# 1 Hidden Charge

1.1 Introduction

An unknown point charge Q is fixed in a region of space. Electrons launched parallel to the z axis far from the charge will scatter electrostatically off of the fixed charge and strike a detecting screen. It is possible learn about the details of the hidden charge by varying the initial kinetic energy as well as the initial x_i and y_i coordinates of the electron beam and measuring the final coordinates x_f and y_f of where an electron strikes a finite flat screen perpendicular to the z axis and located at $z = 0$.

It is useful to know the Rutherford scattering formula,

$$b = \frac{kqQ}{2E} \frac{1}{\tan(\theta/2)}$$

where b is the impact parameter, E is the energy of the electron, $q = -1.602 \times 10^{-19}$C is the charge of the electron, $k = 8.99 \times 10^9$ Nm^2/C^2, and θ is the scattering angle. The impact parameter is defined as the closest approach of the electron to the target, assuming that the electron were unaffected by the target and hence would move in a straight line; the scattering angle is angle between the original velocity vector of the electron far from the target and the final velocity vector of the electron far from the target after scattering.

electron trajectory
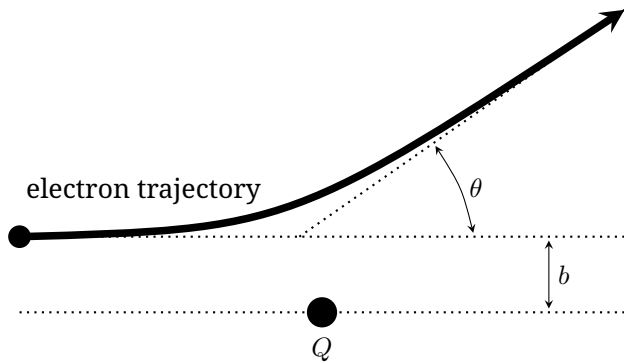
1.2 Task

The task is to determine the position (x_Q, y_Q, z_Q) and also the magnitude and sign of the fixed charge Q, as precisely as possible. You should provide rough, order of magnitude error estimates on these results. There is Gaussian error associated with initial beam location that is on the order of 0.5 mm.

As with all experiments, you must provide clearly labelled tables of data, clearly labelled graphs, and sufficient formulae derivations to make it clear what you have measured, and how you are deriving your results.

1.3 Program Interface

The program asks for an accelerating voltage with the prompt

`Beam accelerating voltage in V:`

Enter a number between 1 and 10000, and press **return**. The program then asks for the initial launch coordinates, starting with x_i, with the prompt

`x-coordinate of the electron beam in cm:`

Enter a number between -20 and 20 and then press **return**. Finally, the program asks for y_i, with the prompt

`y-coordinate of the electron beam in cm:`

Enter a number between -20 and 20 and then press **return**. If you enter an invalid number for any of these three, the program will prompt you with

`Invalid entry.`

and will then prompt you for the value again, reminding you of the allowed limits.

After the three numbers have been entered, the program will output

`Electron beam fired with parameters (x, y, V) =`

and it will restate your entered values, and then

`Electron detected at (x, y) =`

and give the screen location of the detected electron. However, if the electron misses the finite size screen, you will be told

`Electron not detected...`

The program then repeats, allowing you to enter in a new set of initial coordinates.

2 Black box

2.1 Introduction

You have a rigid mechanical black box consisting of a container of mass m_1. Inside the container there is a load of mass m_2 that hangs on an effectively massless spring of stiffness k_1 from the ceiling of the box. Another mass m_3 is hanged to the mass m_2 via another massless spring of stiffness k_2. There is a small viscous drag which depends on the velocity of the objects. The gravity of Earth is $g = 9.81$ m/s^2 and is parallel to the sides of the box.
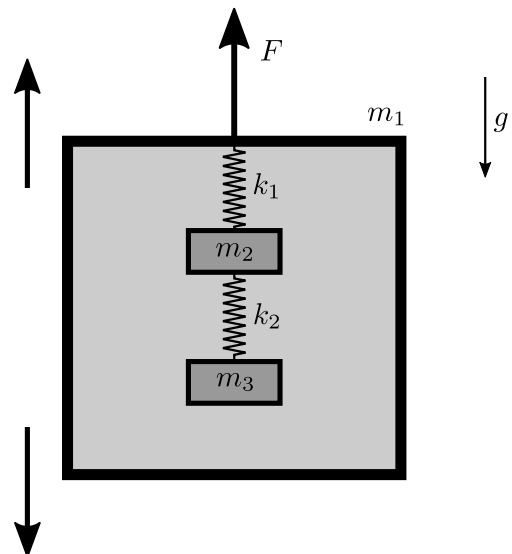
The box can be moved up or down with a piece-wise constant acceleration. The acceleration pattern can be programmed through input by giving the duration (in seconds) and acceleration (in m/s^2) for each step. The simulation shows in "real time" the force F exerted to

the box that is needed to maintain the given acceleration at the current moment of time, together with the reading of time. The simulation will also output the readings to a text file in the same folder as the program. All simulations will always start with the same initial configuration for the masses.

