

Accelerate Molecular Dynamics Simulations

2022 Spring EC526 Project
Group: Hao Yu, Tianyi Xu, Hanyu Wang
2022/05/03

Table of contents

01

**Molecular Dynamics
Simulations**

02

Neighborhood table

03

Results

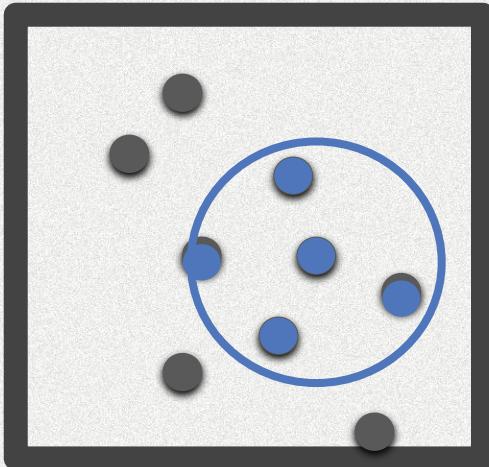
04

Animation

05

Discussion

Molecular Dynamics Simulations



Short range forces matter!

Algorithm

t=0

while t < T_final

for i=1 to N do

$\mathbf{F}_i \leftarrow 0$

 for j = 1 to N do

$\mathbf{r}_{i,j} \leftarrow \mathbf{r}_j - \mathbf{r}_i$

$\mathbf{F}_{i,j} \leftarrow \text{compute}$

 force($\mathbf{r}_{i,j} //$)

$\mathbf{F}_i \leftarrow \mathbf{F}_i + \mathbf{F}_{i,j}$

 end for

end for

$t \leftarrow t + dt$

$O(N^2)$

Molecular Dynamics Simulations

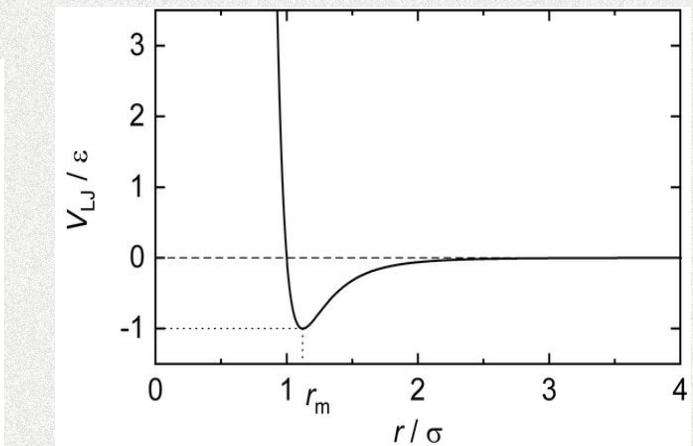
compute force – Verlet method

For the motion of a particle:

$$\begin{aligned}\mathbf{v}(t + \Delta t) &= \mathbf{v}(t) + \mathbf{a}(t)\Delta t + O((\Delta t)^2) \\ \mathbf{r}(t + \Delta t) &= \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2}\mathbf{a}(t)(\Delta t)^2 + O((\Delta t)^3)\end{aligned}$$

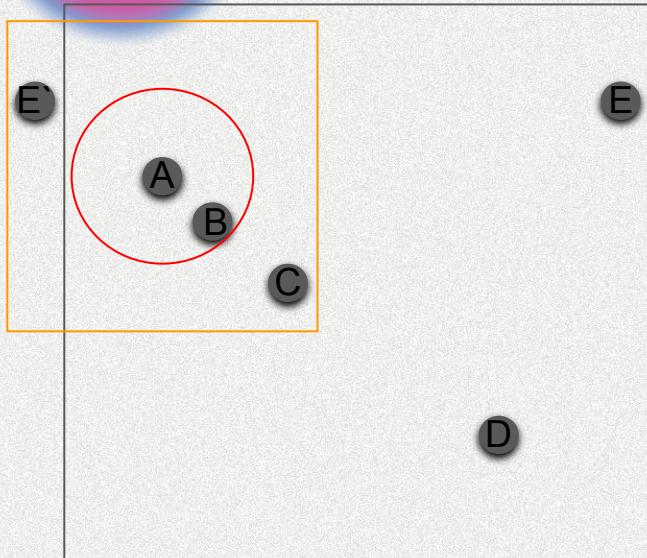
Expressing the speed at mid-interval $v(t + \Delta t/2)$:

$$\left\{ \begin{array}{l} \mathbf{v}(t + \Delta t/2) = \mathbf{v}(t) + \mathbf{a}(t)\Delta t/2 \\ \mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t + \Delta t/2)\Delta t \\ \mathbf{a}(t + \Delta t) = \mathbf{F}(\mathbf{r}(t + \Delta t))/m \\ \mathbf{v}(t + \Delta t) = \mathbf{v}(t + \Delta t/2) + \mathbf{a}(t + \Delta t)\Delta t/2 \end{array} \right.$$



$$\text{Lennard-Jones Potential } V_{LJ}(r_{ij}) = 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right]$$

