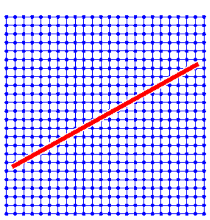
EMPossible

Advanced Electromagnetics:
21st Century Electromagnetics

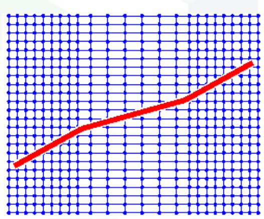
Analytical Transformation Optics

Concept of Transformation Optics

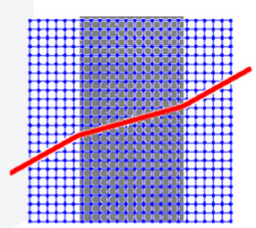
Transformation optics is an analytical technique to calculate the permittivity and permeability functions that will bend fields in a prescribed manner.



Define a uniform grid with uniform rays.



Perform a coordinate transformation such that the rays follow some desired path.



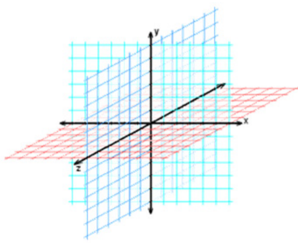
Move the coordinate transformation into the material tensors.

EMPossible

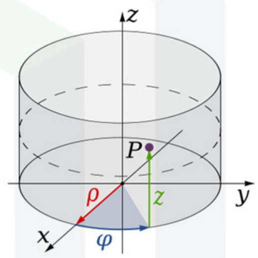
Step 1: Pick a Coordinate System

Pick a coordinate system that is most convenient for the device geometry.

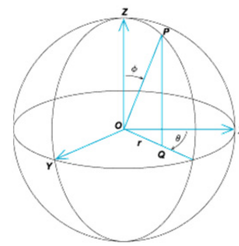
Cartesian



Cylindrical



Spherical



Other?

EMPossible

Step 2: Draw Straight Rays Through the Grid

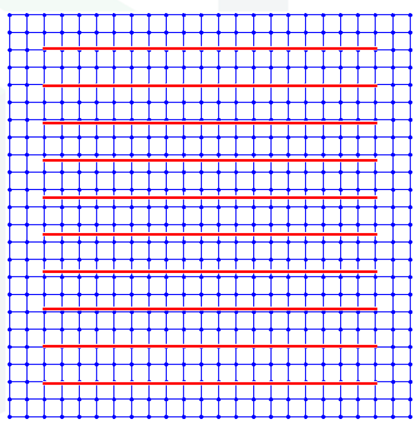
Start with a plane wave in free space, draw the rays passing straight through the coordinate system.

x

y

z

$$[\mu]=[\varepsilon]=\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$$



EMPossible

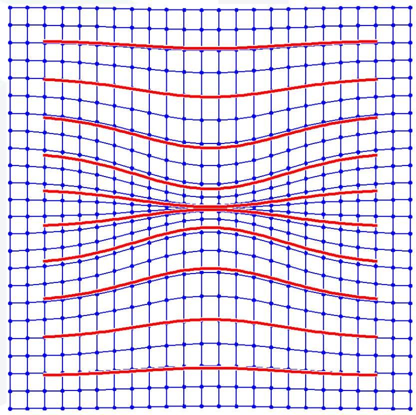
Step 3: Define a Coordinate Transform

Define a coordinate transform so that the rays will follow the desired path. Here, the wave is being "squeezed" at the center of the grid.

$$x' = x$$

$$y' = y\left[1 - \exp\left(-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}\right)\right]$$

$$z' = z$$



EMPossible

5

Step 4: Calculate the Jacobian Matrix

Given the coordinate transformation, the Jacobian matrix $[J]$ is calculated.

$$x' = x$$

$$y' = y\left[1 - \exp\left(-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}\right)\right]$$

$$z' = z$$

$$[J] = \begin{bmatrix} 1 & 0 & 0 \\ \frac{2xy}{\sigma_x^2} e^{-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}} & 1 + \left(\frac{2y^2}{\sigma_y^2} - 1\right) e^{-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\frac{\partial x'}{\partial x} = 1$$

$$\frac{\partial x'}{\partial y} = 0$$

$$\frac{\partial x'}{\partial z} = 0$$

$$\frac{\partial y'}{\partial x} = \frac{2xy}{\sigma_x^2} \exp\left(-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}\right)$$

$$\frac{\partial y'}{\partial y} = 1 + \left(\frac{2y^2}{\sigma_y^2} - 1\right) \exp\left(-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}\right)$$

$$\frac{\partial y'}{\partial z} = 0$$

$$\frac{\partial z'}{\partial x} = 0$$

$$\frac{\partial z'}{\partial y} = 0$$

$$\frac{\partial z'}{\partial z} = 1$$

EMPossible

6

Step 5: Calculate the Material Tensors

The material tensors are calculated according to.

$$[\mu'] = \frac{[J][\mu][J]^T}{\det[J]} \qquad [\varepsilon'] = \frac{[J][\varepsilon][J]^T}{\det[J]}$$

