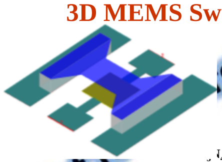
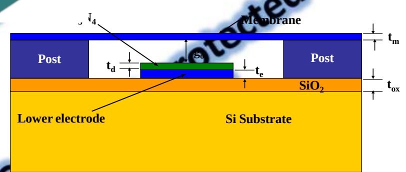
Gauss’s Low with 2D Non-Uniform Grid

3D MEMS Switch and 2D Simulation





Ehab K. I. Hamad, Atef Z. Elsherbeni, Amr M. E. Safwat, and Abbas S. Omar, “Two-dimensional Coupled Electrostatic-mechanical Model For RF MEMS Switches,” ACES Journal, March 2006.

2D Domain with Non-Uniform Grid

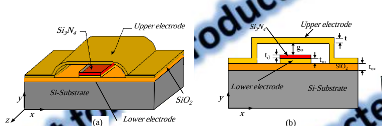
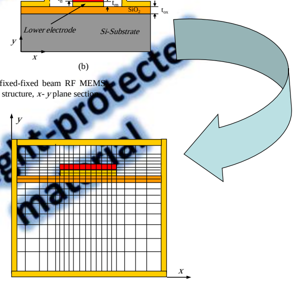
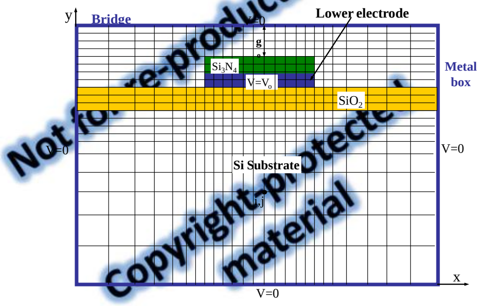


Fig. 1. Schematic diagram of fixed-fixed beam RF MEMS switch, (a) 3D structure, (b) 2D structure, x- y plane section.



MEMS Switch with Non-Uniform Grid for FD Simulation



Ehab K. I. Hamad, Atef Z. Elsherbeni, Amr M. E. Safwat, and Abbas S. Omar, "Two-dimensional Coupled Electrostatic-mechanical Model For RF MEMS Switches," ACES Journal, March 2006.

Potential at Nodes at The Intersection Between Four Different Media

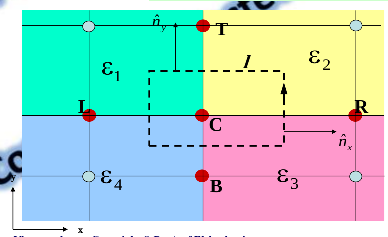
$$\rho^{enc} = -\oint_A \varepsilon\left(\frac{\partial V}{\partial x}\hat{x} + \frac{\partial V}{\partial y}\hat{y}\right) \bullet \hat{n}\, dl$$

where $dl = dA/$(unit length in z),
$\hat{n}$ is unit vctor normal to dA and dl

With no enclosed charge

$$-\oint_l \varepsilon\left(\frac{\partial V}{\partial x}\hat{x} + \frac{\partial V}{\partial y}\hat{y}\right) \bullet \hat{n}\, dl = 0$$

Non-uniform grid in both x and y directions.

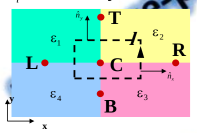


Potential at Nodes at The Interfaces Between Four Different Media for a 2D Problem

$$-\oint_l \varepsilon\left(\frac{\partial V}{\partial x}\hat{x} + \frac{\partial V}{\partial y}\hat{y}\right) \bullet \hat{n}\, dl = 0$$

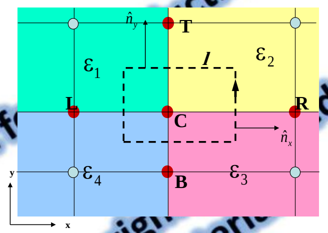
$$-\left[\begin{array}{l} \int_{right} \varepsilon(y)\frac{\partial V}{\partial x}dy + \int_{top} \varepsilon(x)\frac{\partial V}{\partial y}dx \\ -\int_{left} \varepsilon(y)\frac{\partial V}{\partial x}dy - \int_{bottom} \varepsilon(x)\frac{\partial V}{\partial y}dx \end{array}\right] = 0$$

