Project 3

Cory Alexander Balaton & Janita Ovidie Sandtrøen Willumsen
https://github.uio.no/FYS3150-G2-2023/Project-3
(Dated: October 18, 2023)

Add an abstract for the project?

I. INTRODUCTION

We are surrounded by matter, which are made up of elementary particles. In the field of physics we want to understand the properties of these particles, measure their physical quantities, not to mention explain the origin of mass [**?**].

However, to study a particle, it is necessary isolate and contain it over time. The Penning trap is a device, able to confine charged particles for a period of time. This concept was evolved from F. M. Penning's implementation of magnetic fiels to a vaccum gauge, and J. R. Pierce's work with electron beams, and put into practice by Hans Dehmelt. In 1973 Dehmelt and his group of researchers were able to confine a particle and store it over several months [**?**].

In practice, a Penning trap is not easy to obtain, and an experiment is both time consuming and expensive. A numerical approach, allow us to study the effects of the Penning trap on a charged particle, without the cost. We can use ordinary differential equations to model the particle's movement, confined within an Penning trap. Our focus will be on an ideal Penning trap, where an electrostatic field confines the particle in z-direction, and a magnetic field confines it in the radial direction. We will use numerical methods to model a single particle, and study the particle motion in radial direction. In addition, we will model a system of particles, and study their motion both with and without particle interaction.

II. METHODS

When we study the Penning traps effect on a particle with a charge q, we need to consider the forces acting on the particle. The sum of all forces acting on the particle, is given by the Lorentz force (1).

$$\mathbf{F} = q\mathbf{E} + q\mathbf{v} \times \mathbf{B}, \tag{1}$$

We can use Newton's second law (16) to determine this sum by

$$\begin{aligned}\ddot{\mathbf{r}} &= \frac{1}{m}\sum_i \mathbf{F}_i \\ &= \frac{1}{m}(q\mathbf{E} + q\mathbf{v}\times\mathbf{B}) \\ &= \frac{q}{m}\big(\frac{V_0}{d^2}(x, y, -2z) + (B_0\dot{y}, -B_0\dot{x}, 0)\big),\end{aligned}$$

where the complete derivation can be found in . We can now write the particle's position as

$$\ddot{x} - \omega_0\dot{y} - \frac{1}{2}\omega_z^2 x = 0, \tag{2}$$

$$\ddot{y} + \omega_0\dot{x} - \frac{1}{2}\omega_z^2 y = 0, \tag{3}$$

$$\ddot{z} + \omega_z^2 z = 0, \tag{4}$$

In addition, we can find the general solution for eq. (4), when we consider the characteristic equation of a second order differential equation [**?**].

$$\begin{aligned}r^2 + \omega_z^2 = 0 \\ r = \pm\sqrt{-\omega_z^2} = \mp i\omega_z,\end{aligned}$$

two complex roots which gives us solutions in the form of

$$z = c_1 e^{i\omega_z t} + c_2 e^{-i\omega_z t}, \tag{5}$$

For a complex number $z = a + ib$, we can define $e^z \equiv e^a(\cos b + i\sin b)$ [**?**]. We can rewrite (5) as

$$\begin{aligned}c_1 e^{i\omega_z t} + c_2 e^{-i\omega_z t} &= c_1(\cos\omega_z t + i\sin\omega_z t) + c_2(\cos\omega_z t - i\sin\omega_z t \\ &= E\cos\omega_z t + iF\sin\omega_z t\end{aligned}$$

Since (2) and (3) are coupled, we want to rewrite it as a single differential equation. We can obtain this by introducing $f(t) = x(t) + iy(t)$,

$$\begin{aligned}(\ddot{x} - \omega_0\dot{y} - \frac{1}{2}\omega_z^2 x) + i(\ddot{y} + \omega_0\dot{x} - \frac{1}{2}\omega_z^2 y) = 0 \\ \ddot{x} - \omega_0\dot{y} - \frac{1}{2}\omega_z^2 x + i\ddot{y} + i\omega_0\dot{x} - i\frac{1}{2}\omega_z^2 y = 0 \\ \ddot{x} + i\ddot{y} + i\omega_0\dot{x} - \omega_0\dot{y} - \frac{1}{2}\omega_z^2 x - i\frac{1}{2}\omega_z^2 y = 0 \\ \ddot{x} + i\ddot{y} + i\omega_0(\dot{x} + i\dot{y}) - \frac{1}{2}\omega_z^2 x - i\frac{1}{2}\omega_z^2 y = 0 \text{where } i\omega_0\dot{x} + (-1)\omega_0\dot{y} = \\ \ddot{f} + i\omega_0\dot{f} - \frac{1}{2}\omega_z^2 f = 0\end{aligned}$$