Note: Every measurement of force F has a small random error. The springs are linear for reasonably small deformations, but nonlinear for larger deformations. The values k_1 and k_2 are defined to be the stiffness of each spring for small deformations near equilibrium when the box is at rest. Force F and acceleration are considered to be positive if they are directed upwards. The side length of the box is 0.6 m and the box is initially in the middle of a room with height 3 m. An experiment ends automatically if the box hits the ceiling or the floor, or if any of the masses collide with the box or with the other mass. The figure is not drawn to scale.

2.2   Task

The task is to determine all the parameters: m_1, m_2, m_3, k_1, k_2. You do not need to provide an error analysis for these results.

As with all experiments, you must provide clearly labelled tables of data, clearly labelled graphs, and sufficient formulae derivations to make it clear what you have measured, and how you are deriving your results.

2.3   Program Interface

Initially, the program asks for a sequence of input actions. You have the following possibilities.

- Enter two numbers and press **return** to add a step to the acceleration pattern, for example: `1.5 -0.4`
  The first number should be the **duration** of the step in seconds (must be a multiple of 0.01 s) and the second number should be the **acceleration** in m/s^2 (must be between -30 and 30).
- Enter `repeat` and an integer and press **return** to repeat actions, for example: `repeat 10`
  The integer should be the **number of times** you want to repeat actions. Every repeat action should end with an `endrepeat` action (see below).
- Enter `endrepeat` to end repeating actions. If you start the experiment, all actions between `repeat` and `endrepeat` will be repeated a given number of times. You cannot repeat actions inside another repeat.
- Enter `sample` and a number and press **return** to change the sampling time, for example: `sample 0.4`
  The number should be the new **sampling time** which is the time after which every new reading is output to the text file. The sampling time must be a multiple of 0.01 s, which is also the default sampling time.
- Enter `begin` to finish the sequence and start the experiment.

You can also write multiple actions on the same line and then press **return**. For example, you can enter

```
sample 0.4 repeat 10 1.5 0.4 1.5 -0.4
endrepeat begin
```

to start an experiment where you change the sampling time to 0.4 s and accelerate the box respectively with $a = 0.4$ m/s^2 and $a = -0.4$ m/s^2 ten times.

If you enter an invalid input, you will get one of the following error messages and you can try to enter an action again.

- If acceleration is out of range:
  `Acceleration is out of range.`
- If duration of acceleration is out of range:
  `Duration is out of range.`
- If sampling time is out of range:
  `Sampling time is out of range.`
- If the number of repeat times is out of range:
  `Number of repeat times is out of range.`
- If you try to repeat actions inside another repeat action:
  `Cannot repeat actions inside another repeat.`
- If you try to end repeat without a repeat action to end:
  `Cannot end repeat outside repeat.`
- In all other cases:
  `Invalid entry.`

After you enter `begin`, the program will ask you for a name for the output file with the prompt

`Enter name for output file (e.g. "results"). You should use Latin letters and numbers because some special characters are not allowed.`

Enter a name and press **return**. You are advised to use only Latin letters and numbers for the name. Other characters may or may not be allowed in the filename and in case of an invalid filename the readings will not be saved. The readings will be saved in a `.txt` file with the given name in the same folder as the program.

After this, the program will display

`Begin experiment.`

and start the experiment. The program will then display the current time since the beginning of the experiment (`Time (s)`), measured value of force F (`Force (N)`) and acceleration of the box (`Accel (m/s^2)`). The readings will be similarly displayed in the text file.

The program will then display one of the following messages.

- If the experiment ended successfully:
  `Experiment ended successfully.`
- If the box hit the ceiling:
  `The box hit the ceiling. Experiment ended.`
- If the box hit the floor:
  `The box hit the floor. Experiment ended.`
- If the masses inside the box collided or one of the masses inside the box collided with the box:
  `Masses and/or the box collided. Experiment ended.`

After the experiment ends, you can start another experiment.