Ignore the negative sign on the left side.



$$\frac{V_R - V_C}{x_R - x_C}\left(\frac{\varepsilon_3(y_C - y_B)/2 + \varepsilon_2(y_T - y_C)/2}{(y_T - y_B)/2}\right)\frac{(y_T - y_B)}{2} + \frac{V_T - V_C}{y_T - y_C}\left(\frac{\varepsilon_1(x_C - x_L)/2 + \varepsilon_2(x_R - x_C)/2}{(x_R - x_L)/2}\right)\frac{(x_R - x_L)}{2}$$
$$-\frac{V_C - V_L}{x_C - x_L}\left(\frac{\varepsilon_1(y_T - y_C)/2 + \varepsilon_4(y_C - y_B)/2}{(y_T - y_B)/2}\right)\frac{(y_T - y_B)}{2} - \frac{V_C - V_B}{y_C - y_B}\left(\frac{\varepsilon_4(x_C - x_L)/2 + \varepsilon_3(x_R - x_C)/2}{(x_R - x_L)/2}\right)\frac{(x_R - x_L)}{2} = 0$$

Re-naming of the Potentials and the Media Properties in Terms of the Node Coordinates for a 2D Problem



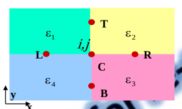
$$V_C = V_{(i,j)}\,,\; V_T = V_{(i,j+1)}\,,\; V_B = V_{(i,j-1)}\,,\; V_R = V_{(i+1,j)}\,,\; V_L = V_{(i-1,j)}$$

$$x_C = x(i,j)\,,\; x_T = x(i,j+1)\,,\; x_B = x(i,j-1)\,,\; x_R = x(i+1,j)\,,\; x_L = x(i-1,j)$$

$$y_C = y(i,j)\,,\; y_T = y(i,j+1)\,,\; y_B = y(i,j-1)\,,\; y_R = y(i+1,j)\,,\; y_L = y(i-1,j)$$

$$\varepsilon_1 = \varepsilon_{(i-1,j)}\,,\; \varepsilon_2 = \varepsilon_{(i,j)}\,,\; \varepsilon_3 = \varepsilon_{(i,j-1)}\,,\; \varepsilon_4 = \varepsilon_{(i-1,j-1)}$$

Potential Equation for a Node at the Corner of Four Different Media for a 2D Problem



$$\frac{V_R - V_C}{x_R - x_C}\left(\frac{\varepsilon_3(y_C - y_B) + \varepsilon_2(y_T - y_C)}{2}\right) + \frac{V_T - V_C}{y_T - y_C}\left(\frac{\varepsilon_1(x_C - x_L) + \varepsilon_2(x_R - x_C)}{2}\right)$$
$$-\frac{V_C - V_L}{x_C - x_L}\left(\frac{\varepsilon_1(y_T - y_C) + \varepsilon_4(y_C - y_B)}{2}\right) - \frac{V_C - V_B}{y_C - y_B}\left(\frac{\varepsilon_4(x_C - x_L) + \varepsilon_3(x_R - x_C)}{2}\right) = 0$$