Physical properties given by newtons second law (16)

$$f(t) = A_+ e^{-i(\omega_+ t + \phi_+)} + A_- e^{-i(\omega_- t + \phi_-)} \tag{6}$$

The particle moves and its position can be determined using newton. where the electric field

Algorithm

Tools

We used matplotlib

A. Units and constants

Before continuing, we need to define the units we'll be working with. Since we are working with particles, we need small units to work with so the numbers we are working with aren't so small that they could potentially lead to large round-off errors in our simulation. The units that we will use are listed in Table I.

Dimension	Unit	Symbol
Length	micrometer	μm
Time	microseconds	μs
Mass	atomic mass unit	u
Charge	the elementary charge	e

TABLE I. The set of units we'll be working with.

With these base units, we get

$$k_e = 1.3893533 \cdot 10^5 \frac{u(\mu m)^3}{(\mu s)^2 e}, \tag{7}$$

and we get that the unit for magnetic field strength (Tesla, T) and electric potential (Volt, V) are

$$\begin{aligned} T &= 9.64852558 \cdot 10^1 \frac{u}{(\mu s)e} \\ V &= 9.64852558 \cdot 10^7 \frac{u(\mu m)^2}{(\mu s)^2 e}. \end{aligned} \tag{8}$$

B. Dealing with a multi–particle system

For a multi-particles system, we need to modify $\mathbf{F}$ to account for the force of other particles in the system acting upon each other. To do that, we add another term to $\mathbf{F}$

$$\mathbf{F}_i(t, \mathbf{v}_i, \mathbf{r}_i) = q_i \mathbf{E}(t, \mathbf{r}_i) + q_i \mathbf{v}_i \times \mathbf{B} - \mathbf{E}_p(t, \mathbf{r}_i), \tag{9}$$

where i and j are particle indices and

$$\mathbf{E}_p(t, \mathbf{r}_i) = q_i k_e \sum_{j \neq i} q_j \frac{\mathbf{r_i} - \mathbf{r_j}}{|\mathbf{r_i} - \mathbf{r_j}|^3}. \tag{10}$$

Newton's second law for a particle i is then

$$\frac{d^2 \mathbf{r}_i}{dt^2} = \frac{\mathbf{F}_i \left(t, \frac{d\mathbf{r}_i}{dt}, \mathbf{r_i}\right)}{m_i}, \tag{11}$$

We can then rewrite the second order ODE from equation 11 into a set of coupled first order ODEs. We now rewrite Newton's second law of motion as

$$\begin{aligned} \frac{d\mathbf{r}_i}{dt} &= \mathbf{v}_i \\ \frac{d\mathbf{v}_i}{dt} &= \frac{\mathbf{F}_i(t, \mathbf{v}_i, \mathbf{r}_i)}{m_i}. \end{aligned} \tag{12}$$

C. Forward Euler

For a particle i, the forward Euler method for a coupled system is expressed as

$$\begin{aligned} \mathbf{r}_{i,j+1} &= \mathbf{r}_{i,j} + h \cdot \frac{d\mathbf{r}_{i,j}}{dt} = \mathbf{r}_{i,j} + h \cdot \mathbf{v}_{i,j} \\ \mathbf{v}_{i,j+1} &= \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{v}_{i,j}}{dt} = \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{F}\left(t_j, \mathbf{v}_{i,j}, \mathbf{r}_{i,j}\right)}{m}, \end{aligned} \tag{13}$$

for particle i where j is the current time step of the particle, m is the mass of the particle, and h is the step length.

When dealing with a multi-particle system, we need to ensure that we do not update the position of any particles until every particle has calculated their next step. An easy way of doing this is to create a copy of all the particles, then update the copy, and when all the particles have calculated their next step, simply replace the particles with the copies. Algorithm 1 provides an overview on how that can be achieved.