Neighborhood Table



Array Version

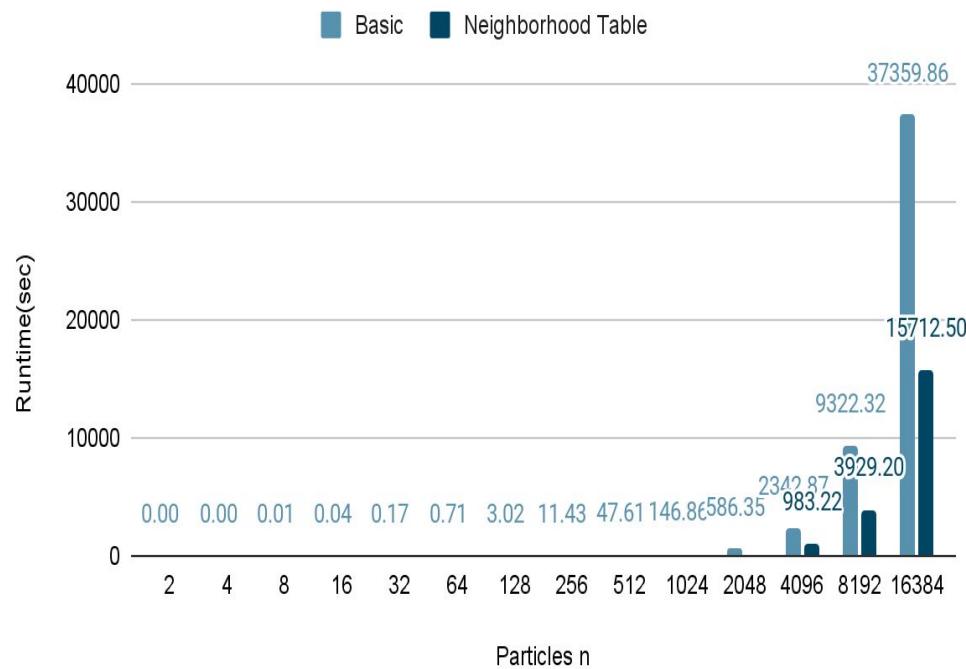
A	0		B	0
B	3		C	1
C	5		E	2
D	D		A	3
E	E		C	4
			B	5
			A	6
			A	7
			-1	8

Vector Version

A	B	C	E
B	A	C	
C	A	B	
D			
E	A		

Result - Brute force and Neighborhood Table

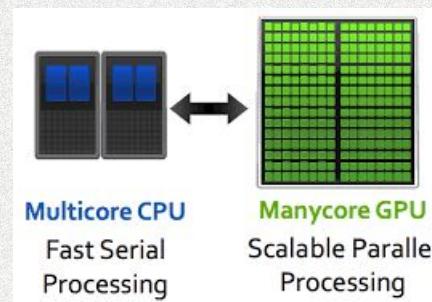
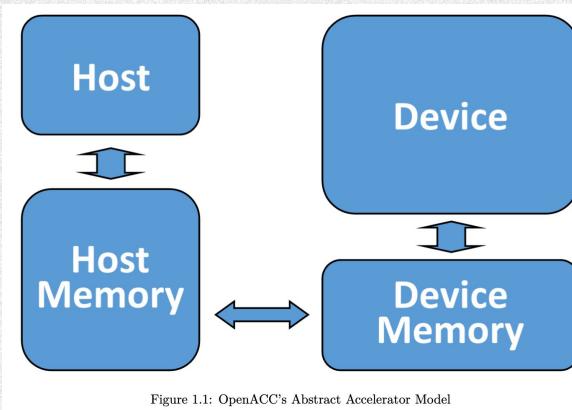
Basic vs Neighborhood Table MD



