

# *Neighbor Tables and the Cell Method*

Kenneth P. Esler Jr.

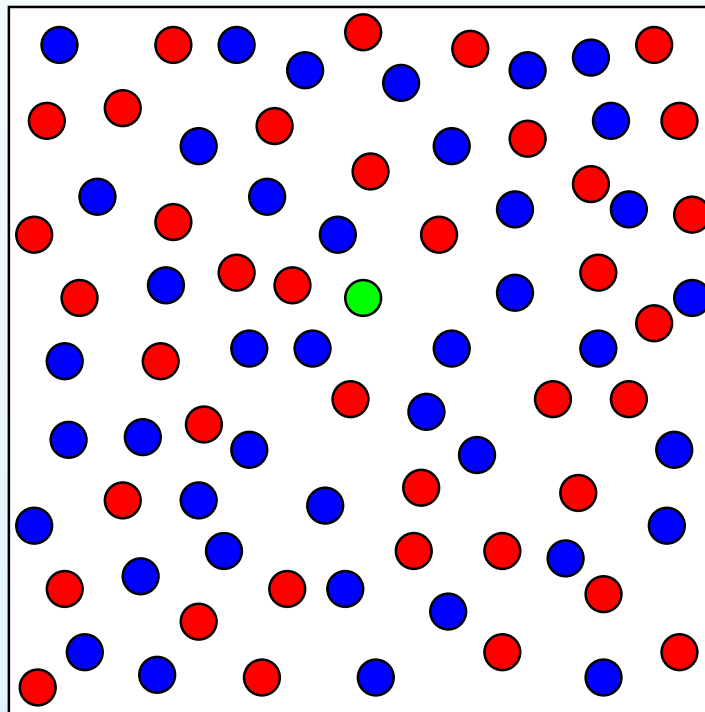
UIUC Department of Physics

## Pair Potentials

For a short range potential, the number of potential calculations is naively  $N(N - 1)/2 \approx \mathcal{O}(N^2)$ . But, if the potential is short range, i.e.

$$v(r) = 0 \text{ for } r > r_c,$$

then we need only consider the particles within  $r_c$  of each other.

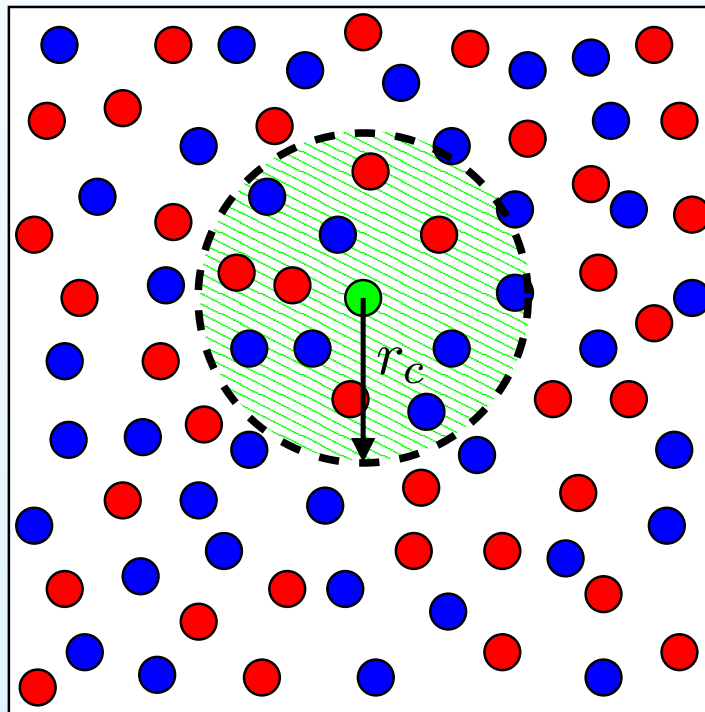


## Pair Potentials

For a short range potential, the number of potential calculations is naively  $N(N - 1)/2 \approx \mathcal{O}(N^2)$ . But, if the potential is short range, i.e.

$$v(r) = 0 \text{ for } r > r_c,$$

then we need only consider the particles within  $r_c$  of each other.