Algorithm 1 Forward Euler method

```
procedure EVOLVE FORWARD EULER(particles, dt)
    N ← Number of particles in particles
    a ← Calculate F_i/m_i for each particle in particles
    for i = 1, 2, ..., N do
        particles_i.r ← particles_i.r + dt · particles_i.v
        particles_i.v ← particles_i.v + dt · a_i
```

D. 4th order Runge-Kutta

For a particle i, we can express the 4th order Runge-Kutta (RK4) method as

$$\begin{aligned} \mathbf{v}_{i,j+1} &= \mathbf{v}_{i,j} + \frac{h}{6}\left(\mathbf{k}_{\mathbf{v},1,i} + 2\mathbf{k}_{\mathbf{v},2,i} + 2\mathbf{k}_{\mathbf{v},3,i} + \mathbf{k}_{\mathbf{v},4,i}\right) \\ \mathbf{r}_{i,j+1} &= \mathbf{r}_{i,j} + \frac{h}{6}\left(\mathbf{k}_{\mathbf{r},1,i} + 2\mathbf{k}_{\mathbf{r},2,i} + 2\mathbf{k}_{\mathbf{r},3,i} + \mathbf{k}_{\mathbf{r},4,i}\right), \end{aligned} \tag{14}$$

where

$$
\begin{aligned}
\mathbf{k}_{\mathbf{v},1,i} &= \frac{\mathbf{F}_i(t_j, \mathbf{v}_{i,j}, \mathbf{r}_{i,j})}{m} \\
\mathbf{k}_{\mathbf{r},1,i} &= \mathbf{v}_{i,j} \\
\mathbf{k}_{\mathbf{v},2,i} &= \frac{\mathbf{F}_i(t_j + \frac{h}{2}, \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{k}_{\mathbf{v},1,i}}{2}, \mathbf{r}_{i,j} + h \cdot \frac{\mathbf{k}_{\mathbf{r},1,i}}{2})}{m} \\
\mathbf{k}_{\mathbf{r},2,i} &= \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{k}_{\mathbf{v},1,i}}{2} \\
\mathbf{k}_{\mathbf{v},3,i} &= \frac{\mathbf{F}_i(t_j + \frac{h}{2}, \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{k}_{\mathbf{v},2,i}}{2}, \mathbf{r}_{i,j} + h\frac{\cdot \mathbf{k}_{\mathbf{r},2,i}}{2})}{m} \\
\mathbf{k}_{\mathbf{r},3,i} &= \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{k}_{\mathbf{v},2,i}}{2} \\
\mathbf{k}_{\mathbf{v},4,i} &= \frac{\mathbf{F}_i(t_j + h, \mathbf{v}_{i,j} + h \cdot \mathbf{k}_{\mathbf{v},3,i}, \mathbf{r}_{i,j} + h \cdot \mathbf{k}_{\mathbf{r},3,i})}{m} \\
\mathbf{k}_{\mathbf{r},4,i} &= \mathbf{v}_{i,j} + h \cdot \frac{\mathbf{k}_{\mathbf{v},1,i}}{2}.
\end{aligned}
\tag{15}
$$

In order to find each $\mathbf{k}_{\mathbf{r},i}$ and $\mathbf{k}_{\mathbf{v},i}$, we need to first compute all $\mathbf{k}_{\mathbf{r},i}$ and $\mathbf{k}_{\mathbf{v},i}$ for all particles, then we can update the particles in order to compute $\mathbf{k}_{\mathbf{r},i+1}$ and $\mathbf{k}_{\mathbf{v},i+1}$. In order for the algorithm to work, we need to save a copy of each particle before starting so that we can update the particles correctly for each step.

This approach would require 8 loops to be able to complete the calculation since we cannot update the particles until after all $\mathbf{k}$ values have been computed, however if we create a temporary array that holds particles, we can put the updated particles in there, and then use that array in the next loop, and would reduce the required amount of loops down to 4.

E. Testing the simulation

F. Relative error and error convergance rate

III. RESULTS AND DISCUSSION

From equation (6) we can find
Problem 9: we have to deal with $|\mathbf{r}| = d$

IV. CONCLUSION

APPENDIX A

Equations given

$$m\ddot{\mathbf{r}} = \sum_i \mathbf{F}_i \tag{16}$$

$$\mathbf{E} = k_e \sum_{j=1}^{n} q_j \frac{\mathbf{r} - \mathbf{r}_j}{|\mathbf{r} - \mathbf{r}_j|^3} \tag{17}$$

