HW2_physics466_Atomic_Sacle_Simulation Jinsheng Wang NetID: jwang278

Periodic Boundary Conditions

1. Check the correctness of function pos_in_box() (code has been send by compass/email)
Input:

```
box_length = 1.0
pset = ParticleSet(1)
pset.change_pos(0,np.array([0.501,0.499,0.501]))
print( pos_in_box(pset.pos(0),box_length) )
pset.change_pos(0,np.array([1.000, -.501, 4.501]))
print( pos_in_box(pset.pos(0),box_length) )
```

get result:

```
[-0.499  0.499 -0.499]
[ 0.     0.499 -0.499]
```

means that pos_in_box() function is implemented correctly.

Minimum Image Convention

2. Check the correctness of function distance()(code has been send by compass/email)
Input:

```
box_length = 1.0
pset = ParticleSet(2)
pset.change_pos(0,np.array([ 0.499, 0.000, 0.000 ]) )
pset.change_pos(1,np.array([ -.499, 0.000, 0.000 ]) )
print( distance(0,1,pset,box_length) )
pset.change_pos(0,np.array([ 0.001, 0.000, 0.000 ]) )
pset.change_pos(1,np.array([ -.001, 0.000, 0.000 ]) )
print( distance(0,1,pset,box_length) )
```

output:

```
0.002
0.002
```

means that distance() function is implemented correctly.

3. implement displacement() function (code has been send by compass/email)

The Lennard-Jones Fluid
Calculating Pair Potentials

4. Check the correctness of function internal_force() (code has been send by compass/email)

Integrator

5. Prove that the Verlet algorithm is time-reversal invariant. That is, show that if we use r(t+h) and r(t) as inputs we get r(t-h) (up to round-offerrors).

Prove:

$$r(t+h) = r(t) + v(t)*h + 1/2*a(t)*h*h + b(t)*h*h*h + O(h^4) \quad (1)$$
$$r(t-h) = r(t) - v(t)*h + 1/2*a(t)*h*h - b(t)*h*h*h + O(h^4) \quad (2)$$

use eq.(1) and eq.(2) we can get that

$$r(t-h) = 2*r(t) - r(t+h) + a(t)*h*h + O(h^4) \quad (3)$$ # question proved

6. State two possible problems with finite-precision arithmetic about the Verlet time-stepping algorithm

problems:

(1) There will be cut-off errors given O(h^4) term if we ignore it in calculation, which will in turn cause systematic errors.

(2) r values are in fact quite small, if subtraction is performed, then round-off errors could be significant.

7. Prove that velocity Verlet is equivalent to Verlet. Which is preferable?

Veloctity-Verlet method:

$$r(t+h) = r(t) + h\,[\, v(t) + (h/2)\, a(t)] \quad (4)$$
$$v(t+h/2) = v(t) + (h/2)\, a(t) \quad (5)$$
$$v(t+h) = v(t+h/2) + (h/2)\, a(t+h) \quad (6)$$

Verlet method:

$$r(t+h) = 2\, r(t) - r(t-h) + a(t)*h^2 \quad (7)$$
$$v(t) = (r(t+h) - r(t-h))/(2h) \quad (8)$$

prove:

set h to –h for (4) and we can get:

$$r(t-h) = r(t) - h\,[\, v(t) - (h/2)\, a(t)] \quad (9)$$

Use (4) + (9) for both sides we get (7).

Use (4) – (9) for both sides we get (8). #question proved

Velocity Verlet method is preferred because It avoids taking difference between large numbers.

8. implement verlet_next_pos() and verlet_next_vel() functions (code has been send by compass/email)

9. call verlet method in main loop (code has been sent by compass/email)

Question: when should force be updated?

When positions of current time step for all the particles has been calculated, then we can update force. Force is dependent on potential, which is solely due to relative position configuration.