N particles in 4096*4096 2D box
dt=0.005
5000 time steps
m=1

Neighborhood Table save
~60% computation time

Result - Open ACC

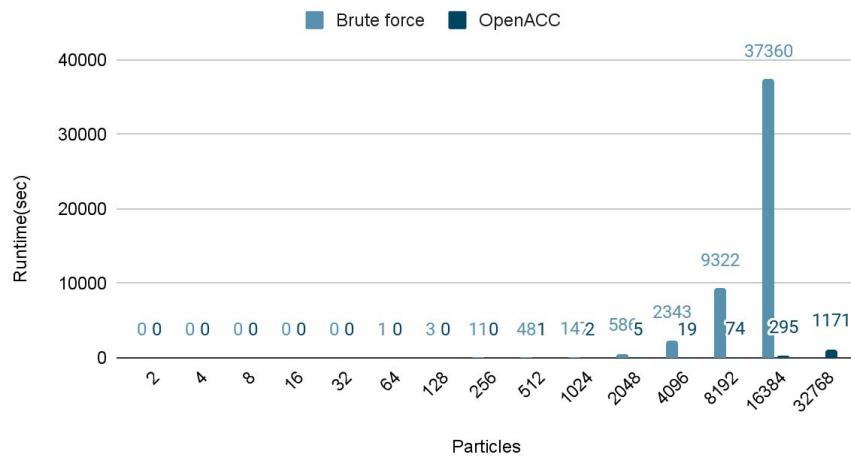


Result - Open ACC

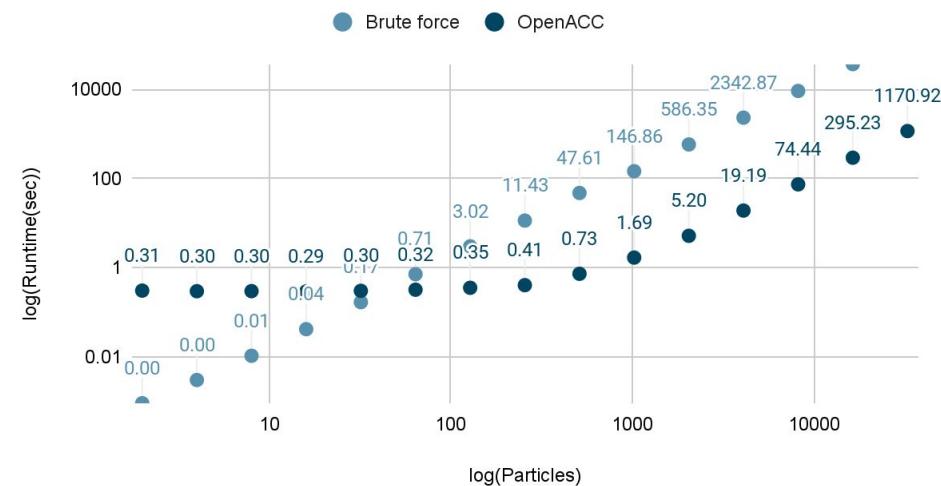
```
sign(double, double):
    205, Generating implicit acc routine seq
        Generating acc routine seq
        Generating Tesla code
calcVerlet(int, int, double, double, double, double, double *, double *, double *):
    336, Generating copy(pos[:nd*np],vel[:nd*np],acc[:nd*np])
        Generating Tesla code
    336, #pragma acc loop gang /* blockIdx.x */
    339, #pragma acc loop vector(128) /* threadIdx.x */
        Generating reduction(+:Ekin,Epot)
    343, #pragma acc loop seq
339, Loop is parallelizable
```