$$\frac{V_{(i+1,j)} - V_{(i,j)}}{x_{(i+1,j)} - x_{(i,j)}}\left(\varepsilon_{(i,j-1)}\frac{(y_{(i,j)} - y_{(i,j-1)})}{2} + \varepsilon_{(i,j)}\frac{(y_{(i,j+1)} - y_{(i,j)})}{2}\right)$$
$$+\frac{V_{(i,j+1)} - V_{(i,j)}}{y_{(i,j+1)} - y_{(i,j)}}\left(\varepsilon_{(i-1,j)}\frac{(x_{(i,j)} - x_{(i-1,j)})}{2} + \varepsilon_{(i,j)}\frac{(x_{(i+1,j)} - x_{(i,j)})}{2}\right)$$
$$-\frac{V_{(i,j)} - V_{(i-1,j)}}{x_{(i,j)} - x_{(i-1,j)}}\left(\varepsilon_{(i-1,j)}\frac{(y_{(i,j+1)} - y_{(i,j)})}{2} + \varepsilon_{(i-1,j-1)}\frac{(y_{(i,j)} - y_{(i,j-1)})}{2}\right)$$
$$-\frac{V_{(i,j)} - V_{(i,j-1)}}{y_{(i,j)} - y_{(i,j-1)}}\left(\varepsilon_{(i-1,j-1)}\frac{(x_{(i,j)} - x_{(i-1,j)})}{2} + \varepsilon_{(i,j-1)}\frac{(x_{(i+1,j)} - x_{(i,j)})}{2}\right) = 0$$

Potential of Nodes at The Corner of Four Different Media

$$\left(V_{(i+1,j)} - V_{(i,j)}\right)\left[\varepsilon_{(i,j-1)}\frac{(y_{(i,j)} - y_{(i,j-1)})}{(x_{(i+1,j)} - x_{(i,j)})} + \varepsilon_{(i,j)}\frac{(y_{(i,j+1)} - y_{(i,j)})}{(x_{(i+1,j)} - x_{(i,j)})}\right] \Rightarrow A(i,j)$$

$$+\left(V_{(i,j+1)} - V_{(i,j)}\right)\left[\varepsilon_{(i-1,j)}\frac{(x_{(i,j)} - x_{(i-1,j)})}{(y_{(i,j+1)} - y_{(i,j)})} + \varepsilon_{(i,j)}\frac{(x_{(i+1,j)} - x_{(i,j)})}{(y_{(i,j+1)} - y_{(i,j)})}\right] \Rightarrow B(i,j)$$

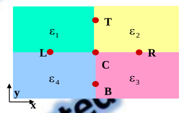
$$-\left(V_{(i-1,j)} - V_{(i,j)}\right)\left[\varepsilon_{(i-1,j)}\frac{(y_{(i,j+1)} - y_{(i,j)})}{(x_{(i-1,j)} - x_{(i,j)})} + \varepsilon_{(i-1,j-1)}\frac{(y_{(i,j)} - y_{(i,j-1)})}{(x_{(i-1,j)} - x_{(i,j)})}\right] \Rightarrow -D(i,j)$$

$$-\left(V_{(i,j-1)} - V_{(i,j)}\right)\left[\varepsilon_{(i-1,j-1)}\frac{(x_{(i,j)} - x_{(i-1,j)})}{(y_{(i,j-1)} - y_{(i,j)})} + \varepsilon_{(i,j-1)}\frac{(x_{(i+1,j)} - x_{(i,j)})}{(y_{(i,j-1)} - y_{(i,j)})}\right] = 0 \Rightarrow -E(i,j)$$

$$(V_{(i+1,j)} - V_{(i,j)})A(i,j) + (V_{(i,j+1)} - V_{(i,j)})B(i,j) + (V_{(i-1,j)} - V_{(i,j)})D(i,j) + (V_{(i,j-1)} - V_{(i,j)})E(i,j) = 0$$

Final Expression of the Potential at Node C

$$-\oint_l \varepsilon\left(\frac{\partial V}{\partial x}\hat{x}+\frac{\partial V}{\partial y}\hat{y}+\frac{\partial V}{\partial z}\hat{z}\right)\bullet\hat{n}\,dl=0$$



$$(V_{(i+1,j)}-V_{(i,j)})A(i,j)+(V_{(i,j+1)}-V_{(i,j)})B(i,j)+(V_{(i-1,j)}-V_{(i,j)})D(i,j)+(V_{(i,j-1)}-V_{(i,j)})E(i,j)=0$$

