PHYS 381 – Computational Physics I (Winter 2023)

Assignment #1: Finding Minima of Functions

Due date: 30 January 2023, 3 pm

Group members (add your name and UCID below):

Member #1: David (Scott) Salmon - 30093320

Member #2: Gradyn Roberts - 30113797

Member #3: Gabriel Komo - 30164690

Please, only add text in this report template using Times New Roman font, non-italic, non-bold, font size 12.

Authors' contributions:

Provide a brief description of how the work of this assignment was distributed among group members. Assignment workload distribution must be balanced among group members, i.e., all group members should be involved equally with coding and written materials for the report. **Authors' contribution** description cannot go over page 1 of this template.

- All 3 of us made contributions throughout the lab report. We all met outside of class on two occasions to work together on the report. All 3 of us worked on the code together during these time periods. For the report itself, Gabe and Scott wrote the Abstract together, Gabe wrote the introduction, Grady wrote the conclusion, and then all 3 of us worked together for the Data and Analysis section.

- Throughout the whole lab we all communicated constantly, and it was a fairly well distributed work distribution.

Abstract (0.5 points):

Provide a summary of your assignment work and findings. The abstract is a paragraph where you summarize what you studied in this computational lab and what are your main findings. Please, stay on the limit size of up to 300 words max. **Abstract** cannot go over page 2 of this template.

In this lab, the main topic of study is how to utilize several numerical methods for finding the roots of functions, as well as their minima and maxima, and then how to write python scripts to compute these points.

For the first part of the lab, a python script is set up with the capabilities to find the roots of a parabolic function using what is called the bisection method. For the second part of the lab, another python script was set up to find the roots of the same parabolic function except this time instead of using the bisection method the Newton-Raphson method was used instead. Finally, for the last section of the lab was mostly an application of the methods learned from the first two scripts to solve a physics problem computationally. We were to use a slight variation of the Newton-Raphson method to determine where the minimum point was for a given potential energy function, and then using this point we were able to determine what the minimum potential energy was of the given function.

Introduction (0.5 points):

Provide an overview of the topics studied in the assignment. Concentrate on the ‘physics’ part of the assignment, in this case, potential energy and stable equilibrium analysis of physical systems. **Introduction** cannot go over page 3 of this template.

A property that a physical particle can hold is its potential energy. In mechanics, this value often varies as a function of position. If the potential energy of a particle is to be plotted against its positions, the extrema (the positions in which the slope of the function is 0), are considered equilibrium points. Extrema that are maxima (points in which the slope goes from positive to negative, or ones where the second derivation of the function is negative), are considered unstable equilibrium points. This is because if the particle is disturbed from an equilibrium point of this type, it will tend away from the point. Stable equilibriums are quite the opposite. They are minima of the function of potential energy versus position, meaning the slope goes from negative to positive around the point. Also, if a particle is displaced from a point of stable equilibrium, it will tend to return to it.

A physical example of a one dimensional system in which potential energy is a function of position that is explored in this lab is that of two ions. Specifically, Na+ and Cl-. the potential energy based on positions is derived from two terms. The first being the Pauli repulsion of electron clouds. This repels the particles, as the Pauli exclusion principle essentially states that two fermions cannot be in the same state at once, so they are repelled from each other. The second is the electrostatic attraction, which brings the ions together as they are of opposite charges. This is subtracted from the first term, as they are in opposite directions; potential energy from the electrostatic attraction decreases as x approaches 0, and potential energy from Pauli repulsion increases.

In the above example, there is a stable equilibrium point in which the slope of potential energy is equal to 0. This is the point in which the slope of both terms (the electrostatic attraction and Pauli repulsion), otherwise written as their derivatives, are equal and opposite to each other. Since the derivative of energy with respect to position is force, the equilibrium points in this case are when the electrostatic force and the force from Pauli repulsion are equal and opposite.

Methods (1 point):

Describe the used numerical methods. Provide your own explanations about the two numerical methods used in this assignment: bisection (page 4) and Newton-Raphson (page 5). **Methods** cannot go over page 5 of this template.

