Problem Set 3 Latex Report

Eren Tuksal

September 26, 2023

1 Problem 1

1.1 Part a

The derivative we numerically calculated is not precisely the same as the analytical derivative because for the analytical derivative $\delta \to 0$ while our numerical derivative uses $\delta = 10^-2$

1.2 Part b

In [8]:
```
print(1-fp_num(1,10**-4))
print(1-fp_num(1,10**-6))
print(1-fp_num(1,10**-8))
print(1-fp_num(1,10**-10))
print(1-fp_num(1,10**-12))
print(1-fp_num(1,10**-14))
```

```
-9.999999988985486e-05
-9.99917733279787e-07
-3.922528746258536e-09
-8.284037100736441e-08
-8.890058334132256e-05
0.0007992778373491216
```

Figure 1: Figure 1: The difference of the numerical derivative from analytical

As seen in Figure 1 above, making δ closer to 0 helps initially, but after $\delta = 10^-8$, the error in the numerical derivative starts growing larger again. This is due to the error of the computer adding floats

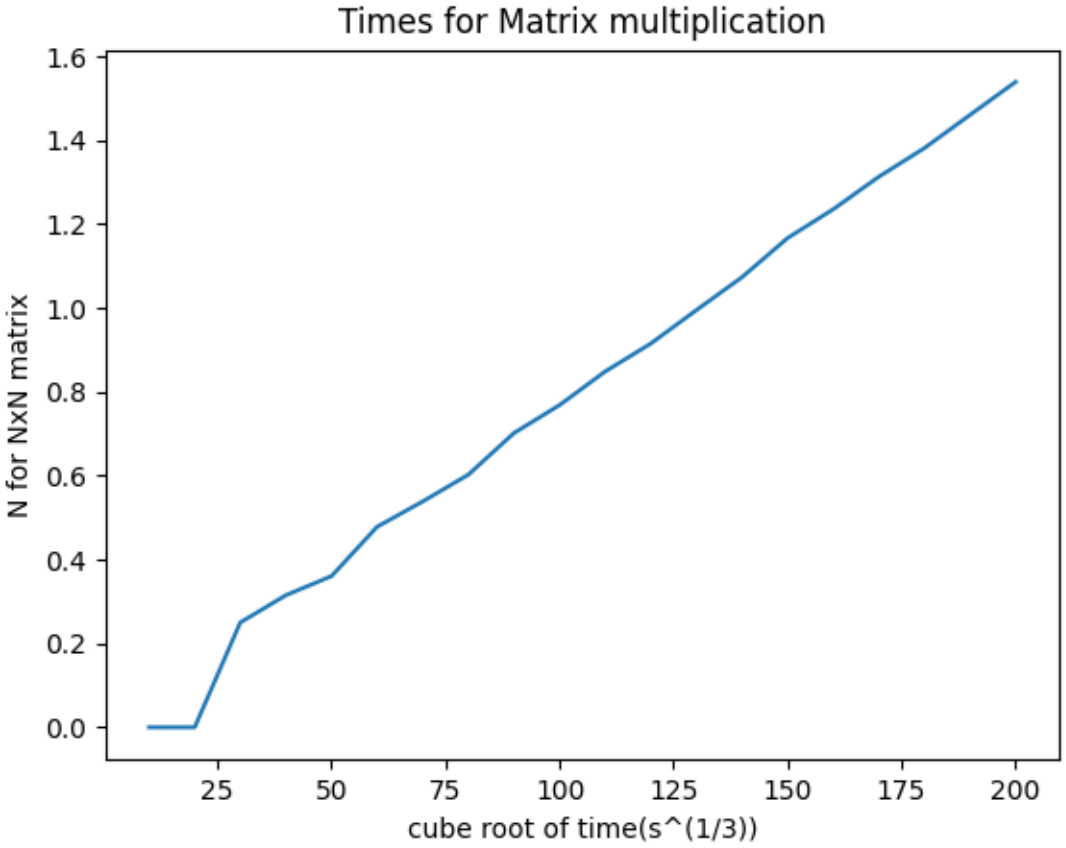

Figure 2: Figure 2: N vs. the cube root of time the multiplication takes to execute

2 Problem 2

2.1 Using self defined Matrix Product

As seen in Figure 2 above the cube root of the time the multiplication takes to execute vs. the size of the matrix is roughly a diagonal line, showing that, as suspected, the time it taxes to execute the matrix multiplication goes proportional to the size cubed.