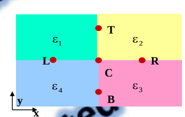
$$A(i,j)V_{(i+1,j)}+B(i,j)V_{(i,j+1)}+D(i,j)V_{(i-1,j)}+E(i,j)V_{(i,j-1)}=V_{(i,j)}(A(i,j)+B(i,j)+D(i,j)+E(i,j))$$

with $(A(i,j)+B(i,j)+D(i,j)+E(i,j))=G(i,j)$

$$V_{(i,j)}=\frac{A}{G}V_{(i+1,j)}+\frac{B}{G}V_{(i,j+1)}+\frac{D}{G}V_{(i-1,j)}+\frac{E}{G}V_{(i,j-1)}$$
$$=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}$$

Final Expression of the Potential at Node C

$$-\oint_l \varepsilon\left(\frac{\partial V}{\partial x}\hat{x}+\frac{\partial V}{\partial y}\hat{y}+\frac{\partial V}{\partial z}\hat{z}\right)\bullet\hat{n}\,dl=0$$

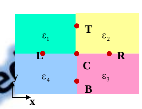


$$V_{(i,j)}=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}$$
$$\Downarrow$$
$$V_{(i,j)}=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}+V^0_{(i,j)}$$

where $V^0_{(i,j)}$ is the fixed potential at node (i,j) if it is held fixed.

Summary - Coefficients of the Potential at Node C

$$V_{(i,j)}=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}+V^0_{(i,j)}$$



$$C_R=\frac{A}{G},\quad C_T=\frac{B}{G},\quad C_L=\frac{D}{G},\quad C_B=\frac{E}{G}$$

$$G(i,j)=\left[A(i,j)+B(i,j)+D(i,j)+E(i,j)\right]$$

$$A(i,j)=\left[\varepsilon_{(i,j-1)}\frac{(y_{(i,j)}-y_{(i,j-1)})}{(x_{(i+1,j)}-x_{(i,j)})}+\varepsilon_{(i,j)}\frac{(y_{(i,j+1)}-y_{(i,j)})}{(x_{(i+1,j)}-x_{(i,j)})}\right]$$

$$B(i,j)=\left[\varepsilon_{(i-1,j)}\frac{(x_{(i,j)}-x_{(i-1,j)})}{(y_{(i,j+1)}-y_{(i,j)})}+\varepsilon_{(i,j)}\frac{(x_{(i+1,j)}-x_{(i,j)})}{(y_{(i,j+1)}-y_{(i,j)})}\right]$$

$$D(i,j)=-\left[\varepsilon_{(i-1,j)}\frac{(y_{(i,j+1)}-y_{(i,j)})}{(x_{(i-1,j)}-x_{(i,j)})}+\varepsilon_{(i-1,j-1)}\frac{(y_{(i,j)}-y_{(i,j-1)})}{(x_{(i-1,j)}-x_{(i,j)})}\right]$$

$$E(i,j)=-\left[\varepsilon_{(i-1,j-1)}\frac{(x_{(i,j)}-x_{(i-1,j)})}{(y_{(i,j-1)}-y_{(i,j)})}+\varepsilon_{(i,j-1)}\frac{(x_{(i+1,j)}-x_{(i,j)})}{(y_{(i,j-1)}-y_{(i,j)})}\right]$$

Special Cases for the Coefficients of the Potential

$$A(i,j)=\left[\varepsilon_{(i,j-1)}\frac{(y_{(i,j,k)}-y_{(i,j-1)})}{(x_{(i+1,j)}-x_{(i,j)})}+\varepsilon_{(i,j)}\frac{(y_{(i,j+1)}-y_{(i,j)})}{(x_{(i+1,j)}-x_{(i,j)})}\right]$$

With uniform grid in x and y

$$A(i,j)=\left(\varepsilon_{(i,j-1)}\frac{\Delta y}{\Delta x}+\varepsilon_{(i,j)}\frac{\Delta y}{\Delta x}\right)$$