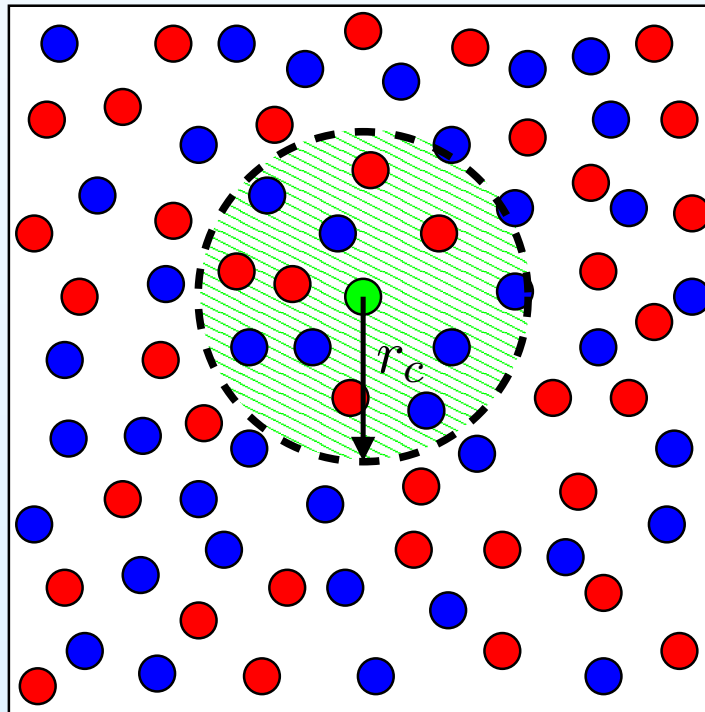


# Neighbor Tables

		Particles						
		1	2	3	4	5	6	...
Neighbors	6	6	45	9	15	19	1	...
	34	34	57	16	43	31	3	...
	17	17	16	25	36	14	19	...
	29	29	10	61	13	59	40	...
	12	12	8	58	7	63	21	...
	18	18		12	27	26	64	...
	47	47		30	38		39	...
	19	19		21			23	...
				51			14	...

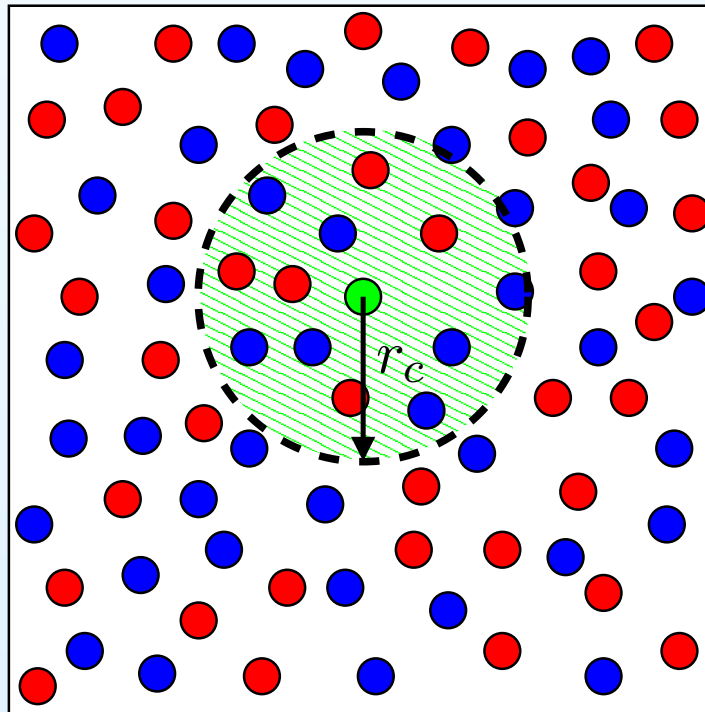
## Scaling

- Number of potential computations is  $mN$ , where  $m$  is the average number of neighbors inside  $r_c$ .