$$\mathbf{E} = -\nabla V \tag{18}$$

Algorithm 2 RK4 method

procedure EVOLVE RK4($particles, dt$)
  $N \leftarrow$ Number of particles inside the Penning trap
  $orig\_p \leftarrow$ Copy of particles
  $tmp\_p \leftarrow$ Array of particles of size N
  $\mathbf{k_r} \leftarrow$ 2D array of vectors of size $4 \times N$
  $\mathbf{k_v} \leftarrow$ 2D array of vectors of size $4 \times N$
  for $i = 1, 2, \ldots, N$ **do**
    $\mathbf{k}_{\mathbf{r},1,i} \leftarrow particles_i.\mathbf{v}$
    $\mathbf{k}_{\mathbf{v},1,i} \leftarrow \frac{\mathbf{F}_i}{m_i}$
    $tmp\_p_i.\mathbf{r} \leftarrow orig\_p_i.\mathbf{r} + \frac{dt}{2} \cdot \mathbf{k}_{\mathbf{r},1,i}$
    $tmp\_p_i.\mathbf{v} \leftarrow orig\_p_i.\mathbf{v} + \frac{dt}{2} \cdot \mathbf{k}_{\mathbf{v},1,i}$
  $particles \leftarrow tmp\_p$ ▷ Update particles
  for $i = 1, 2, \ldots, N$ **do**
    $\mathbf{k}_{\mathbf{r},2,i} \leftarrow particles_i.\mathbf{v}$
    $\mathbf{k}_{\mathbf{v},2,i} \leftarrow \frac{\mathbf{F}_i}{m_i}$
    $tmp\_p_i.\mathbf{r} \leftarrow orig\_p_i.\mathbf{r} + \frac{dt}{2} \cdot \mathbf{k}_{\mathbf{r},2,i}$
    $tmp\_p_i.\mathbf{v} \leftarrow orig\_p_i.\mathbf{v} + \frac{dt}{2} \cdot \mathbf{k}_{\mathbf{v},2,i}$
  $particles \leftarrow tmp\_p$ ▷ Update particles
  for $i = 1, 2, \ldots, N$ **do**
    $\mathbf{k}_{\mathbf{r},3,i} \leftarrow particles_i.\mathbf{v}$
    $\mathbf{k}_{\mathbf{v},3,i} \leftarrow \frac{\mathbf{F}_i}{m}$
    $tmp\_p_i.\mathbf{r} \leftarrow orig\_p_i.\mathbf{r} + dt \cdot \mathbf{k}_{\mathbf{r},3,i}$
    $tmp\_p_i.\mathbf{v} \leftarrow orig\_p_i.\mathbf{v} + dt \cdot \mathbf{k}_{\mathbf{v},3,i}$
  $particles \leftarrow tmp\_p$ ▷ Update particles
  for $i = 1, 2, \ldots, N$ **do**
    $\mathbf{k}_{\mathbf{r},4,i} \leftarrow particles_i.\mathbf{v}$
    $\mathbf{k}_{\mathbf{v},4,i} \leftarrow \frac{\mathbf{F}}{m}$
    $tmp\_p_i.\mathbf{r} \leftarrow orig\_p_i.\mathbf{r} + \frac{dt}{6} \cdot (\mathbf{k}_{\mathbf{r},1,i} + \mathbf{k}_{\mathbf{r},2,i} + \mathbf{k}_{\mathbf{r},3,i} + \mathbf{k}_{\mathbf{r},4,i})$
    $tmp\_p_i.\mathbf{v} \leftarrow orig\_p_i.\mathbf{v} + \frac{dt}{6} \cdot (\mathbf{k}_{\mathbf{v},1,i} + \mathbf{k}_{\mathbf{v},2,i} + \mathbf{k}_{\mathbf{v},3,i} + \mathbf{k}_{\mathbf{v},4,i})$
  $particles \leftarrow tmp\_p$ ▷ Final update

$\mathbf{F}$ in the algorithm does not take any arguments as it uses the velocities and positions of the particles inside the array *particles* to calculate the total force acting on particle i.

APPENDIX B

Sum of all forces

$$sum$$

We find the physical coordinates from $x(t) = \mathrm{Re} f(t)$ and $y(t) = \mathrm{Im} f(t)$, where $f(t)$ is given by equation (6).

We can rewrite $f(t)$ using the definition as

$$
\begin{aligned}
f(t) = &A_+(\cos\omega_+ t + \phi_+ - i\sin\omega_+ t + \phi_+) \\
&+ A_-(\cos\omega_- t + \phi_- - i\sin\omega_- t + \phi_-)
\end{aligned}
\tag{19}
$$

If we rearrange the right side of equation (19), we find the physical coordinates

$$x(t) = A_+(\cos\omega_+ t + \phi_+) + A_-(\cos\omega_- t + \phi_-) \tag{20}$$

$$y(t) = -A_+(i\sin\omega_+ t + \phi_+) - A_-(i\sin\omega_- t + \phi_-) \tag{21}$$