The elements of these tensors are

$$\varepsilon_{xx} = \varepsilon_{zz} = \mu_{xx} = \mu_{zz} = \frac{\sigma_y^2}{\sigma_y^2 + \left(2y^2 - \sigma_y^2\right) e^{-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}}}$$

$$\varepsilon_{xy} = \varepsilon_{yx} = \mu_{xy} = \mu_{yx} = \frac{2xy\sigma_y^2}{\sigma_x^2 \left[2y^2 - \sigma_y^2 \left(1 - e^{\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}}\right)\right]}$$

$$\varepsilon_{yy} = \mu_{yy} = \frac{\sigma_y^2 e^{\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}}}{2y^2 - \sigma_y^2 \left(1 - e^{\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}}\right)} \left[\left(1 + \left(\frac{2y^2}{\sigma_y^2} - 1\right) e^{-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}}\right)^2 + \frac{x^2 y^2}{\sigma_x^4} e^{-\frac{2x^2}{\sigma_x^2} - \frac{2y^2}{\sigma_y^2}}\right]$$

$$\varepsilon_{xz} = \varepsilon_{yz} = \varepsilon_{zx} = \varepsilon_{zy} = \mu_{xz} = \mu_{yz} = \mu_{zx} = \mu_{zy} = 0$$

EMPossible

7

MATLAB Code to Generate the Equations

```
% DEFINE VARIABLES
syms x y z;
syms sx sy;

% INITIALIZE MATERIALS
UR = eye(3,3);
ER = eye(3,3);

% DEFINE COORDIANTE TRANSFORMATION
xp = x;
yp = y*(1 - exp(-x^2/sx^2)*exp(-y^2/sy^2));
zp = z;

% COMPUTE ELEMENTS OF JACOBIAN
J = [ diff(xp,x) diff(xp,y) diff(xp,z) ; ...
      diff(yp,x) diff(yp,y) diff(yp,z) ; ...
      diff(zp,x) diff(zp,y) diff(zp,z) ];

% ABSORB TRANSFORM INTO MATERIALS
UR = J*UR*J.'/det(J);
ER = J*ER*J.'/det(J);

% SHOW TENSOR
pretty(ER);
```

$x,\ y,\ z,\ \sigma_x,\ \sigma_y$

$$[\mu] = [\varepsilon] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$x' = x$$

$$y' = y\left[1 - \exp\left(-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2}\right)\right]$$

$$z' = z$$

$$[J] = \begin{bmatrix} \partial x'/\partial x & \partial x'/\partial y & \partial x'/\partial z \\ \partial y'/\partial x & \partial y'/\partial y & \partial y'/\partial z \\ \partial z'/\partial x & \partial z'/\partial y & \partial z'/\partial z \end{bmatrix}$$

$$[\mu'] = \frac{[J][\mu][J]^T}{\det[J]} \qquad [\varepsilon'] = \frac{[J][\varepsilon][J]^T}{\det[J]}$$

EMPossible

8

MATLAB Code to Fill Grid

```
% PARAMETERS
ER = subs(ER,sx,0.25);
ER = subs(ER,sy,0.25);

% FILL GRID
for ny = 1 : Ny
    for nx = 1 : Nx
        er = subs(ER,x,X(nx,ny));
        er = subs(er,y,Y(nx,ny));

        ERxx(nx,ny) = er(1,1);
        ERxy(nx,ny) = er(1,2);
        ERxz(nx,ny) = er(1,3);

        ERyx(nx,ny) = er(2,1);
        ERyy(nx,ny) = er(2,2);
        ERyz(nx,ny) = er(2,3);

        ERzx(nx,ny) = er(3,1);
        ERzy(nx,ny) = er(3,2);
        ERzz(nx,ny) = er(3,3);
    end
end
```

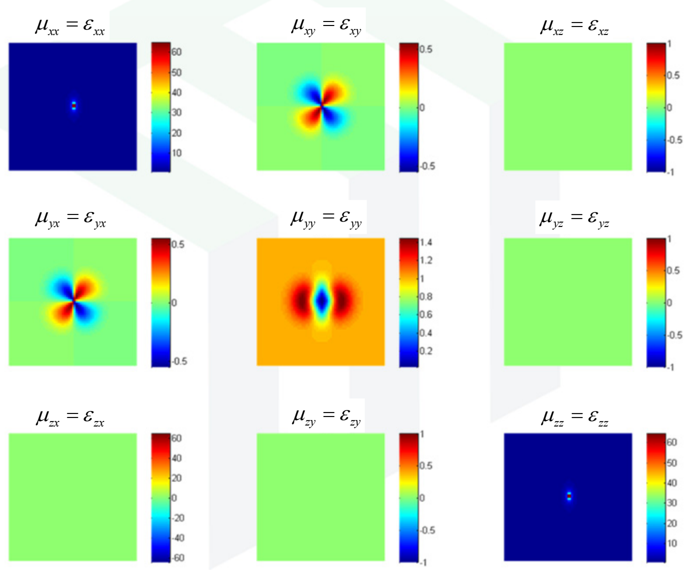
X and Y generated by MATLAB's meshgrid() command.

This is VERY slow!!!

Recommend calculating analytical equations and then manually typing these into MATLAB to construct the device.

9

Plot of the Final Tensor Over Entire Device



10