveS(map, XY, Vp);
  % Exibe os resultados
  ref = @(i,j,h) FEM_Tri_SolveR1(i, j, h);
  Vmap = FEM_Tri_SolveP(V, ref, n, l/n);
end

function Vref = FEM_Tri_SolveR1(i, j, h)
% Calcula a distribuição ideal
  pih = pi * h;
  infty = 100;
  sum = 0;
  for k = 0:infty
    m = 2 * k + 1;
    sum = sum + sin(m * pih * (i-1)) * sinh(m * pih * (j-1)) / (m * sinh(m * pi));
  end
  Vref = 4 * 100 / pi * sum;
end
```

LISTING 2. FEM_Tri_SolveI.m

```
function [map, XY] = FEM_Tri_SolveI(n, l)
% Inicializa os dados para um quadrado de lado 'l' e 'n' divisões por dimensão
  % Cálculo das dimensões da grade
  nd = (n + 1) ^ 2;      % número de nós
  ne = 2 * n ^ 2;        % número de elementos
  np = 4 * n;          % número de nós fixos
  h = l / n;           % largura da grade retangular
  % Inicialização
  global de
  de = @(i,j) (j - 1) * (n + 1) + i;
  map = zeros(ne, 3);      % ... mapa dos nós pertencentes a cada elemento
  XY = zeros(nd, 3);       % ... coordenadas e tipos dos nós
  for i = 1:n+1
    for j = 1:n+1
      % ... para cada nó
      % ...... obtém os nós que limitam o quadrado adjacente
      ei = de(i, j);       % esquerda inferior
      es = de(i, j+1);     % esquerda superior
      di = de(i+1, j);     % direita inferior
      ds = de(i+1, j+1);     % direita superior
      if ((i <= n) && (j <= n))
        % ...... associa os nós aos elementos triangulares contidos no quadrado
        % ......... (sentido anti-horário)
        e = (j - 1) * 2 * n + 2 * (i - 1) + 1;
```

```
    map(e, :) = [ei, ds, es];
    map(e + 1, :) = [ei, di, ds];
   end
   % ...... verifica se o nó é fixo
   fixo = (j == 1) || (j == n+1) || (i == 1) || (i == n+1);
   id = ei * fixo;
   % ...... associa as coordenadas ao nó
   XY(ei, :) = [[i-1 j-1] * h, id];
  end
 end
end
```

LISTING 3. FEM_Tri_SolveS.m

```
function V = FEM_Tri_SolveS(map, XY, Vp)
% Resolve o problema pelo método dos elementos finitos
 [ne, nd, np] = deal(size(map,1), size(XY,1), size(Vp,1));
 b = zeros(nd, 1);
 C = zeros(nd, nd);
 for i = 1:ne
  % ... para cada elemento
  % ...... obtém os nós que o limitam e suas coordenadas
  d = map(i, :);
  xy = XY(d, :);
  % ...... calcula a matriz de coeficientes locais
  p = xy([2 3 1],2) - xy([3 1 2],2);
  q = xy([3 1 2],1) - xy([2 3 1],1);
  area = 2 * abs(p(2) * q(3) - q(2) * p(3));
  ce = (p * p' + q * q') / area;
  % ....... para cada nó
  for j = 1:3
   % ......... verifica se é nó fixo
   ir = d(j);
   p = xy(j,3);
   if (p > 0)
    % ........... nó fixo; potencial constante
    C(ir, ir) = 1;
    b(ir) = Vp(p);
   else
    % ........... nó livre
    for k = 1:3
     % ............... calcula a contribuição dos nós do elemento
     ic = d(k);
     p = xy(k,3);
     if (p > 0)
      % .................. constribuição de nó fixo
      b(ir) = b(ir) - ce(j,k) * Vp(p);
     else
      % .................. constribuição de nó livre
      C(ir, ic) = C(ir, ic) + ce(j, k);
     end
    end
   end
  end
 end
 % Resolve o sistema
 V = C \ b;
end
```

LISTING 4. FEM_Tri_SolveP.m