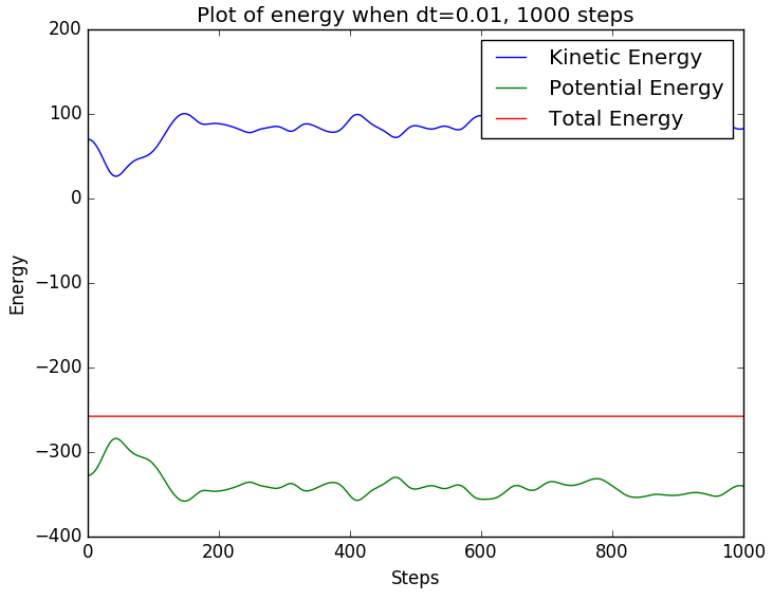
Examining the System Energy

10. finish the routine to monitor the energy (code has been sent by compass/email)

Declaration: in my calculation, 64 atoms were used. For the largest distance between 2 atoms are only sqrt(3)/2*box_length, then **I did not set the cutoff for potential energy or internal force for consistancy**. But if I want to do that, it is pretty easy just change cutoff radius to ½*box_length. I did try cutoff for potential at ½*box_length, and the TE I got is basically around -241. **Since we**

care more about the trend of variation, for consistency, I did not use cutoff for the rest of my discussion.

Results and discussion:
Time step = 0.01s, 1000 steps, seed = 1. The mean of Total Energy (TE) is about **-257.6773056** and its standard deviation is **0.003109922**, which can seen below the good stability of Total energy curve. Total energy can be treated as constant.



The configuration (position) of particle 0 of first 10 time steps are shown below (the seed used to generate random number is 9):

[[-1.58943487 -1.58699064 -1.5872625]
[-1.59175012 -1.58686221 -1.5874065]
[-1.59406246 -1.58673413 -1.58755002]
[-1.59636987 -1.58660684 -1.58769284]
[-1.59867042 -1.58648067 -1.58783475]
[-1.60096222 -1.58635599 -1.58797551]
[-1.60324346 -1.58623317 -1.58811489]
[-1.60551238 -1.58611259 -1.58825265]
[-1.60776725 -1.58599464 -1.58838854]
[-1.61000641 -1.58587971 -1.5885223]]

11. time steps comparison (1000 steps for all):

dt	0.01	0.04	0.08	0.12	0.16
seed	1	2	3	4	5
mean	-257.6773056	-257.7390748	-257.6547991	-257.5120141	-257.1081079
stddev	0.003109922	0.02768503	0.10572314	0.23219395	0.45823917
dt	0.17	0.18	0.19	0.20	0.21
seed	11	15	13	6	9

