

**Universidade Federal de Santa Catarina – UFSC**  
**Centro Tecnológico – CTC**  
Departamento de Engenharia de Engenharia Elétrica - DEEL

**MARCO AURÉLIO DE OLIVEIRA MOREIRA**

**TRABALHO DE INTRODUÇÃO A MODELAGEM NUMÉRICA EM ELETROMAGNETISMO I**

**FLORIANÓPOLIS**

**2017**

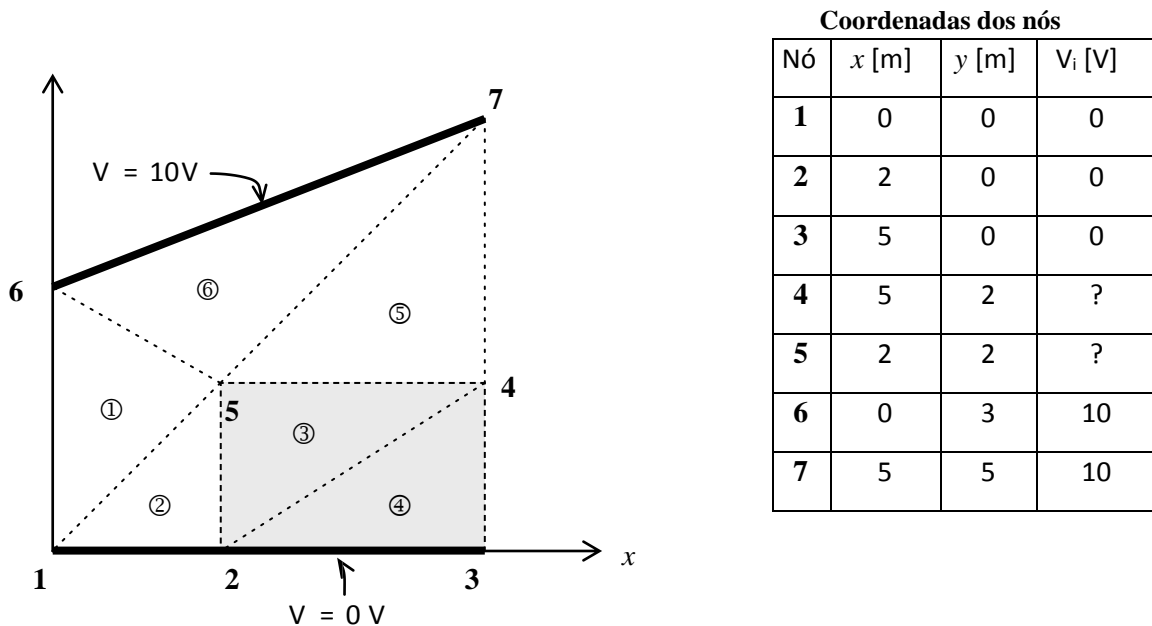
## Sumário

1. Trabalho nº 1.....	3
2. Resultados do Programa.....	3
3. Comparação de Resultados .....	5
4. Código do Programa em VBA.....	6

## 1. Trabalho nº 1 - Enunciado

No problema eletrostático abaixo, utilize o método de elementos finitos para obter:

- o potencial nos nós 4 e 5 e no ponto  $P$  (coordenadas  $x_P = 4,5$  e  $y_P = 1,0$  [m]);
- o campo elétrico em cada elemento (fazer um esboço dos vetores em cada elemento).



Permissividade dos elementos:

$$\epsilon_1 = \epsilon_2 = \epsilon_5 = \epsilon_6 = \epsilon_0$$

$$\epsilon_3 = \epsilon_4 = 2\epsilon_0$$

- Resolva o problema fazendo os cálculos manualmente.
- Escreva um programa de computador para resolver o problema acima. Utilize o mesmo programa para analisar o problema apresentado no artigo do Sadiku ("A simple introduction to finite element analysis of electromagnetic problems") e compare os resultados obtidos com os apresentados no referido artigo.