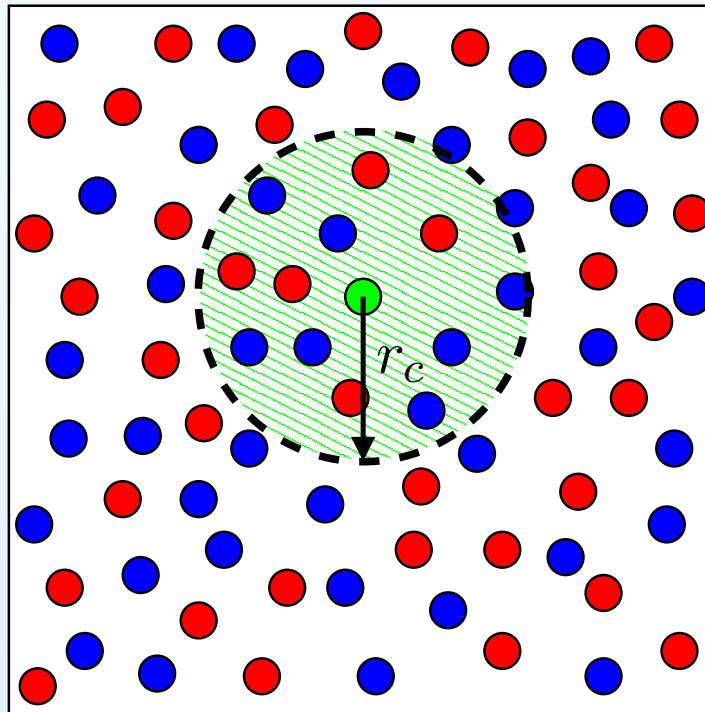
## Scaling

- Number of potential computations is  $mN$ , where  $m$  is the average number of neighbors inside  $r_c$ .
- $m$  is independent of the system size!



## Scaling

- Number of potential computations is  $mN$ , where  $m$  is the average number of neighbors inside  $r_c$ .
- $m$  is independent of the system size!
- Hence, once we have the neighbor table, computation is  $\mathcal{O}(N)$ !



## Constructing Neighbor Tables

---

- To construct a table, we must check to see if each particle is inside each other's radius.

## Constructing Neighbor Tables

---

- To construct a table, we must check to see if each particle is inside each other's radius.
- Hence, construction of table is  $\mathcal{O}(N^2)$ .

## Constructing Neighbor Tables

---

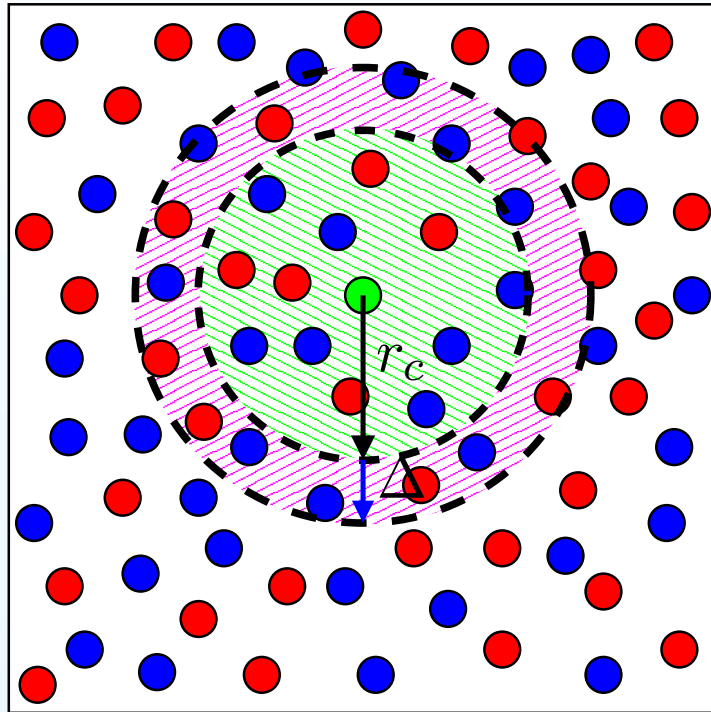
- To construct a table, we must check to see if each particle is inside each other's radius.
- Hence, construction of table is  $\mathcal{O}(N^2)$ .
- Is this useless, then?