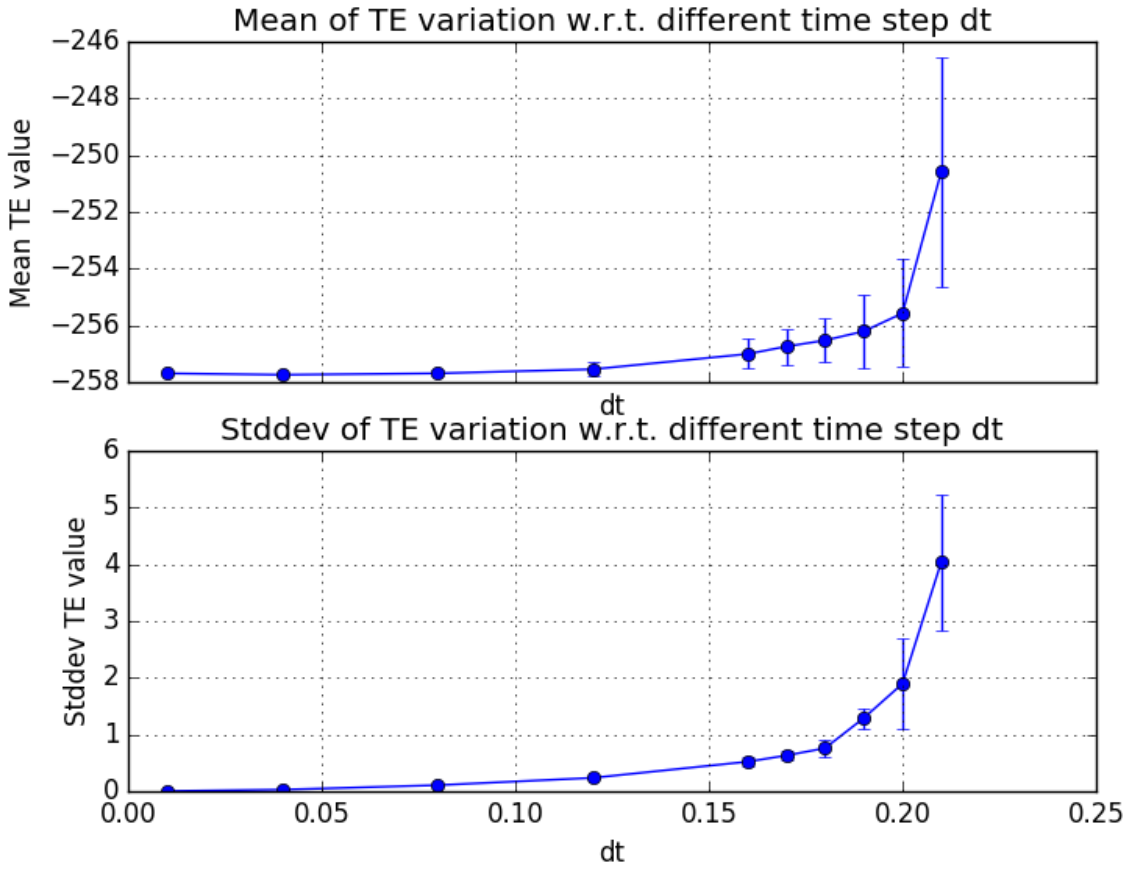
mean	-256.4068432	-256.4238266	-256.483974	-253.1829523	-245.2093645
stddev	0.76109949	0.70874196	1.43078724	2.98065495	9.42455094

The fact is that TE is sure to be around -257 (I ran code with dt =0.01 for 5000 steps and get mean = -257.6811027 and stddev of mean = 0.00282926), so 5% deviation means the standard deviation could be as large as 12.8, which is huge and impossible if we want stable and trustable TE values.

We found that if increase the time step length dt from 0.01 to 0.21, the mean of TE increased. When I set dt = 0.22, the TE will blow up near 1000 steps, let alone dt larger than 0.22.

The fact is that if we want the mean of TE to be accurate, we should not use large dt. Of course if the TE is conserved and stable then at the same time TE fluctuation would be very small (absolutely within 5%)

Look at the plot below:



Note, the errors bars are calculated by repeating one simulation several times with different initial seed. If dt goes beyond 0.18 then both mean and stddev of TE would increase greatly. If dt is larger than 0.23, then TE will blow up.

Q1: what happen if we choose large dt?

If we choose large dt, the TE will gradually increase to cause untrustable TE. Mean of TE will go higher than true value and of course the standard deviation will increase.

Q2: choose a largest dt to maintain energy conservation.

From the plot above we can see that when dt>0.18, both mean and stddev of TE will increase greatly and finally blow up. As for me, I would choose dt = 0.18 as my largest time step dt.

NOTE: all supplement code and data have been uploaded in other files through Compass 2g.