## 2. Resultados do Programa

Os resultados obtidos através do programa efetuado em Excel estão descritos na tabela abaixo. Os resultados dos cálculos feitos a mão para comparação serão entregues diretamente ao professor. No Excel há instruções do funcionamento do programa e como ele deve ser utilizado. Para que o código funcione, os macros devem ser habilitados e para visualização do código

basta pressionar Alt+F11 e então é possível visualizar o código. Abaixo está calculado os potenciais dos nós, os campos elétricos em cada elemento, bem como o potencial solicitado no ponto da letra a).

Podemos observar que a discrepância entre os resultados do trabalho feito a mão e o programa confeccionado são mínimas (Ordem de  $10^{-4}$ ).

*Tabela 1 - Resultados do Programa*

<b>Nó</b>	<b>Potencial</b>	<b>Potencial Ponto</b>	<b>Elemento</b>	<b>Campo X</b>	<b>Campo Y</b>
1	0	1,618870089	1	1,082114521	-3,33333
2	0		2	0	-2,25122
3	0		3	0,421565816	-2,25122
4	3,23774		4	0	-1,61887
5	4,502438		5	0,421565816	-2,25409
6	10		6	1,221680528	-3,0542
7	10				

### 3. Comparação de Resultados

Abaixo está a comparação entre os resultados obtidos pelo programa feito pelo aluno e a tabela de resultados do artigo do Sadiku. Como é possível observar, não há discrepâncias entre os resultados obtidos e os do Sadiku.

*Tabela 2 - Programa Excel*

Nó	Potencial
1	0
2	0
3	0
4	0
5	0
6	50
7	0
8	18,18181818
9	36,36363636
10	59,09090909
11	100
12	0
13	36,36363636
14	68,18181818
15	100
16	0
17	59,09090909
18	100
19	0
20	100
21	50

*Tabela 3 - Resultados Sadiku*

TABLE IV OUTPUT DATA OF THE PROGRAM IN FIG. 7			
No. of Nodes = 21 No. of Elements = 25 No. of Fixed Nodes = 15			
Node	X	Y	Potential
1	0.00	0.00	0.000
2	0.20	0.00	0.000
3	0.40	0.00	0.000
4	0.60	0.00	0.000
5	0.80	0.00	0.000
6	1.00	0.00	50.000
7	0.00	0.20	0.000
8	0.20	0.20	18.182
9	0.40	0.20	36.364
10	0.60	0.20	59.091
11	0.80	0.20	100.000
12	0.00	0.40	0.000
13	0.20	0.40	36.364
14	0.40	0.40	68.182
15	0.60	0.40	100.000
16	0.00	0.60	0.000
17	0.20	0.60	59.091
18	0.40	0.60	100.000
19	0.00	0.80	0.000
20	0.20	0.80	100.000
21	0.00	1.00	50.000

## 4. Código do Programa em VBA

Para que o programa seja melhor visualizado, faz-se necessário lê-lo no próprio excel, onde as cores de funções e comentários auxiliam a compreender melhor os componentes do programa.