2.2 Using dot module of numpy

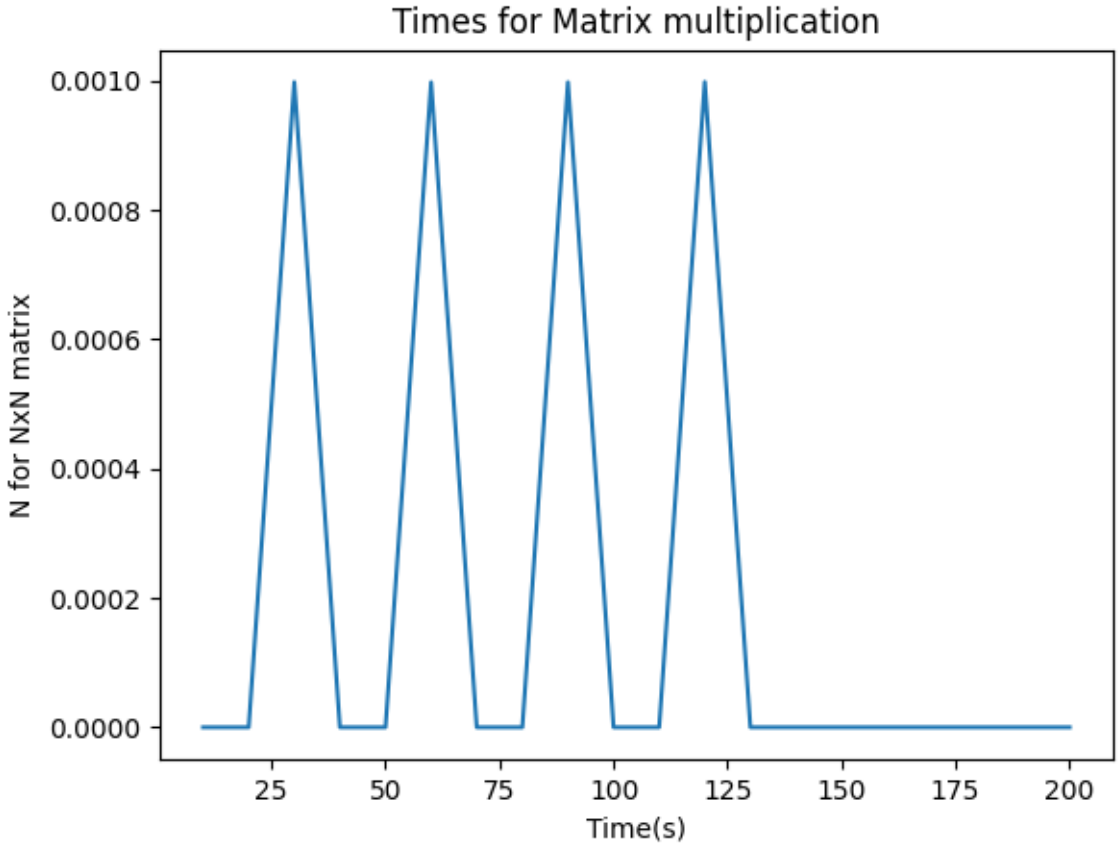

Figure 3: N vs. the time it takes to execute the dot module of numpy
As seen in Figure 3 the dot module of NumPy does not depend cubically on the size of the matrices and takes overall much shorter than the user-defined matrix product .