With identical uniform grid in x and y

$$A(i,j)=\left(\varepsilon_{(i,j-1,k)}+\varepsilon_{(i,j,k)}\right)$$

With identical uniform grid in x and y and uniform media

$$A=2\varepsilon \quad , \quad G=8\varepsilon \quad , \quad C_R=C_T=C_L=C_B=\frac{1}{4}$$

$$V_{(i,j)}=\frac{1}{4}\left[V_{(i+1,j)}+V_{(i,j+1)}+V_{(i-1,j)}+V_{(i,j-1)}\right]+V^0_{(i,j)}$$

Gauss-Seidel Iterative Scheme

Gauss-Seidel method characterizes the ordering of the iterations. There are two ways to do this:

1) *Lexicographical ordering*
2) *Chequer-board ordering (red-black ordering)* → **Applied in this work**

$$V_{(i,j)}=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}+V^0_{(i,j)}$$

The following is the updating procedure based on the chequer-board ordering scheme applied at each iteration

$V(i,j)$ for *even* (i,j)
$V(i,j)$ for *odd* (i,j)
$V(i,j)$ for *even* (i), and *odd* (j)
$V(i,j)$ for *odd* (i), and *even* (j)

Gauss-Seidel Scheme

Sample updating procedure within each iteration

$$V_{(i,j)}=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}+V^0_{(i,j)} \quad (i,j)\ even$$

```
V(2:2:nx,2:2:ny) = +Cr(2:2:nx,2:2:ny).*V(3:2:nx+1,2:2:ny) …
                  +Ct(2:2:nx,2:2:ny).*V(2:2:nx,3:2:ny+1)…
                  + Cl(2:2:nx,2:2:ny).*V(1:2:nx-1,2:2:ny)…
                  +Cb(2:2:nx,2:2:ny).*V(2:2:nx,1:2:ny-1)…
                  +Vo(2:2:nx,2:2:ny);
```

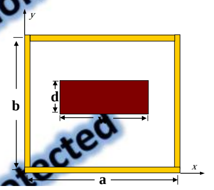
$$V_{(i,j)}=C_R(i,j)V_{(i+1,j)}+C_T(i,j)V_{(i,j+1)}+C_L(i,j)V_{(i-1,j)}+C_B(i,j)V_{(i,j-1)}+V^0_{(i,j)} \quad (i)\ odd,\ (j)\ even$$

```
Voltage(3:2:nx-1,2:2:ny) =     +Cr(3:2:nx-1,2:2:ny).*V(4:2:nx,2:2:ny) …
                               +Ct(3:2:nx-1,2:2:ny).*V(3:2:nx-1,3:2:ny+1)…
                               + Cl(3:2:nx-1,2:2:ny).*V(2:2:nx-2,2:2:ny)…
                               +Cb(3:2:nx-1,2:2:ny).*V(3:2:nx-1,1:2:ny-1)…
                               +Vo(3:2:nx-1,2:2:ny);
```

Example of a 2D Domain with Non-Uniform Grid

a = 6.4 b = 6.4 w = 1 d = 0.5

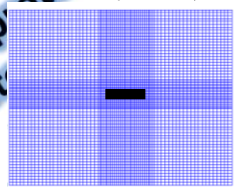
All dimensions are in mm



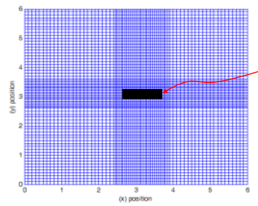
Uniform descritization: dx_course = 0.1, dy_course = 0.1

Non-uniform descritization: dx_fine = 0.05, dy_fine = 0.05

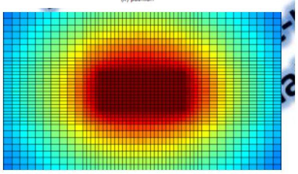
Total number of cells is ***5396*** with nx = **76** and ny = **71**