- The bisection method for determining the roots of a quadratic works as follows:
    1. Two points must be chosen, the first point must make the given function result in a negative number, i.e. f(x1) < 0. The second point must make the given function result in a positive number, i.e. f(x3) > 0. The reason for this is to forcibly choose two points with one being on each side of a root.
    2. Next, a third point is created as an average of the two inputted values, i.e. x2 = (x1 + x3) / 2.0.
    3. Now, calculate f(x2). If the result is a positive number, then set x3 to x2 and repeat step 2 to create a new x2 and then do step 3 again. If the result is a negative number, then set x1 to x2 and then repeat step 2 to create a new x2 and repeat step 3.
    4. Every time you do this, x2 will become smaller and smaller, becoming closer to 0 every "iteration" or "step".
    5. Continue doing this until the magnitude of f(x2) is extremely small (i.e. smaller than a specific tolerance variable defined in the code).
    6. The resulting x2 variable is now a very close approximation to the real root.

- You can repeat all the same steps except change the initial values of x1 and x3 to find the other root for a quadratic.
- The magnitude of f(x1) and f(x3) in step 1 do not matter - as long as f(x1) produces a negative number and f(x3) produces a positive number. However, choosing values closer to the root for x1 and x3 makes the program have to go through fewer "steps" or rounds of calculations to determine the root.

Methods:

Describe the used numerical methods. Provide your own explanations about the two numerical methods used in this assignment: bisection (page 4) and Newton-Raphson (page 5). **Methods** cannot go over page 5 of this template.

- The Newton-Raphson method for determining the roots of a quadratic works as follows:
    1. Choose a point that appears to be “close” to the root. Let's denote this point as x_o.
    2. Find the result of the function with x_o as its input, i.e. $f(x_o)$
    3. Now, find the derivative function and again find the result with x_o as its input, i.e. $f'(x_o)$
    4. Now, divide the $f(x_o)$ result by the $f'(x_o)$ result.
    5. Now subtract the result from step 4 from x_o. i.e. $x = x_o - [f(x_o)/f'(x_o)]$
    6. Now you have a new approximated root that is closer to the real root then the initial point given!
    7. Repeat the process with the new approximated root in place of the initial value inputted in step 1 to create an even more accurate approximation. Continue repeating this process until it converges to a fixed point.

- Similar to the bisection method, the magnitude of the initial value given does not matter, however choosing a value that is closer to the real root will make the program go through much fewer “steps” or rounds of calculations to determine the root.

Code workflow (1 point):

Explain how the code you used in exercise 5(e) works. Do not simply copy and paste lines from your code. Use your own words to explain its procedures and algorithm. **Code workflow** cannot go over page 6 of this template.

- The way the minimum value was calculated was through using a slightly modified version of the Newton-Raphson method. The biggest difference this time compared to the NR method discussed above is that instead of doing $x = x_o - [f(x_o)/f'(x_o)]$ as outlined earlier in the report, we are doing $x = x_o - [f'(x_o)/f''(x_o)]$ instead.

- This is because this time we are not using the NR method to find the roots of the original function, but rather the roots of the derivative function. As we learned in Calculus 1, the roots of the derivative function are critical points in the original function, so the NR method can be used to find the roots of the derivative function and therefore the minima point of the original function.

- To do this, we need to determine what U'(x) and what U''(x) are. All variables are constant besides x, so this is a pretty straightforward derivative as shown below, using a chain rule in the first term both times and the power rule for the second term both times.

