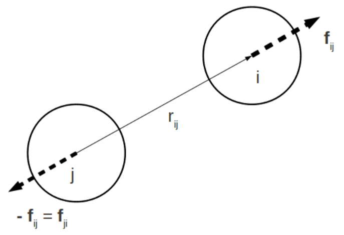
"Tricks" of the Trade

- *Efficiency* is the word
- But the straightforward implementation of algorithms to solve eq. of motion may not be the most efficient way
- There are ways of skipping some calculations and still get the "right" answer
- Force calculations are expensive: can we avoid parts of it, square roots? (show bench)

Force calculation shortcuts



- Newton's 3rd: once calculate F_{ij}, have F_{ji}
- If $V=V(r_{ij})$ is an even power, then because

$$\mathbf{F}_{ij} = -\frac{1}{r_{ij}}\left(\frac{dv(r_{ij})}{dr_{ij}}\right)\mathbf{r}_{ij}$$

only need r_{ij}^2 for F, thus no sqrt()

Potential Interpolation

One way to not calculate a complicated potential at every step is to use interpolations:

$$\text{e.g.}\, V(r_{ij}) = A\exp(-C\cdot r_{ij}) - \frac{B}{r_{ij}^6}$$

Barker, et.al. 1971
For Argon

- Use tables to evaluate the potential explicitly for only a small number of distance values
- For arbitrary distances, interpolate the potential from bracketing values in the table

Potential Interpolation

Newton-Gregory forward difference method:

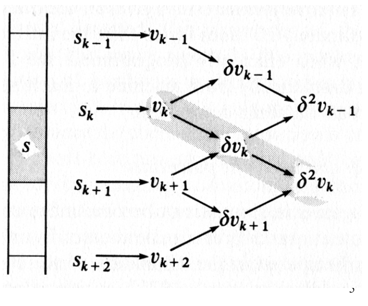
- Use $s=r_{ij}^2$ and calculate $V_1 = V(s_1)$, $V_2 = V(s_2)$, ... at equal intervals δs.
- Define 1st and 2nd differences:
  - $\delta V_k = V_{k+1} - V_k$
  - $\delta^2 V_k = \delta V_{k+1} - \delta V_k$
- For a value *s* between s_k and s_{k+1}, *interpolate*

Potential Interpolation

$$V(s) \approx V_k + \xi\delta V_k + \tfrac{1}{2}\xi(\xi-1)\delta^2 V_k$$

- where

$$\xi = (s - s_k)/\delta s$$



Potential Interpolation

For the force, using

$$\mathbf{f}_{ij} = -\frac{1}{r_{ij}}\left(\frac{dV(r_{ij})}{dr_{ij}}\right)\mathbf{r}_{ij} = -\frac{w(r_{ij})}{r_{ij}^2}\mathbf{r}_{ij}$$

and noting that

$$\frac{w(r_{ij}^2)}{r_{ij}^2} = \frac{w(s)}{s} = 2\frac{dV}{ds}$$

Thus, for **f** can just differentiate

$$V(s) \approx V_k + \xi\delta V_k + \tfrac{1}{2}\xi(\xi-1)\delta^2 V_k$$

Potential Interpolation

- An alternate method. For each interval (s_k, s_{k+1}) represent potential by 5th order polynomial:

$$V(s) \approx c_0 + c_1 \delta s + c_2 \delta s^2 + c_3 \delta s^3 + c_4 \delta s^4 + c_5 \delta s^5$$

- Where $\delta s = s - s_k$.
- c_i coeffs are determined by the exact values of V(s), dV(s)/ds, and $d^2V(s)/ds^2$ evaluated at the two ends of the interval.
- Advantage: the s_k need not be evenly spaced.

7

Not only by algorithms...

Compiler optimizations can help:

- -O3 (illustrate)

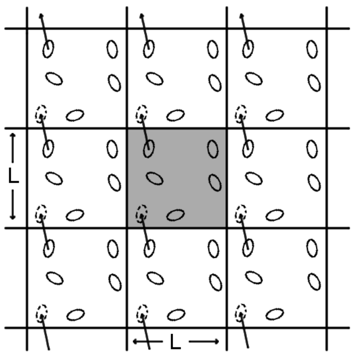
Hardware compiler type and version:

- commercial

8

Problems with PBC: time to pay...

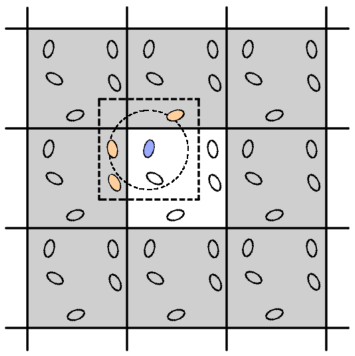
- PBC introduces the problem that <u>all</u> image pair interactions should be considered (infinite).



