Phy482Hw2Prob6Wood

January 26, 2018

1 Computing Electric Field Along the Axis of a Charged Ring With a Computer

This program computes the net electric field due to a uniformly charged ring of radius R and charge Q at a given point in space.

In [1]:
```
from __future__ import division, print_function
from vpython import *
from math import *
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.Javascript object>


<IPython.core.display.Javascript object>


<IPython.core.display.Javascript object>


<IPython.core.display.Javascript object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>
```

In [2]:
```
scene=canvas(title="Electric field due to uniformly charged ring at various points")
scene.background=color.white

R=0.02 #radius of ring in m
```

```
Q=1e-9 #charge of ring in C
N=5 #number of unique pieces
npoints = 3

#draw the objects
myring=ring(pos=vector(0,0,0), radius=R, axis=vector(0,0,1), color=color.blue, thicknes
zaxis=cylinder(pos=-2*R*myring.axis, radius=0.015*R, axis=4*R*myring.axis, color=color
points = []
for i in range(npoints):

    xr = 0.01*sin(i*2*pi/npoints)
    yr = 0.01*cos(i*2*pi/npoints)

    points.append(sphere(pos=vector(xr,yr,0), color=color.red, radius=5*zaxis.radius))

for j in range(npoints,2*npoints):

    xr2 = 0.005*sin(j*2*pi/npoints)
    yr2 = 0.005*cos(j*2*pi/npoints)

    points.append(sphere(pos=vector(xr2,yr2,0), color=color.red, radius=5*zaxis.radius)

# Find net field
for a in points:

    oofpez=9e9 #1/(4pi epsilon_0) in N m^2/C^2
    dq=Q/N #charge of a piece
    dtheta=2*pi/N #theta increment for our loop
    theta=dtheta/2 #initial theta for first piece of loop
    Enet=vector(0,0,0) #net electric field of all pieces

    rpoint = a.pos

    scale=1.2*mag(rpoint)/8000 #used to scale the arrows representing E-field

    while theta<2*pi:
        rpiece = R*vector(cos(theta),sin(theta),0) #location of piece
        r = rpoint-rpiece #vector from piece to point in space
        rmag = mag(r) #magnitude of r
        rhat = norm(r) #unit vector for r
        dE = oofpez * dq / rmag / rmag * rhat #Electric field due to piece at rpoint
        Enet = Enet + dE
        particle=sphere(pos=rpiece, radius=a.radius, color=color.yellow) #draw a parti
        dEvector=arrow(pos=rpoint, axis=scale*dE, color=color.magenta, shaftwidth=a.rad
        theta=theta+dtheta

    print("The net electric field at ", rpoint, "= ",Enet, "N/C")
    Evector=arrow(pos=rpoint, axis=scale*Enet, color=color.orange, shaftwidth=a.radius/
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.Javascript object>
```

```
The net electric field at  <0, 0.01, 0> =  <3936.48, -7576.91, 0> N/C
The net electric field at  <0.00866025, -0.005, 0> =  <-8867.63, 7582.05, 0> N/C
The net electric field at  <-0.00866025, -0.005, 0> =  <4577.03, 535.732, 0> N/C
The net electric field at  <-1.22465e-18, 0.005, 0> =  <222.772, -3023.69, 0> N/C
The net electric field at  <0.00433013, -0.0025, 0> =  <-2732.29, 1706.42, 0> N/C
The net electric field at  <-0.00433013, -0.0025, 0> =  <2505.45, 1318.21, 0> N/C
```

In [4]:
```
scene=canvas(title="Electric field inside wire")
scene.background=color.white

R=0.02 #radius of ring in m
Q=1e-9 #charge of ring in C
N=4 #number of unique pieces
npoints = 5
nrings = 11
spacing = 0.005
c = 1e-11 #change in charge from center ring

#draw the objects
zaxis=cylinder(pos=-2*R*myring.axis, radius=0.015*R, axis=4*R*myring.axis, color=color
rings = []
charge = [Q-5*c,Q-4*c,Q-3*c,Q-2*c,Q-c,Q,Q+c,Q+2*c,Q+3*c,Q+4*c,Q+5*c]

for i in range(nrings):

    loc = i - (nrings)//2
    rings.append(ring(pos=vector(0,0,spacing*loc), radius=R, axis=vector(0,0,1), color=

points = []
for i in range(npoints):

    xr = 0.01*sin(i*2*pi/npoints)
    yr = 0.01*cos(i*2*pi/npoints)

    points.append(sphere(pos=vector(xr,yr,0), color=color.red, radius=5*zaxis.radius))

for j in range(npoints,2*npoints):

    xr2 = 0.005*sin(j*2*pi/npoints)
    yr2 = 0.005*cos(j*2*pi/npoints)
```

```
            points.append(sphere(pos=vector(xr2,yr2,0), color=color.red, radius=5*zaxis.radius)

        # Find net field
        for a in points:

            for i in range(nrings):
                b = rings[i] # look at one ring

                dq = charge[i]/N #charge of a piece
                dtheta = 2*pi/N #theta increment for our loop
                theta=dtheta/2 #initial theta for first piece of loop
                Ering = vector(0,0,0) #net electric field for one ring
                oofpez=9e9 #1/(4pi epsilon_0) in N m^2/C^2

                rpoint = a.pos

                scale=1.2*mag(rpoint)/32000 #used to scale the arrows representing E-field

                while theta<2*pi:
                    rpiece = R*vector(cos(theta),sin(theta),b.pos.z/R) #location of piece
                    r = rpoint-rpiece #vector from piece to point in space
                    rmag = mag(r) #magnitude of r
                    rhat = norm(r) #unit vector for r
                    dE = oofpez * dq / rmag / rmag * rhat #Electric field due to piece at rpoi
                    Enet = Enet + dE
                    particle=sphere(pos=rpiece, radius=a.radius, color=color.yellow) #draw a p
                    theta=theta+dtheta

            Evector=arrow(pos=rpoint, axis=scale*Enet, color=color.orange, shaftwidth=a.radius/
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.Javascript object>
```