3 Problem 3

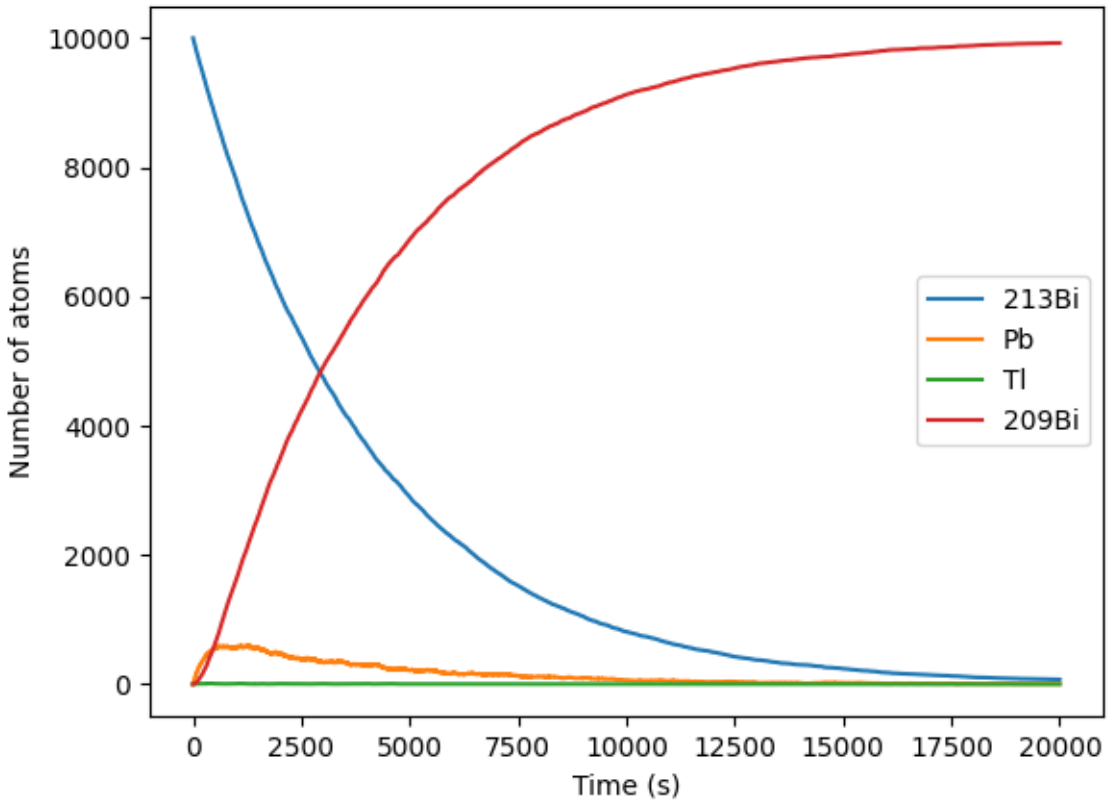

Figure 4: Decay of 213Bi into 209Bi

4 Problem 4

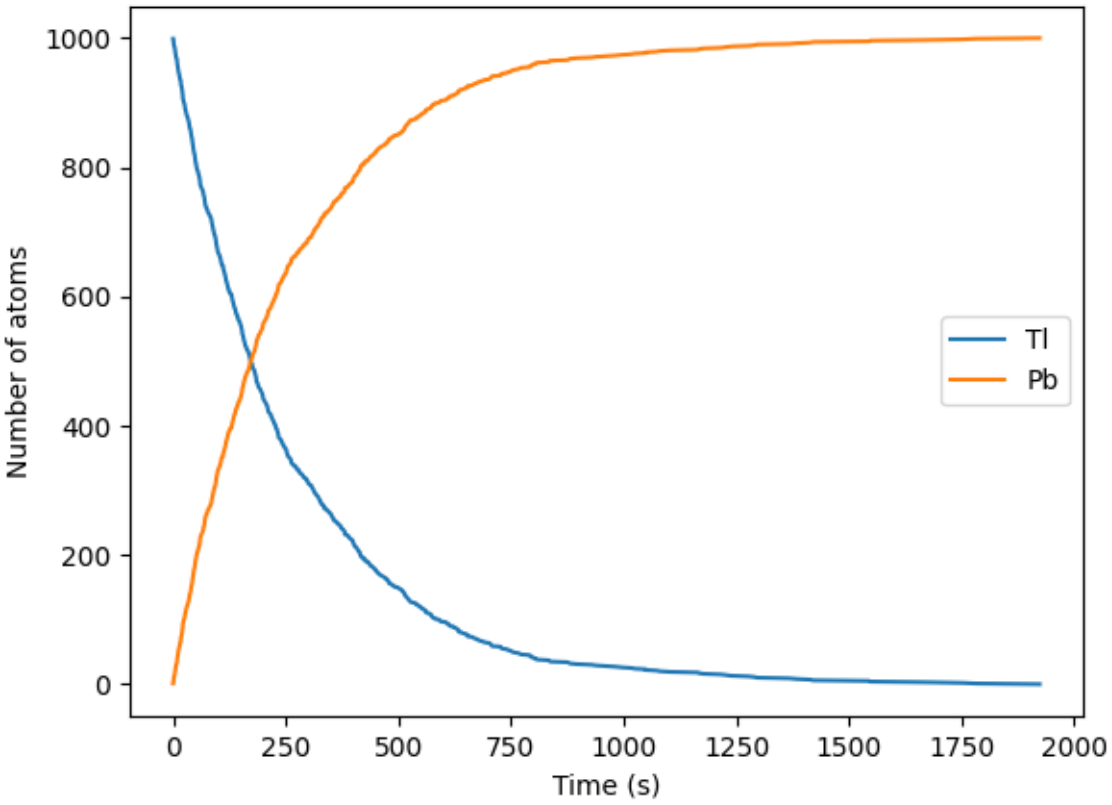

Figure 5: Decay of Tl into Pb