Lab #1 – The Electric Field of Charged Particles

OBJECTIVES

In this lab you will:

- Practice calculating *Coulomb* Electric fields
- Learn to display the Electric field of a single particle using VPython
- Learn to display the Electric fields of multiple particles using Vpython

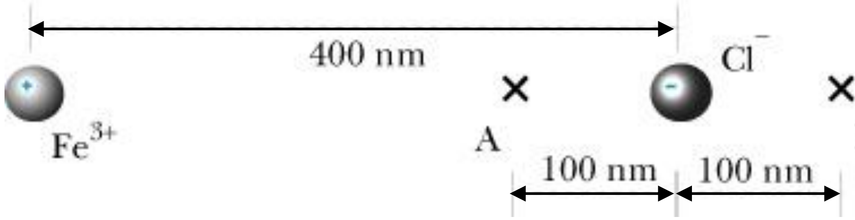
Electric fields created by single particles are both simple to envision and simple to calculate. However, the electric fields created by an arrangement of particles are much harder to visualize – and extremely tedious to calculate. VPython is an excellent tool for creating complex arrangements of particles and modeling their electric fields. VPython also provides the added benefit of visualization in 3D.

1) Warm-Up Problem

Problem (1) In the diagram below, an Fe^{3+} ion is located 400 nm from a Cl^{-} ion. *(Recall that* $1\,nm = 10^{-9}m$*)*

(a) Determine the magnitude and direction of the electric field, $\mathbf{E_A}$, at location **A**, 100 nm to the left of the Cl^{-} ion. $\mathbf{E_A} = \langle ?\,,?\,,?\,\rangle$

(b) Determine the magnitude and direction of the electric field, $\mathbf{E_B}$, at location **B**, 100 nm to the right of the Cl^{-} ion. $\mathbf{E_B} = \langle ?\,,?\,,?\,\rangle$



CHECKPOINT 1: Ask an instructor to check your work for credit.
You may proceed while you wait to be checked off.

2) Electric Field Due to a Single Particle

a) **Open IDLE for VPython. As always, you should begin the code with the following lines:**

```
from __future__ import division
from visual import *
```

The extent to which you comment your code is up to you, but you need to at least define these four sections in the following order: “Define Constants,” “Set Initial Values,” “Create Objects,” and “Calculations.”

b) Insert these titles now.

c) Define the following two constants:

```
e = 1.6e-19
oofpez = 9e9
```

The first of these constants is the fundamental unit of charge and has units of *Coulombs*. The second constant is $\frac{1}{4\pi\varepsilon_0}$ and has units of Nm^2 / C^2

d) Also in the "Define Constants" section, add the following line:

```
scalefactor = 1
```

The purpose of the scale factor will be explained later in this lab.

e) In your create objects section, type code to create the first particle (the particles will represent atoms), give the atom the charge of 3 excess protons, and create the arrow you will use to visualize the E field:

```
atom1 = sphere(pos=vector(-300e-9,0,0), radius=10e-9,color=color.red)
atom1.q = 3*e

Earrow = arrow(pos=vector(0,0,0),axis=vector(0,0,0), color=color.cyan)
```

As you may recall from Phys 172, there are three characteristics to a sphere: its position *(pos)*, radius, and color. There are also three characteristics to an arrow: the position of its tail *(pos)*, the vector property of the arrow *(axis)*, and its color. This arrow initially has no magnitude. The position of the tail of the arrow represents the ***Observation Location*** – the location at which the electric field will be calculated. From the code for Earrow, it should be evident that the *Observation Location* has been chosen to be the origin (i.e. We are going to display the Electric Field produced by atom1 at the origin)

Now you can begin your calculations to make VPython calculate the electric field at the tail of the arrow. You will use Coulomb's law to find the electric field:

$$\vec{E}_{net} = \frac{1}{4\pi\varepsilon_0}\frac{q}{r^2}\hat{r}$$

When calculating the electric field at an observation location, *the relative position vector* always points from the source particle to the observation location.

The two vector positions needed to calculate the *relative position vector* are denoted in VPython as:

```
atom1.pos
Earrow.pos
```

f) In your program, complete the code that correctly calculates the relative position vector. Name this vector "r1".

Now to find the magnitude of r1, you need to translate the Pythagorean Theorem into the syntax of VPython:

$$|\vec{r}| = \sqrt{r_x^2 + r_y^2 + r_z^2}$$

The components of a vector are attributes of the vector, so they can be referred to with dot-syntax. The components of your relative position vector are:

```
r1.x
r1.y
r1.z
```

Recall that in VPython, the operator ** raises a quantity to a power, e.g. b^3 is written as

```
b**3
```

In VPython, sqrt() is used to compute a square root, e.g. $\sqrt{3}$ should be written

```
sqrt(3)
```

g) Write a line of code to calculate the magnitude of the relative position vector. Name it "r1mag."

You can find a *unit vector* by dividing a vector by its magnitude.

h) Write a line of code to calculate the unit vector in the direction of the relative position vector. Name it "r1hat."