Result - Open ACC

Brute force and OpenACC

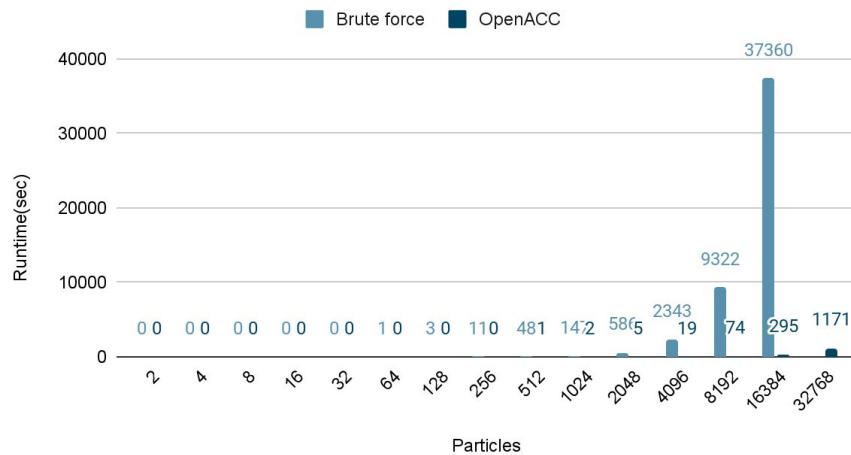


Brute force paralleled by OpenACC

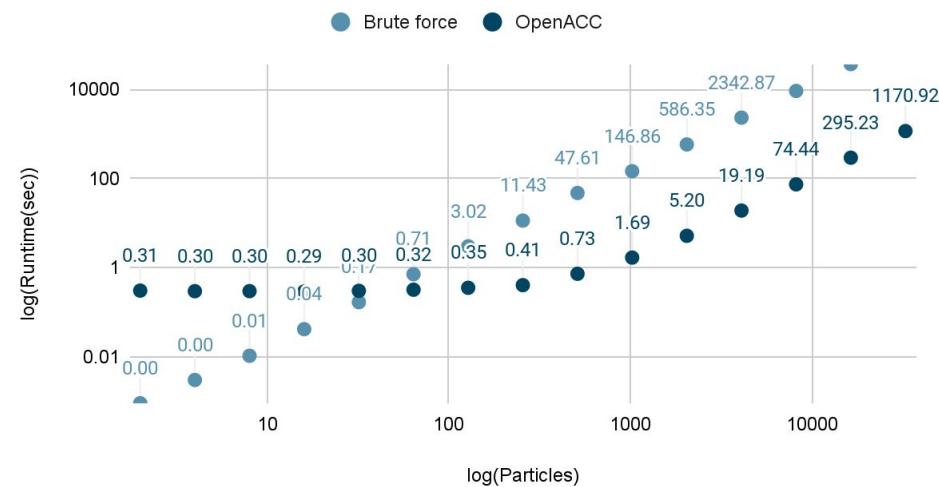


Result - OpenMP

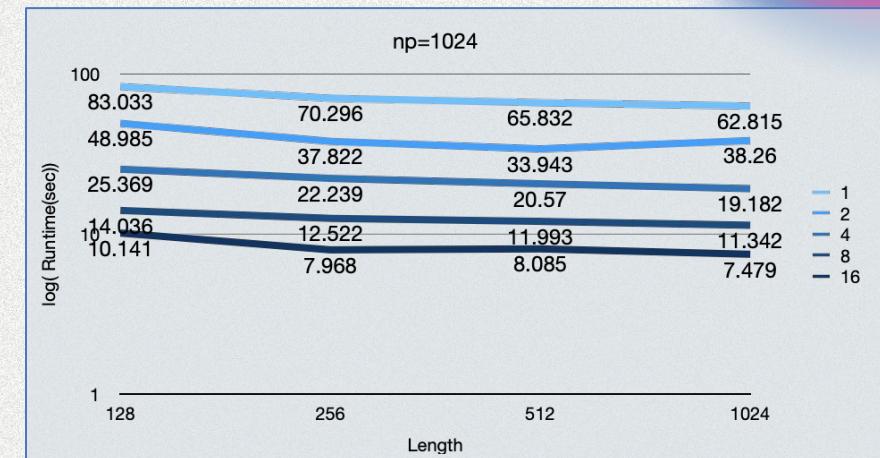
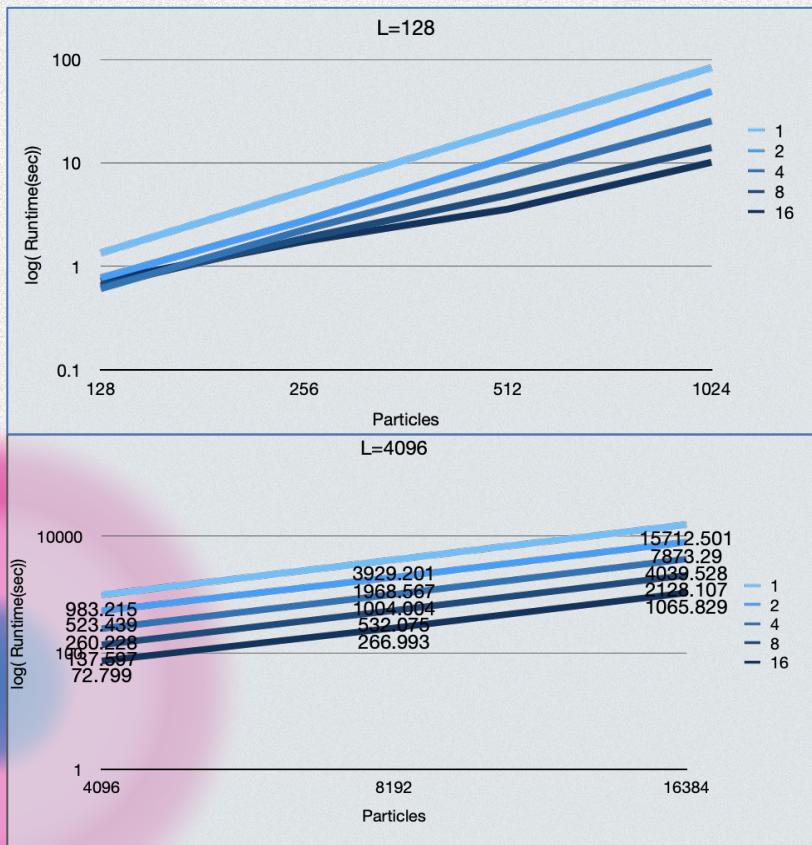
Brute force and OpenACC