$$U(x) = Ae^{(-x/p)} - \frac{e^2}{4\pi\epsilon_o}\frac{1}{x}$$

$$U'(x) = -\frac{A}{p}e^{(-x/p)} + \frac{e^2}{4\pi\epsilon_o}\frac{1}{x^2}$$

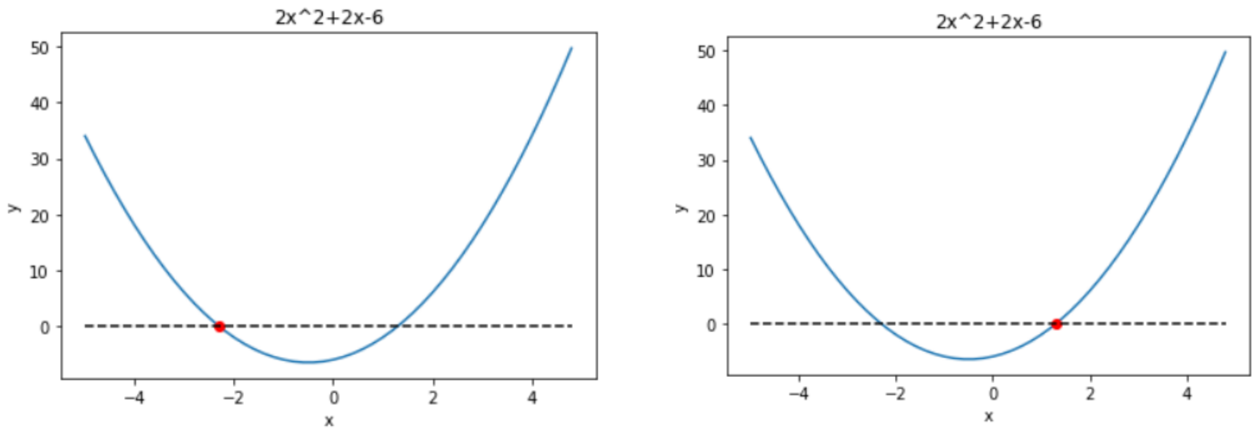
$$U''(x) = \frac{A}{p^2}e^{(-x/p)} - 2\frac{e^2}{4\pi\epsilon_o}\frac{1}{x^3}$$

- The code itself is pretty straightforward after you determine the derivative functions. We set the initial minimum value to 0.2 as outlined in the lab manual, and then ran a while loop that kept doing the NR method until the calculated root was less than 0.0001. We then calculated what the potential energy was at the minimum point, and then plotted the graphs for the regular function and the derivative function as a function of x.

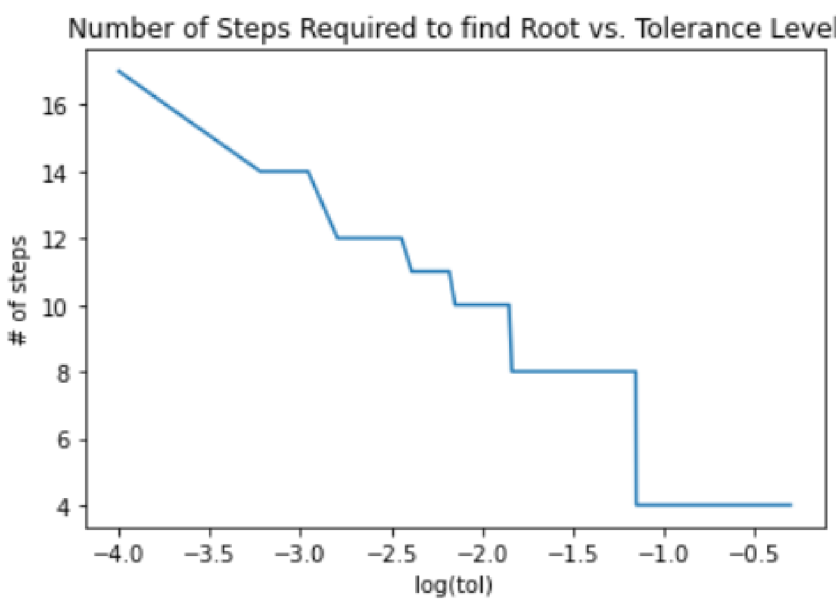
- The results generated by the code in 5e) are discussed in the results and analysis section below with the figures generated.

Results and analysis (1 point):

Present the plots requested in items 3(k), 4(k), and 5(e). Discuss and explain the plots and their results. **Results and analysis** cannot go over page 10 of this template. Do not insert too large figure panels just to cover space. Choose figure sizes typically seen in textbooks and scientific manuscripts.