## Constructing Neighbor Tables

---

- To construct a table, we must check to see if each particle is inside each other's radius.
- Hence, construction of table is  $\mathcal{O}(N^2)$ .
- Is this useless, then?
- No! If we are clever, we don't have to reconstruct the table very often.

## Constructing Neighbor Tables: Skin Depth

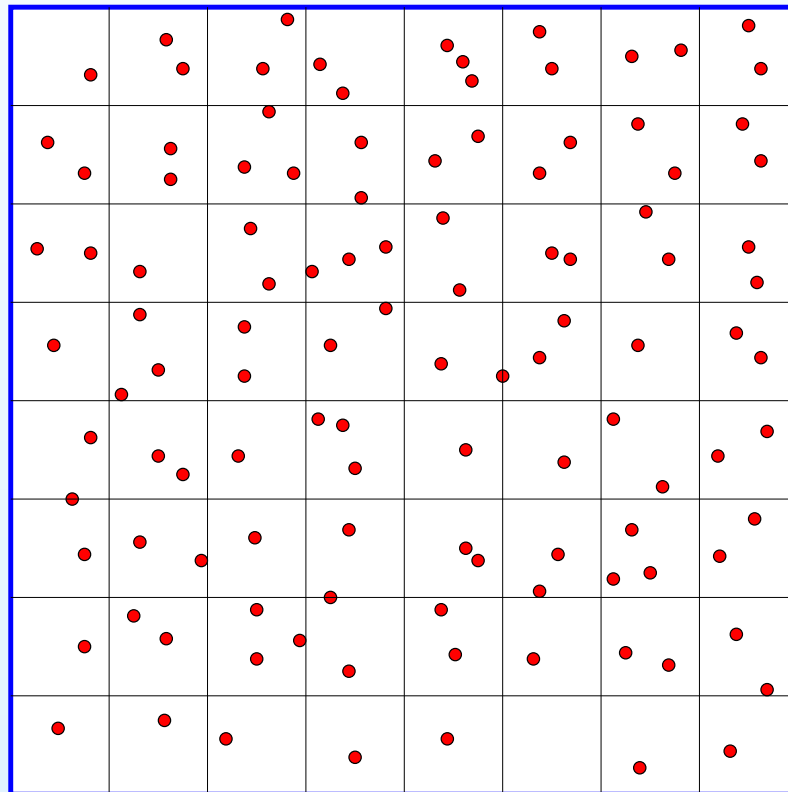


- Instead of cutting off our table at  $r_c$ , we cut it off at  $r_c + \Delta$ .
- $\Delta$  is a *skin depth*, which allows particles to move for a while before we need to reconstruct table.
- Keep track of two largest distances traveled,  $d_1$  and  $d_2$ . When  $d_1 + d_2 \geq \Delta$ , need to recompute table.

- As  $\Delta$  increases, computing forces takes longer since each particle will have more neighbors
- As  $\Delta$  increases, we have to reconstruct neighbor table less frequently.
- Choose  $\Delta$  to optimize efficiency.