```
function Vmap = FEM_Tri_SolveP(V, ref, n, h)
  % Organiza os potenciais calculado e de referência em uma matriz
  Vmap = zeros(n+1,n+1);
  Vref = Vmap;
  global de
  for i = 1:n+1
    for j = 1:n+1
      ei = de(i, j);
      Vmap(i,j) = V(ei);
      Vref(i,j) = ref(i,j,h);
    end
  end
  % Plota a distribuição encontrada e a de referência
  ticks = linspace(1, n+1, 6);
  figure(1), contour(Vmap'), colorbar, title('Potencial (V)'); ax = gca; ax.XTick=ticks
      ; ax.YTick=ticks; grid on;
  figure(2), contour(Vref'), colorbar, title('Potencial (V)'); ax = gca; ax.XTick=ticks
      ; ax.YTick=ticks; grid on;
end
```

A distribuição de potencial obtida, bem como aquela calculada analiticamente, estão ilustradas abaixo:

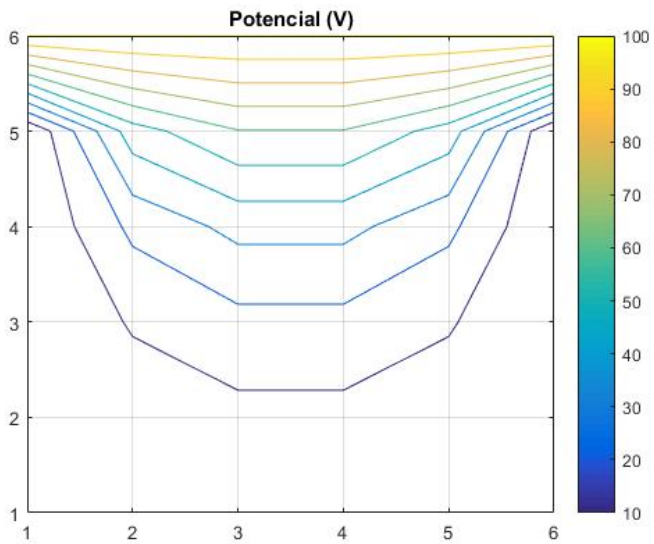


FIGURA 1. Distribuição calculada (grade com 5 intervalos)

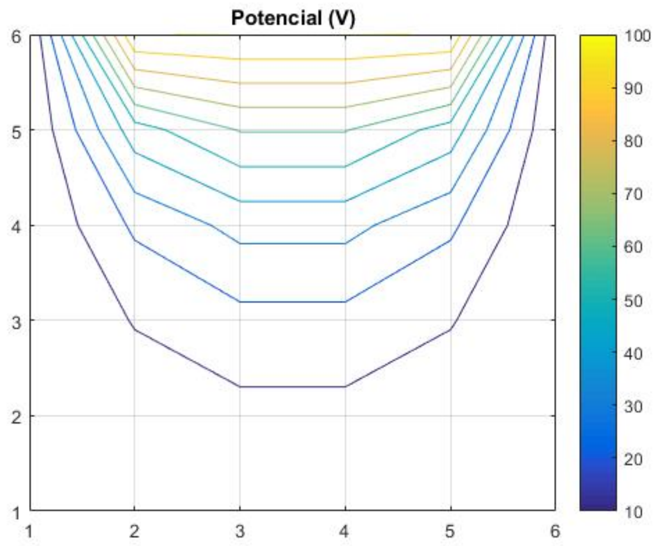


FIGURA 2. Distribuição teórica (grade com 5 intervalos)

Percebe-se que a solução não é satisfatória para os pontos onde o potencial varia abruptamente (cantos superiores da grade). A generalidade dos programas permite variar à vontade o tamanho da malha, o que melhora a precisão. Para n = 50, o resultado foi o seguinte:

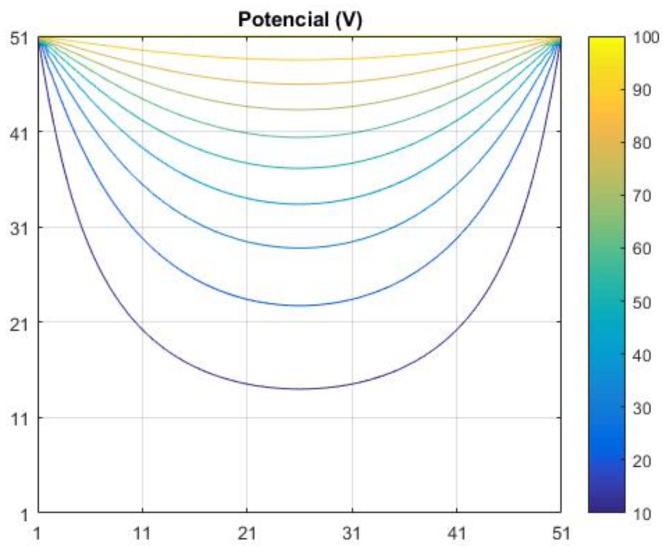


FIGURA 3. Distribuição calculada (grade com 50 intervalos)

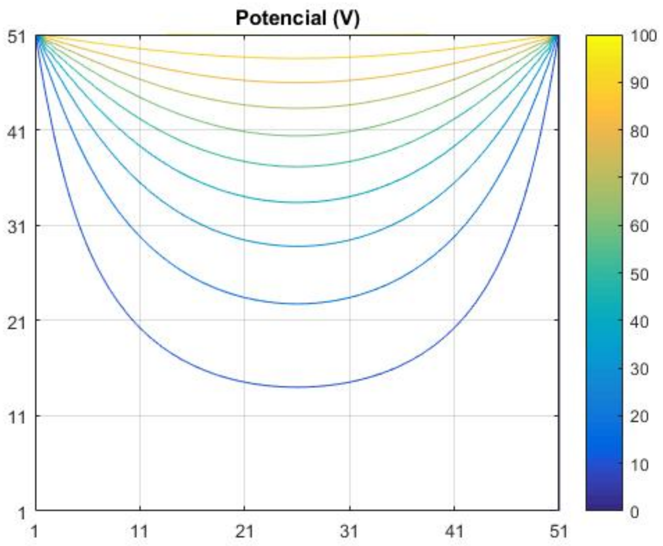


FIGURA 4. Distribuição teórica (grade com 50 intervalos)

Nota-se que as equipotenciais são mais lisas e que a solução é precisa nos cantos superiores.

---

b) O problema consiste em resolver o problema anterior com um potencial $V_0 = 100\sin(\pi x)$. A solução exata é:

$$V(x,y) = \frac{100\sin(\pi x)\sinh(\pi y)}{\sinh(\pi)} \quad (2)$$

Foi usado o mesmo código empregado no problema anterior, uma vez que a geometria é a mesma e muda apenas o valor do potencial em alguns pontos fixos. Essa mudança foi tratada por meio de uma mudança na função de controle geral da execução.

LISTING 5. FEM_Tri_SolveG2.m