9

PBC: still hope...

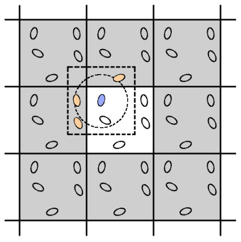
- For short-ranged interactions could possibly skip far-away interactions.
- Approximate to only consider closest real or image particles: *<u>minimum image convention.</u>*
- Pairwise interactions only require N(N-1)/2 terms.



10

Avoid N^2 calculation with r_c

- Minimum Image convention still $O(N^2)$
- What if for short-ranged interactions could possibly skip far-away interactions – do less!
- WARNING: This introduces an error and discontinuity in force and potential at cutoff r_c
- Thus, energy will not be conserved for truncated interactions

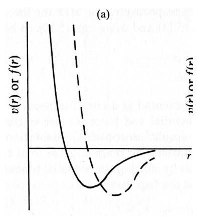


11

Avoid the N^2 calculation

<u>Solution</u>:

- Truncate interaction at r_c
- Shift potential by an amount $V_c = V(r_c)$



$$V^T(r_{ij}) = \begin{cases} V(r_{ij}) - V_c & r_{ij} \le r_c \\ 0 & r_{ij} > r_c \end{cases}$$

Corresponds to (a)

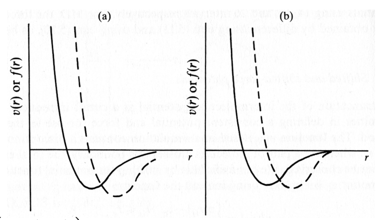
12

Avoid the N^2 calculation

- Additional term V_c is constant and does not affect force calculation nor eq. of motion
- However, contribution to total energy varies from step to step. Have to account for it in the energy
- Force is still discontinuous at $r=r_c$
- Solution: add yet another term

Avoid the N^2 calculation

- Additional term is linear such that derivative is zero at cutoff distance
- This is the 'shifted-force potential'



$$V^T(r_{ij}) = \begin{cases} V(r_{ij}) - V_c - \left(\frac{dV(r_{ij})}{dr_{ij}}\right)_{r_{ij}=r_c}(r_{ij}-r_c) & r_{ij} \le r_c \\ 0 & r_{ij} > r_c \end{cases}$$

Corresponds to (b)

Avoid the N^2 calculation

<u>Caveats:</u>

- Discontinuity now shifts to the gradient of the force.
- The 'shifted-force potential' does not correspond anymore to desired model potential.
- However, thermodynamics are still very similar to original problem.

<u>Alternate Route:</u>

- Introduce a “switching” function to smoothly taper potential to zero at large *r*.

In sum: on reducing the distance “checking”

<u>Can do</u>:

- Reduced use of expensive functions
- Could use potential interpolations
- Get good compiler and use optimization flags
- PBC with minimum image convention
- Add distance cutoff r_c and correct energy

<u>BUT</u>:

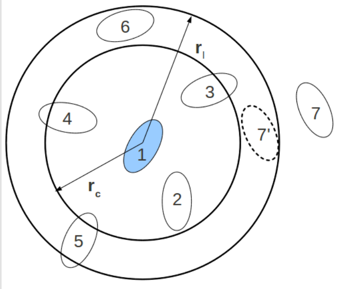
- Still have to check (but not compute) all pairs - $O(N^2)$

Neighbor Lists

<u>Solution</u>:

- Again rely on short-range interactions and keep a list of only neighbors
- Only update this list occasionally
- Between updates, calculate interactions with all neighbors in the list
- Can do:
  - Verlet neighbor list
  - Cell structures

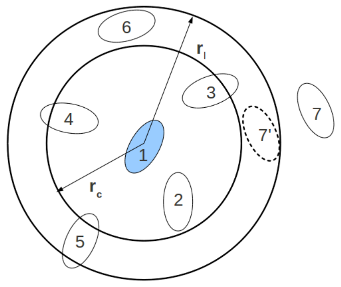
Verlet Neighbor List



- Surround r_c by a “skin” of r_l
- Loop over all particles inside r_l (Note, more interactions than with pure r_c)
- But number of pairs ~O(N)

Verlet Neighbor List

- Once in a while, update list (depends on size of r_l)
- r_l should be big enough so as to prevent particle 7 into penetrating within r_c in between updates
- 10-20 updates typical, proportional to r_l
- Can do automatic updates



Verlet Neighbor List

Caveats:

- As the size of the system increases, the total size of the neighbor lists also increases thus affecting storage.
- Update (of the lists) is still N^2

Show bench

- Next: Use alternative method of cell structures.