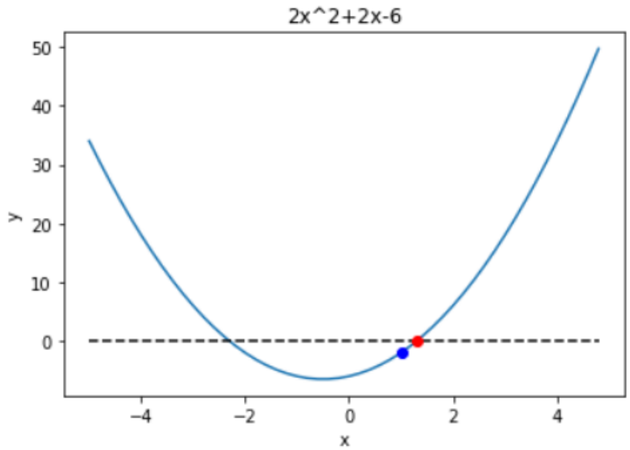
- Above are two graphs of the parabolic function $2x^2+2x-6$. The part of the assignment these plots are generated in is 3e) using the bisection method outlined above. The left figure is showing the location of the left root for the function, while the right figure is showing the location of the right root of the function.

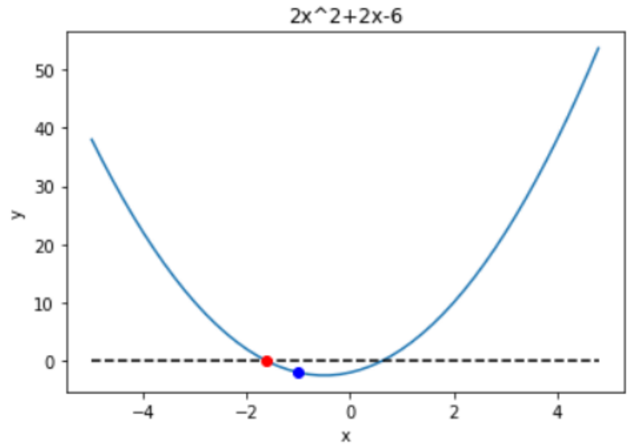


- This graph is showing the amount of iterations that were required for the calculated root to be less than the input tolerance value. This graph was generated in 3j).
- The initial tolerance value was 0.0001, hence why the first data point is at log(tol) = -4.0. The final data point is collected at tol = 0.5, or about log(tol) ≅ -0.3010. As can be seen, when the tolerance value is increased, the number of steps required to calculate a root that is less than the tolerance value decreases.

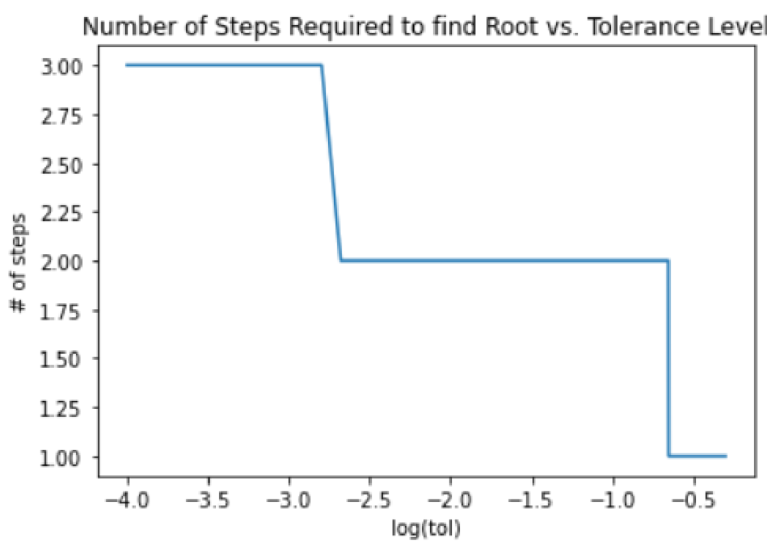
Results and analysis (1 point):

Present the plots requested in items 3(k), 4(k), and 5(e). Discuss and explain the plots and their results. **Results and analysis** cannot go over page 10 of this template. Do not insert too large figure panels just to cover space. Choose figure sizes typically seen in textbooks and scientific manuscripts.