```
function [Vmap] = FEM_Tri_SolveG2(n, l)
% Resolve a equação de Laplace pelo método dos elementos finitos em um quadrado de lado
    'l', usando 'n' elementos triangulares.
% Os elementos são triângulos retângulos e formam uma grade retangular.
% Problema 15.30 do livro "", de Matthew Sadiku
% Retorna um mapa do potencial em cada ponto da grade retangular.
  % Inicializa os dados para o método, de acordo com a geometria dada
  [map, XY] = FEM_Tri_SolveI(n, l);
  % Inicializa os pontos fixos
  nd = size(XY,1);
  h = l / n;
  Vp = zeros(nd, 1);              % V = 0 na base e nos lados
```

```
  Vp((nd-n):nd) = 100 * sin(pi * h * (0:n));  % V = 100 sin(pi x) no topo
  % Calcula o potencial em cada nó
  V = FEM_Tri_SolveS(map, XY, Vp);
  % Exibe os resultados
  ref = @(i,j,h) FEM_Tri_SolveR2(i, j, h);
  Vmap = FEM_Tri_SolveP(V, ref, n, h);
end

function Vref = FEM_Tri_SolveR2(i, j, h)
% Calcula a distribuição ideal
  pih = pi * h;
  Vref = 100 * sin(pih * (i-1)) * sinh(pih * (j-1)) / sinh(pi);
end
```

A distribuição de potencial obtida, bem como aquela calculada analiticamente, estão ilustradas abaixo:

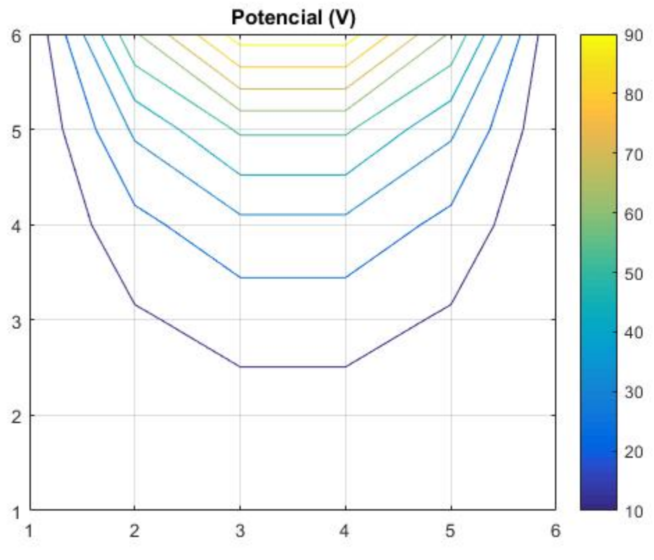


FIGURA 5. Distribuição calculada (grade com 5 intervalos)

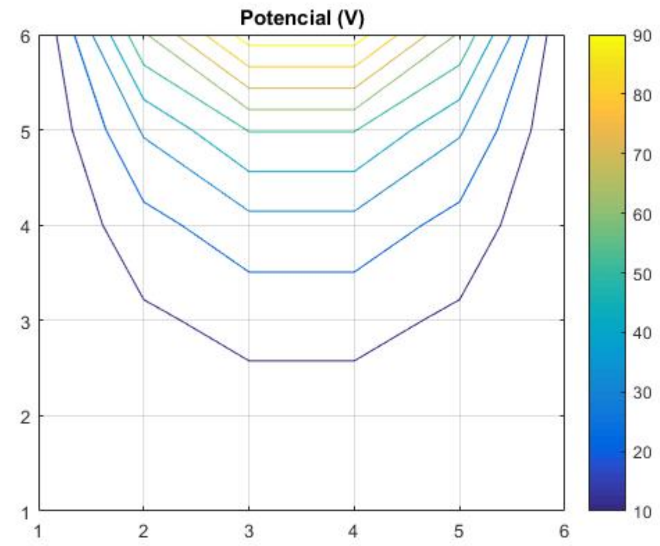


FIGURA 6. Distribuição teórica (grade com 5 intervalos)

Para n = 50, o resultado foi o seguinte:

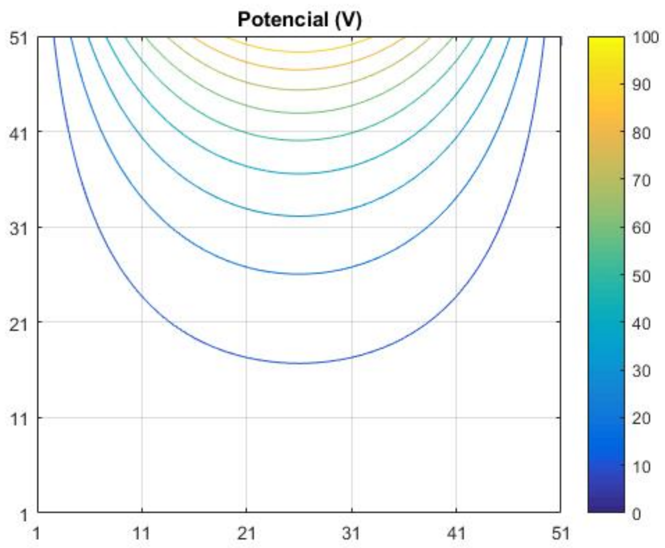