At this point, you have calculated r1, r1mag, and r1hat. All of these quantities are needed to calculate the Electric Field vector at the *observation location*.

i) Write down the symbolic algebraic equation used to calculate the electric field vector then convert it to VPython. Use the name "E1" for the electric field vector.

Initially, the arrow you created was temporarily assigned an *axis = vector(0,0,0).* In order to make this arrow represent the Electric Field, E1, the axis of this arrow must be reassigned to be equal to E1.

j) Enter the following line of code.

```
Earrow.axis = E1
```

k) Run the Program.

You will notice that all you see is an arrow. You may recall from Physics 172 that scale factors were used to scale vectors that did not have units of length:

$$\frac{\text{desired magnitude}}{\text{current magnitude}} = \text{scalefactor}$$

The *desired magnitude* should be inferred from sizes of and distances between the objects in your system. In the current situation, we have a sphere located 300 nm from the origin having a radius of 10 nm. Therefore, in order to accommodate an arrow within the same screen, it should have a length with order of magnitude between ten and one hundred nanometers. (e.g. 100 nm)

l) To find the *current magnitude* (The magnitude of E1), add the following print statement to the end of your code:

```
print "The magnitude of E1 field =",mag(E1),"N/C"
```

m) Now calculate an appropriate scale factor for this program, and enter that number for the value of "scalefactor," at the top of your program under the "Define Constants" section.

You MUST enter the scale factor as a number. If you enter a variable, the value of your scale factor may change during the program run and this will corrupt the physics.

n) Change the line of code you entered in step 2.j to:

```
Earrow.axis = scalefactor*E1
```

o) Run your program, and check to make sure everything looks correct.

You can tweak you scale factor to make your arrow slightly larger or smaller as necessary.

CHECKPOINT 2: Ask an instructor to check your work for credit.
You may proceed while you wait to be checked off

3) E Field Due to Several Atoms

You will now create an additional charged atom and display the ***net electric field*** at various *observation locations*. When introducing multiple particles, there may be several objects referencing the observation location. If one wishes to move the observation location, then that quantity may have to be changed in multiple places. It is convenient to define the *observation location* separately, and from then on simply reference that variable. This affords you the ease of varying the observation location while only changing one quantity:

a) Within the "Set Initial Values" section, enter the following code:

```
Obslocation = vector(0,0,0)
```

b) Then where you have previously defined Earrow, change the position of the arrow to be at the observation location. This line of code should now look like the following:

```
Earrow = arrow(pos=Obslocation, axis=vector(0,0,0), color=color.cyan)
```

c) Immediately after the lines of code creating the first atom and giving it a charge, add code to create another atom. Name this sphere "atom2" and locate it at <100e-9 0, 0>

d) Give the new particle a charge of an electron, -e.

You will need to use the ***Superposition Principle*** to find the net electric field. Therefore, you need to calculate the electric field contribution from each particle individually at the observation location – and then by adding the fields together, the net Electric Field can be obtained. You can perform each step of the calculation in groups, labeling each variable with its appropriate number, e.g.

```
r1hat=r1/r1mag
r2hat=r2/r2mag
```

Instead of using the Pythagorean Theorem to calculate the magnitude of each relative position vector, you can use the "mag" command built into VPython:

```
r1mag = mag(r1)
r2mag = mag(r2)
```

e) After you have calculated both electric fields, add a line of code to sum up the fields and find the net field. Name the net field "Enet."

f) Change your print statement to print out the net electric field:

```
print "Enet =",Enet,"N/C"
```

g) Lastly, change the arrow axis to represent the net electric field.

```
Earrow.axis=scalefactor*Enet
```

The observation location should still be at (0, 0, 0) m.

Before you run your code, can you predict what Enet will be? Draw a diagram labeling the locations of the atoms and their relative charges. Does this diagram look familiar?

h) Run your code – Adjust your scale factor using your printed value of Enet
- **Compare the Diagram and value for Enet to part (a) of the Warm-Up problem, do your values of Enet agree?**
- **Use your program to find an answer to part (b) of the Warm-Up problem. Where must you move the observation location to?**

CHECKPOINT 3: Ask an instructor to check your work for credit. You may proceed while you wait to be checked off

Save your code! You will need it for future labs!

4) Final Problem

Problem 2) **Charge 1 (12e) is located at <-500 x 10^{-9}, 0, 0> m. Charge 2 (5e) is located at <600 x 10^{-9}, 0, 0> m. The observation location is <0, 0, 0>.**

(a) Where would you place an electron so that the net E field at the observation location would be 0? *Hint: You may use your code to find the E-field produced from charges 1 and 2 at the origin. Then, what must the Electric Field of the electron be (at the origin) in order for the Net E-Field to be zero?*

(b) Add to your code an electron at the location you found in part (a). Run your code. Is the electric field zero at the origin? If not, why?

CHECKPOINT 4: Ask an instructor to check your work for credit.