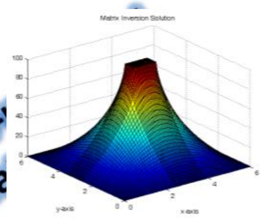


2D Domain with Non-Uniform Grid

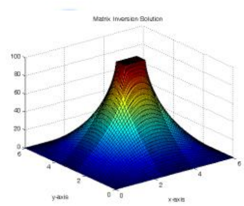


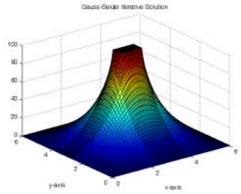
Strip position in the grid

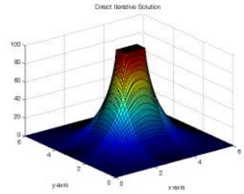




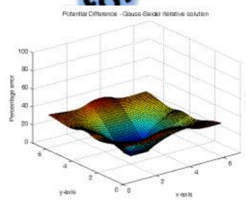
Potential Distribution 500 Iterations

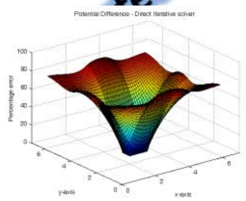






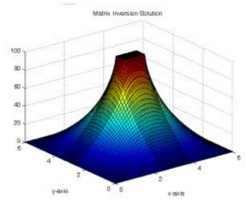
Relative Error in the Potential with respect to the Matrix Solution - 500 Iterations

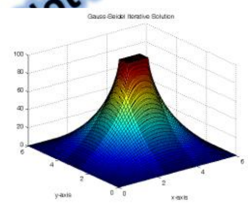


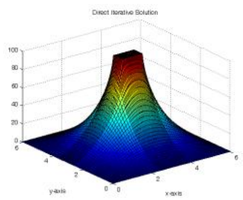


11/11/2012 Viewgraphs are Copyright © Dr. Atef Elsherbeni 19

Potential Distribution 1000 Iterations

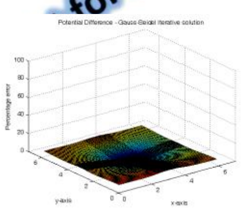


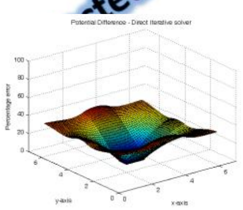




11/11/2012 Viewgraphs are Copyright © Dr. Atef Elsherbeni 20

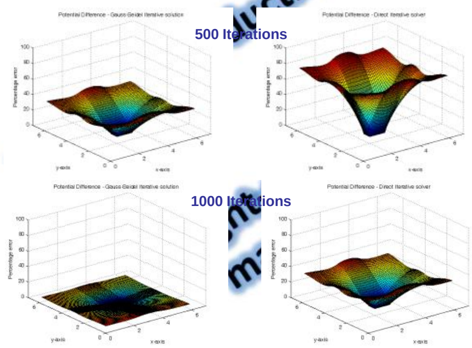
Relative Error in the Potential with respect to the Matrix Solution - 1000 Iterations





11/11/2012 Viewgraphs are Copyright © Dr. Atef Elsherbeni 21

Relative Error in the Potential with respect to the Matrix Solution – 500 and 1000 Iterations



2D Domain with Non-Uniform Grid

Three different Solution procedures

1) Matrix inversion solution
2) Direct iterative solution
3) Gauss-Seidel iterative solution

PC Specifications:
2 GHz CPU
2 GB RAM

Matrix inversion solution Time in Sec	*Direct iterative solution* Time in Sec./No. of Iterations	*Gauss-Seidel iterative solution* Time in Sec./No. of Iterations
0.391	0.531/500	0.406/500
0.475	0.781/500	0.547/500
0.375	0.828/1000	0.484/1000
0.375	0.672/1000	0.578/1000
0.391	1.953/3000	1.719/3000
0.375	5.859/10000	4.344/10000

With Pre-Specified Error

Matrix Solution
CPU time =0.375 Sec.