FIGURA 7. Distribuição calculada (grade com 50 intervalos)

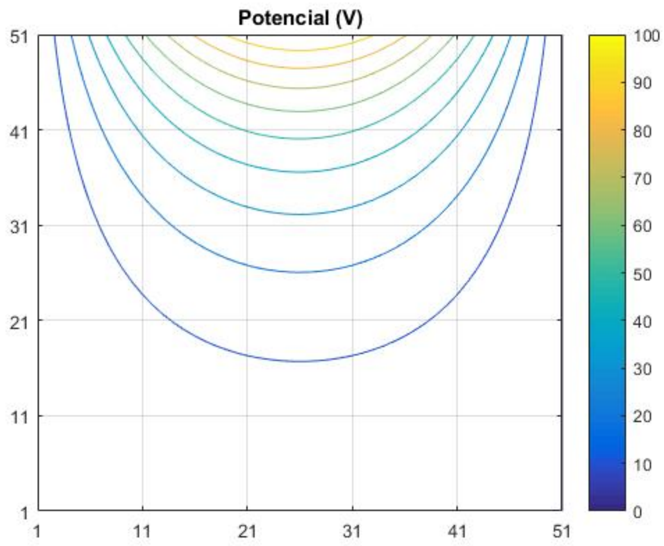


FIGURA 8. Distribuição teórica (grade com 50 intervalos)

Neste caso, a malha de 5 divisões também não se mostra fina o bastante mas, como o potencial nos cantos superiores não varia de forma abrupta, o erro não é tão grande. Um valor de n = 50 é mais do que suficiente para uma solução perfeita.

---

c) O problema consiste em demonstrar que, quando uma malha quadrada é usada no método das diferenças finitas, obtém-se a mesma matriz de coeficientes que quando se usa o método dos elementos finitos com os quadrados divididos ao meio, formando triângulos.
Pode-se notar facilmente que, em ambos os casos, os nós são exatamente os mesmos. Sendo assim, a matriz de coeficientes e o vetor correspondente à contribuição dos pontos fixos são idênticos nos dois casos, porque eles são determinados unicamente pela geometria do problema. Portanto, o sistema linear a ser resolvido:

$$\mathbf{Ax} = \mathbf{b}$$

é o mesmo para ambos os métodos. Pode-se concluir daí que o método dos elementos finitos é mais geral que o das diferenças finitas, e que em alguns casos ambos coincidem, inclusive em precisão e esforço computacional necessário.

Programas testados com **MATLAB** R2016a
`https://www.mathworks.com`
Texto formatado com **pdflatex** em ambiente **MiKTeX** 2.9:
`http://miktex.org/download/`