Programa:

```
Public inversa() As Double ' Variável global para calculo de inversa
```

```
Option Explicit
```

```
Sub Calculo()
```

```
ThisWorkbook.Worksheets("Resultados").Range("A1:Z10000").Delete 'deleta os dados existentes na planilha de resultados
```

```
Dim no1, no2, no3, per, c4, V(0 To 100) As String 'Declaração de variáveis a serem utilizadas
```

```
Dim c1, c2 As Object
```

```
Dim t As Object
```

```
Dim xx1, yy1 As Double
```

```
Dim x1(0 To 100), x2(0 To 100), x3(0 To 100) As Double
```

```
Dim y1(0 To 100), y2(0 To 100), y3(0 To 100) As Double
```

```
Dim q1(0 To 100, 0 To 100), q2(0 To 100, 0 To 100), q3(0 To 100, 0 To 100) As Double
```

```
Dim r1(0 To 100, 0 To 100), r2(0 To 100, 0 To 100), r3(0 To 100, 0 To 100) As Double
```

```
Dim p1(0 To 100, 0 To 100), p2(0 To 100, 0 To 100), p3(0 To 100, 0 To 100) As Double
```

```
Dim q(0 To 100, 0 To 100), r(0 To 100, 0 To 100), p(0 To 100, 0 To 100) As Double
```

```
Dim D(0 To 100), Cl(0 To 1000, 0 To 1000) As Integer
```

```
Dim fi(0 To 100, 0 To 100) As Double
```

```
Dim Ec(0 To 100, 0 To 100) As Double
```

```
Dim k, e, j, M, n, z, i, h, b, g, a As Integer
```

```
Dim elem(0 To 1000, 0 To 1000) As Double
```

```
Dim Vp As Double
```

```
Dim S(0 To 100, 0 To 100) As Double
```

```
Dim S1(0 To 100, 0 To 100) As Double
```

```
Dim Aplus(0 To 100, 0 To 100), Inv()
```

```
Dim Vtotal(0 To 100) As Double
```

```
no1 = 0 ' Alocação de valor inicial para algumas variáveis
```

```
no2 = 0
```

```
no3 = 0
```

```
k = 0
```

```
'Ativar a primeira planilha
```

```
ThisWorkbook.Worksheets("Dados").Activate
```

```
'Selecionar a célula com o primeiro elemento
```

```
Range("F2").Select
```

```
'Fazer a contagem de elementos existentes
```

```
Do While (IsEmpty(ActiveCell) = False)
```

```
    ActiveCell.Offset(1, 0).Select
```

```
    k = k + 1
```

```
Loop
```

```
Range("A2").Select 'Selecionar a célula com o primeiro nó
```

```
'Fazer a contagem de nós
```

```
Do While (IsEmpty(ActiveCell) = False)
```

```
    ActiveCell.Offset(1, 0).Select
```

```
    z = z + 1
```

```
Loop
```

```
For i = 0 To z ' Preenche a matriz global com zeros
```

```
    For j = 0 To z
```

```
        S(i, j) = 0
```

```
    Next j
```

```
Next i
```

```
ThisWorkbook.Worksheets("Dados").Activate
```

```
For e = 1 To k ' Gravando os nós para encontrá-los
```

```
    no1 = Range("F2").Offset(e - 1, 1).Value 'Leitura dos nós
```

```
    no2 = Range("F2").Offset(e - 1, 2).Value 'Leitura dos nós
```

```
    no3 = Range("F2").Offset(e - 1, 3).Value 'Leitura dos nós
```

```
    per = Range("F2").Offset(e - 1, 4).Value 'Leitura da permissividade relativa relativas aos  
elementos
```

```
    elem(e, 1) = no1
```

```
    elem(e, 2) = no2
```

```
    elem(e, 3) = no3
```

```
    elem(e, 4) = per
```

```
Next e
```

```
ThisWorkbook.Worksheets("Dados").Activate
```

```
For j = 1 To k 'Definindo x1,x2 e x3
```

```
    'O With juntamente om as variaveis c1, c2 e c3, procuram os elementos e associação
```

```

'com os respectivos pontos

With Worksheets("Dados").Range("A:A")

Set c1 = .Find(elem(j, 1), LookIn:=xlValues, LookAt:=xlPart, MatchCase:=True)

c1.Activate
x1(j) = c1.Offset(0, 1).Value
y1(j) = c1.Offset(0, 2).Value

Set c2 = .Find(elem(j, 2), LookIn:=xlValues, LookAt:=xlPart, MatchCase:=True)

c2.Activate
x2(j) = c2.Offset(0, 1).Value
y2(j) = c2.Offset(0, 2).Value

Set t = .Find(elem(j, 3), LookIn:=xlValues, LookAt:=xlPart, MatchCase:=True)

t.Activate
x3(j) = t.Offset(0, 1).Value
y3(j) = t.Offset(0, 2).Value

End With

' Calculo de q, r, p e D
q(j, 1) = y2(j) - y3(j)
q(j, 2) = y3(j) - y1(j)
q(j, 3) = y1(j) - y2(j)

r(j, 1) = x3(j) - x2(j)
r(j, 2) = x1(j) - x3(j)
r(j, 3) = x2(j) - x1(j)

p(j, 1) = x2(j) * y3(j) - x3(j) * y2(j)
p(j, 2) = x3(j) * y1(j) - x1(j) * y3(j)
p(j, 3) = x1(j) * y2(j) - x2(j) * y1(j)

D(j) = (x2(j) * y3(j) - x3(j) * y2(j)) + (x3(j) * y1(j) - x1(j) * y3(j)) + (x1(j) * y2(j) - x2(j) * y1(j))

c4 = elem(j, 4)

For b = 1 To 3 'Calculo da Matriz Global
    For g = 1 To 3

        S(elem(j, b), elem(j, g)) = (q(j, b) * q(j, g) + r(j, b) * r(j, g)) * c4 / (2 * D(j)) + S(elem(j, b),
elem(j, g))