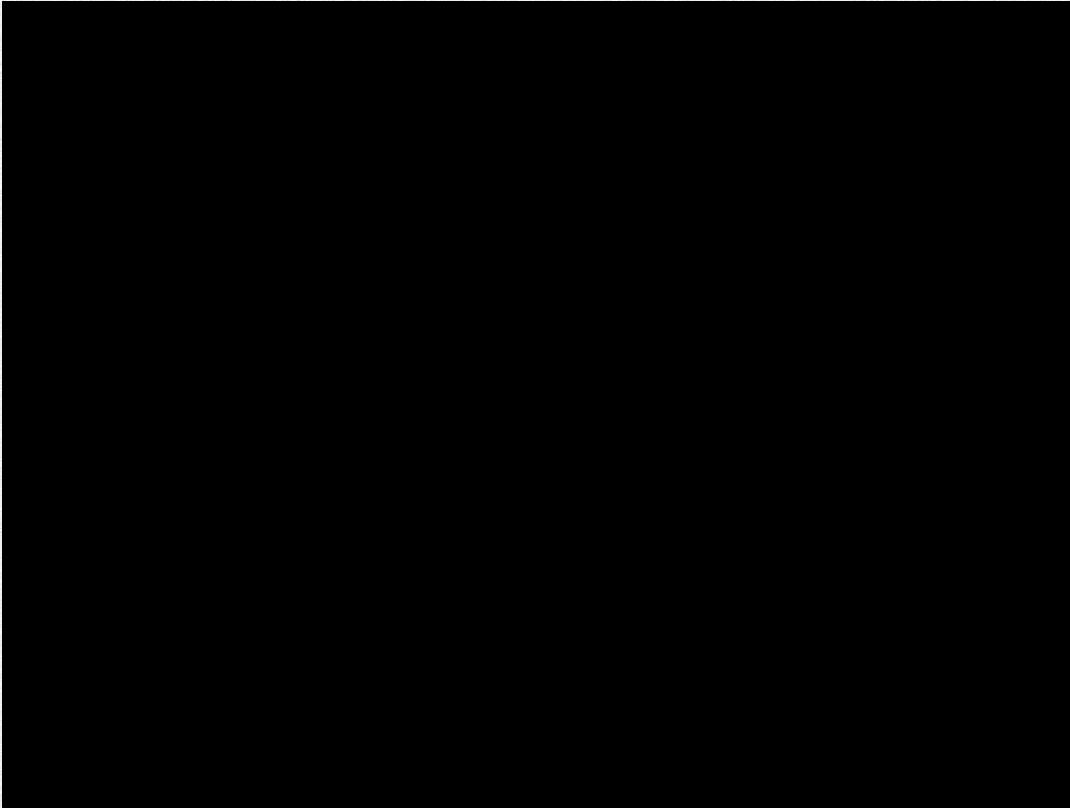
Brute force paralleled by OpenACC



Result - OpenMP



Animation



Discussion

What is this error mean?

```
[[imhaoyu@scc-c13 openacc]$ ./openACC 10
Failing in Thread:1
call to cuCtxCreate returned error 101: Invalid device
```

**Thank
You**