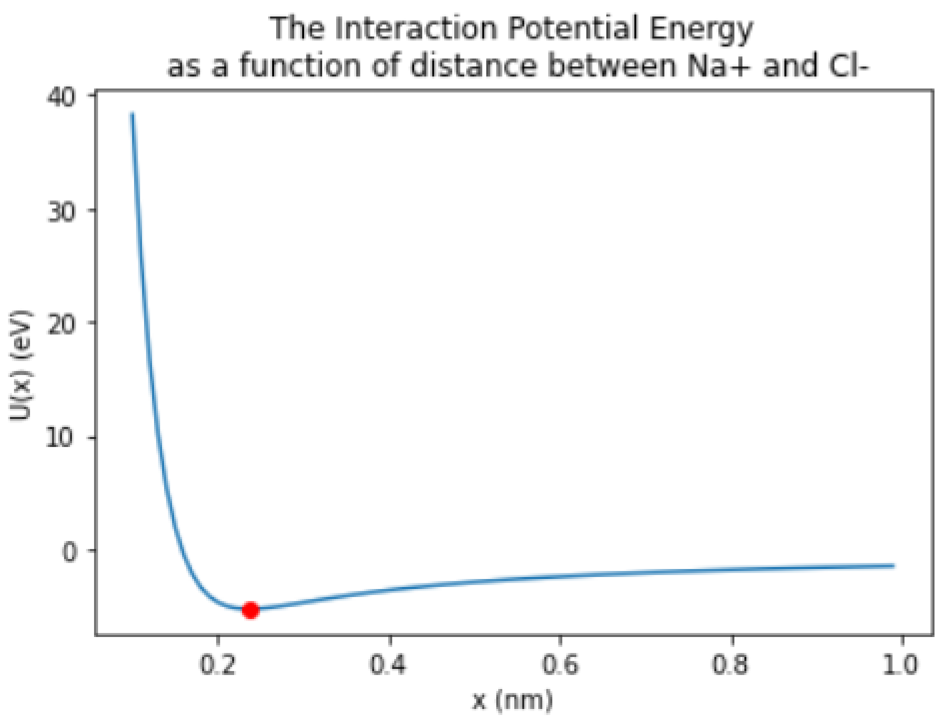
- Above are two graphs of the parabolic function $2x^2+2x-6$. The part of the assignment these plots are generated in is 4d) using the Newton-Raphson method outlined above. The left figure is showing the location of the left root for the function, while the right figure is showing the location of the right root of the function. The red dots are the calculated roots, while the blue dots are the initial x_o values used to generate the root. $x_o = 1$ on the left figure, and $x_o = -1$ for the second figure.



- This graph is showing the amount of iterations that were required for the calculated root to be less than the input tolerance value. This graph was generated in 4i).
- The initial tolerance value was 0.0001, hence why the first data point is at log(tol) = -4.0. The final data point is collected at tol = 0.5, or about log(tol) ≅ -0.3010. As can be seen, when the tolerance value is increased, the number of steps required to calculate a root that is less than the tolerance value decreases.

Results and analysis (1 point):

Present the plots requested in items 3(k), 4(k), and 5(e). Discuss and explain the plots and their results. **Results and analysis** cannot go over page 10 of this template. Do not insert too large figure panels just to cover space. Choose figure sizes typically seen in textbooks and scientific manuscripts.



- The graph above is the calculated Interaction Potential energy as a function of the distance between the Na+ and Cl- ions. This graph is based on the formula given in the lab manual shown below:

$$U(x) \; = \; Ae^{(-x/p)} - \frac{e^2}{4\pi\epsilon_o}\frac{1}{x}$$

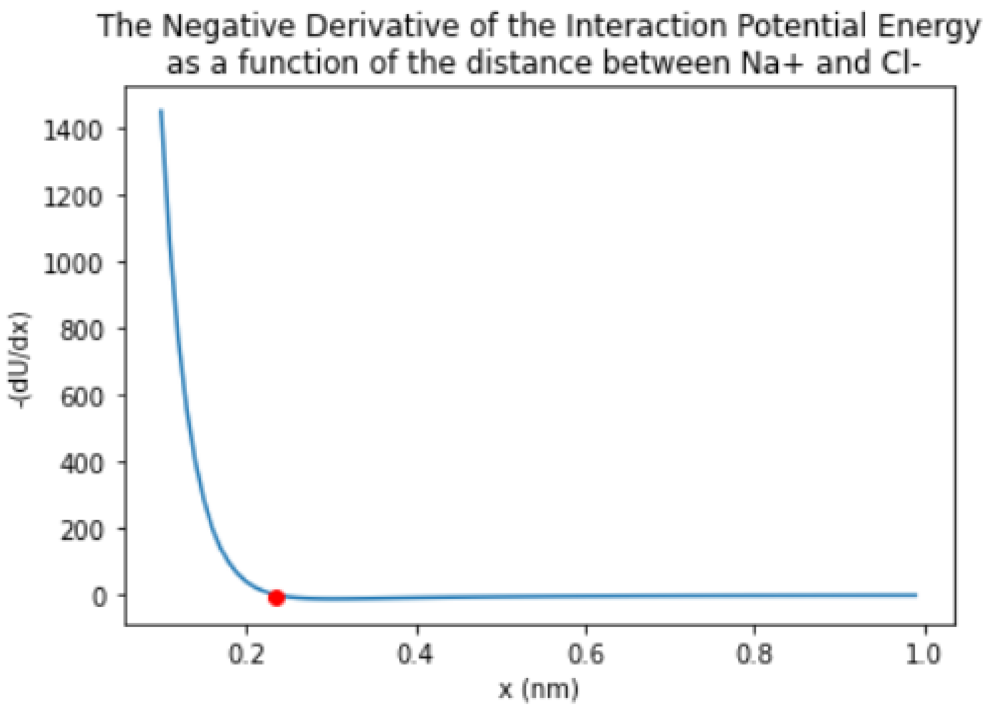
Where:

- $e^2/4\pi\epsilon_o$ = 1.44eV*nm
- A = 1090eV
- p = 0.033nm

- The calculated minimum point was x = 0.2360538484nm, and U(x) at this point is equal to -5.2474891185 eV. The red dot on the graph is the minimum point of the function.

Results and analysis (1 point):

Present the plots requested in items 3(k), 4(k), and 5(e). Discuss and explain the plots and their results. **Results and analysis** cannot go over page 10 of this template. Do not insert too large figure panels just to cover space. Choose figure sizes typically seen in textbooks and scientific manuscripts.



- This is the graph of the negative of the derivative of the interaction potential energy as a function of the distance between Na+ and Cl-. The function becomes 0 at exactly x = 0.2360538484nm, and that's why it must be an extreme point in the original function.

Conclusions (0.5 points):

Wrap up the assignment by highlighting your main findings and results. Please, stay on the limit size of up to 300 words max. **Conclusions** cannot go over page 11 of this template.

In this lab, we demonstrated two ways to use python programming to solve physics related problems. The bisection method and the Newton-Raphson method had different approaches to solving problems in different scenarios.

We concluded that the iterative process of the bisection method works well for finding the roots of a function, but requires a longer process which may be harder to implement at a large scale. However, this method is easy to understand and requires less mathematics knowledge. This is contrasted by the Newton-Raphson method which can find the roots of a function very quickly and precisely. The Newton-Raphson method does not use an iterative process to find the roots for a function, making it able to find the roots of a function in less steps; this makes it better for large scale implementation. The downside of this method is that it requires derivatives of the function to be found to create highly accurate models, which can sometimes be difficult to calculate. This can be especially difficult if a function is complicated or not precisely defined.

Overall, both methods provide accurate ways to find the roots of a given function, and demonstrate ways to use the tools in python to solve problems related to physics.

References:

Include any citations or references used during the preparation of this report and codes. Consulted webpages should also be cited as well as our assignment manuals. Choose a citation style of your preference and adopt it consistently throughout. There is no page limit for the **References** section.

1. Assignment #1 Manual: Finding Minima of Functions - PHYS 381 Winter term 2023 - Retrieved from https://d2l.ucalgary.ca/d2l/le/content/497542/viewContent/5727190/View

Other (0.5 points):

This page is to be filled by the instructor or TAs ONLY.

Remaining 0.5 points are granted for following the template and overall quality of the report.

Was the assignment submitted before the due date (mark one option)? YES   NO