```



```

Next g
Next b
Next j

```

```

ThisWorkbook.Worksheets("Dados").Activate
For e = 1 To z ' Procura das condições de contorno do problema

```

```

    V(e) = Range("D2").Offset(e - 1, 0).Value

```

```

    If V(e) <> "x" Then ' Caso exista X na célula, associasse 0 nesse ponto, pois é o potencial a ser descoberto.

```

```

        V(e) = V(e)
        For i = 1 To z ' Coloca 1 e 0 nas linhas de potenciais conhecidos
            S(e, i) = 0
            S(e, e) = 1
        Next i

```

```

    Else
        V(e) = 0
    End If
Next e

```

```

Aplus(z, z) = Inverter(S, z) ' Calculo da Inversa da Matriz Global
Vtotal(z) = 0

```

```

For i = 1 To z 'Calculo dos potenciais nos nós
    For j = 1 To z

```

```

        Vtotal(i) = inversa(i, j) * V(j) + Vtotal(i)

```

```

    Next j
Next i

```

```

Range("M2").Select

```

```

If (IsEmpty(ActiveCell) = False) Then 'Calculo de fi para ponto

```

```

    xx1 = Range("M2").Value 'Leitura do x do Ponto
    yy1 = Range("M3").Value 'Leitura do y do Ponto
    For j = 1 To k
        fi(j, 1) = (1 / D(j)) * ((p(j, 1)) + ((q(j, 1)) * xx1) + ((r(j, 1)) * yy1))
        fi(j, 2) = (1 / D(j)) * ((p(j, 2)) + ((q(j, 2)) * xx1) + ((r(j, 2)) * yy1))
        fi(j, 3) = (1 / D(j)) * ((p(j, 3)) + ((q(j, 3)) * xx1) + ((r(j, 3)) * yy1))
    Next j