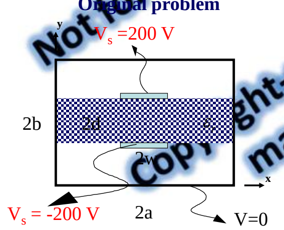
Direct Iterative Solution
Number of steps used =2389
Percentage error = 4.9935 %
CPU time =2.312 Sec.

Gauss Seidel Iterative Solution
Number of steps used =1195
Percentage error = 4.9912 %
CPU time =1.203 Sec.

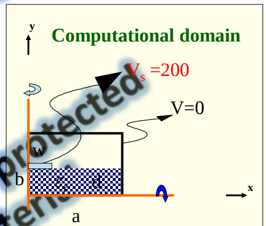
Home Work - Shielded Double Strip Line

Write a MatLab program based on the Gauss Siedel iterative solution to compute and display the potential distribution for a shielded double strip line as shown in the original figure below. Your computational domain should be the top right quarter of the geometry as shown in the right figure. Use non-uniform grid and assume that the percentage error does not exceed 0.05% of $|V_s|$. Report the number of steps and the CPU time used for the solution.

Original problem



Computational domain

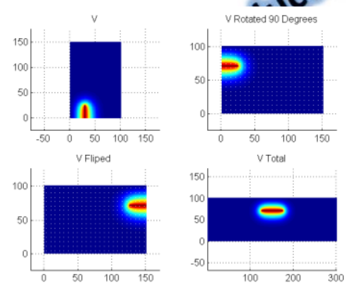


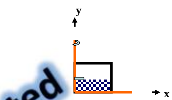
$a = 3.5, b = 3, w = 0.5, d = 1.5, \varepsilon_r = 10$
Thickness of the strip =.05
All dimensions are in cm.

Properties of Flipping Commands

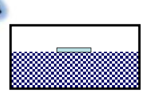
```
x =
     1     4
     2     5
     3     6

» flipud(x)

ans =
     3     6
     2     5
     1     4

» fliplr(x)

ans =
     4     1
     5     2
     6     3

» rot90(x)

ans =
     4     5     6
     1     2     3

» flipdim(x,1)

ans =
     3     6
     2     5
     1     4

» flipdim(x,2)

ans =
     4     1
     5     2
     6     3
```


Example on How to Use Matlab Commands to Construct the Full Domain Solution From One Half Domain Solution

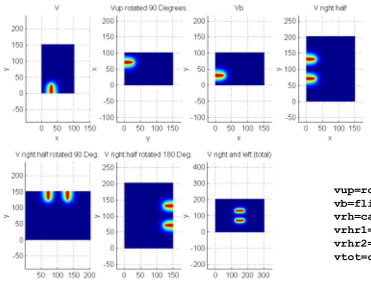


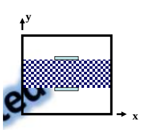



```
vrot90 = rot90(v);
vflip=fliplr(vrot90,2);
vrh = cat(2,vflip,vrot90);
```



Example on How to Use Matlab Commands to Construct the Full Domain Solution From One Quarter Solution

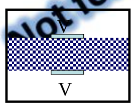


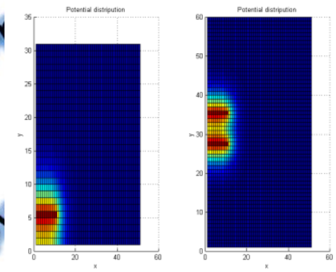



```
vup=rot90(v);
vb=flipud(vup);
vrh=cat(1,vup,vb);
vrhr1=rot90(vrh);
vrhr2=rot90(vrhr1);
vtot=cat(2,vrhr2,vrh);
```

"flipud" and "cat" Commands for Symmetric Lines


```
vt = v';
vup = flipud(v');
vtt = vt(2:ny,:);
vft = cat(1,vup,vtt);
```





"flipud" and "cat" Commands for Anti-Symmetric Lines


```
vt = v';
vup = flipud(v');
vtt = -vt(2:ny,:);
vf = cat(1,vup,vtt);
vft = vf';
```