```

```

    If fi(j, 1) >= 0 And fi(j, 1) <= 1 And fi(j, 2) >= 0 And fi(j, 2) <= 1 And fi(j, 3) >= 0 And fi(j, 3) <= 1
Then
    '      Usando fi e depois parametros para calculo de potencial no ponto
    Vp = ((fi(j, 1) * (Vtotal(elem(j, 1))) + (fi(j, 2) * (Vtotal(elem(j, 2)))) + (fi(j, 3) * (Vtotal(elem(j,
3))))))

    End If

    Next j
End If

ThisWorkbook.Worksheets("Resultados").Activate 'Selecione a planilha Resultados para escrita
de resultados

For j = 1 To k ' Calculo dos Campos elétricos
    Ec(j, 1) = (-1 / D(j)) * ((q(j, 1) * Vtotal(elem(j, 1)) + q(j, 2) * Vtotal(elem(j, 2)) + q(j, 3) *
Vtotal(elem(j, 3))))
    Ec(j, 2) = (-1 / D(j)) * ((r(j, 1) * Vtotal(elem(j, 1)) + r(j, 2) * Vtotal(elem(j, 2)) + r(j, 3) *
Vtotal(elem(j, 3))))
    Range("E1").Value = "Elemento"
    Range("E2").Offset(j - 1, 0).Value = j
    Range("F1").Value = "Campo X"
    Range("G1").Value = "Campo Y"
    Range("F2").Offset(j - 1, 0).Value = Ec(j, 1)
    Range("G2").Offset(j - 1, 0).Value = Ec(j, 2)
Next j

For i = 1 To z ' Grava na planilha os resultados encontrados, na Planilha Resultados

    Range("A1").Value = "Nó"
    Range("A2").Offset(i - 1, 0).Value = i
    Range("B1").Value = "Potencial"
    Range("B2").Offset(i - 1, 0).Value = Vtotal(i)
    Range("C1").Value = "Potencial Ponto"
    Range("C2").Value = Vp

Next i

End Sub

Public Function Inverter(ByVal S1 As Variant, ByVal z As Double) As Variant ' função para calcular
inversa

Dim antes As Variant

```

antes = Now

Dim i As Long

Dim j As Long

Dim k As Long

Dim a As Double

Dim celulas As Variant

Dim ordem As Integer

Dim matriz(0 To 1000, 0 To 1000) As Double

ordem = z

For i = 1 To ordem ' Escreve matriz para calculo de inversa

For j = 1 To ordem

matriz(i, j) = S1(i, j)

Next j

Next i

ReDim inversa(ordem, ordem)

Application.ScreenUpdating = False 'Este comando desativa a atualização da tela

Application.Calculation = xlManual 'Este comando desativa o cálculo automático das células

Application.EnableEvents = False 'Este comando desativa os eventos do Excel

'laço de repetição que cria uma matriz identidade e armazena na variável inversa

For i = 1 To ordem

For j = 1 To ordem

If i = j Then

inversa(i, j) = 1

Else

inversa(i, j) = 0

End If

Next j

Next i

'laço de repetição para fazer a triangulação inferior da matriz

For k = 1 To ordem

If matriz(k, k) <> 0 Then

For i = k To ordem

If matriz(i, k) <> 0 And matriz(i, k) <> 1 Then

a = matriz(i, k)

For j = 1 To ordem

matriz(i, j) = matriz(i, j) / a

```

        inversa(i, j) = inversa(i, j) / a

    Next j
End If
Next i

For i = k + 1 To ordem
    If matriz(i, k) <> 0 Then
        For j = 1 To ordem
            matriz(i, j) = matriz(i, j) - matriz(k, j)
            inversa(i, j) = inversa(i, j) - inversa(k, j)

        Next j
    End If
Next i
Else
    MsgBox "Não existe Matriz Inversa."

End If
Next k
'laço de repetição para fazer a triangulação superior da matriz

```

```

For k = 0 To ordem - 1
    For i = 1 To ordem - 1 - k
        If matriz(i, ordem - k) <> 0 Then
            a = matriz(i, ordem - k)
            For j = 1 To ordem
                matriz(i, j) = matriz(i, j) - a * matriz(ordem - k, j)
                inversa(i, j) = inversa(i, j) - a * inversa(ordem - k, j)
            Next j
        End If
    Next i
Next k

```

Inverter = inversa() ' Envia resultado para a função principal

```

Application.EnableEvents = True
Application.Calculation = xlAutomatic
Application.ScreenUpdating = True

```

End Function