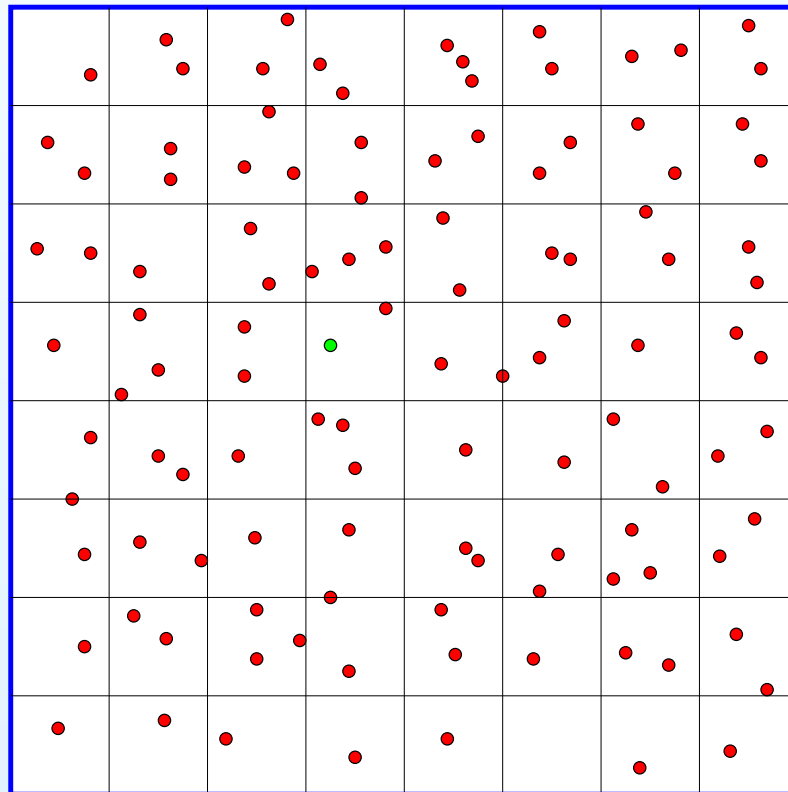
## Further Improvements: The Cell Method

- When computing the neighbor table, divide box into cells with size  $L > r_c + \Delta$ .



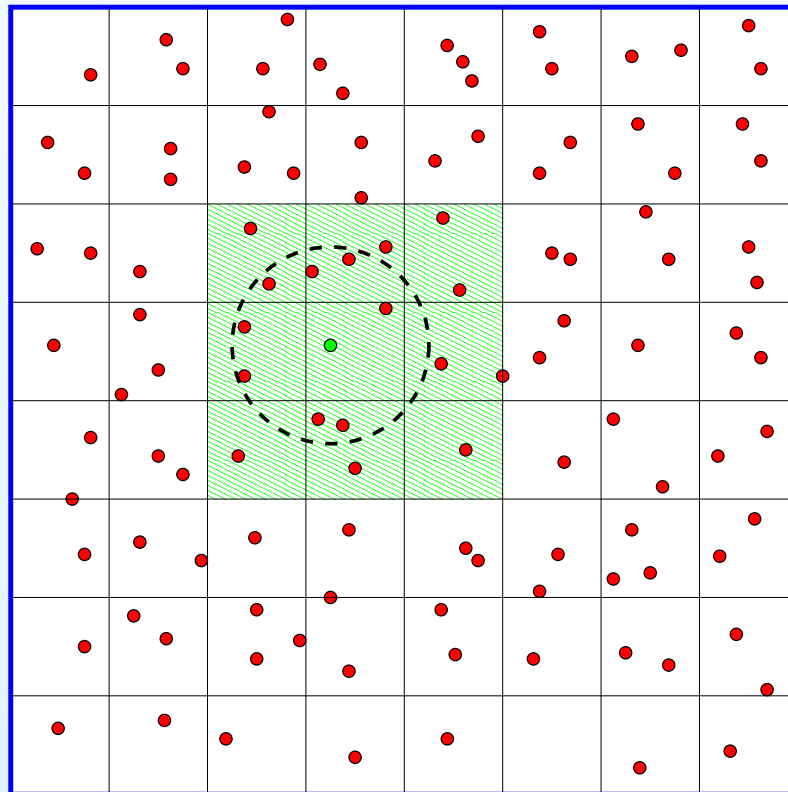
## Further Improvements: The Cell Method

- When computing the neighbor table, divide box into cells with size  $L > r_c + \Delta$ .
- Need only consider a ptcl's own and neighboring cells.



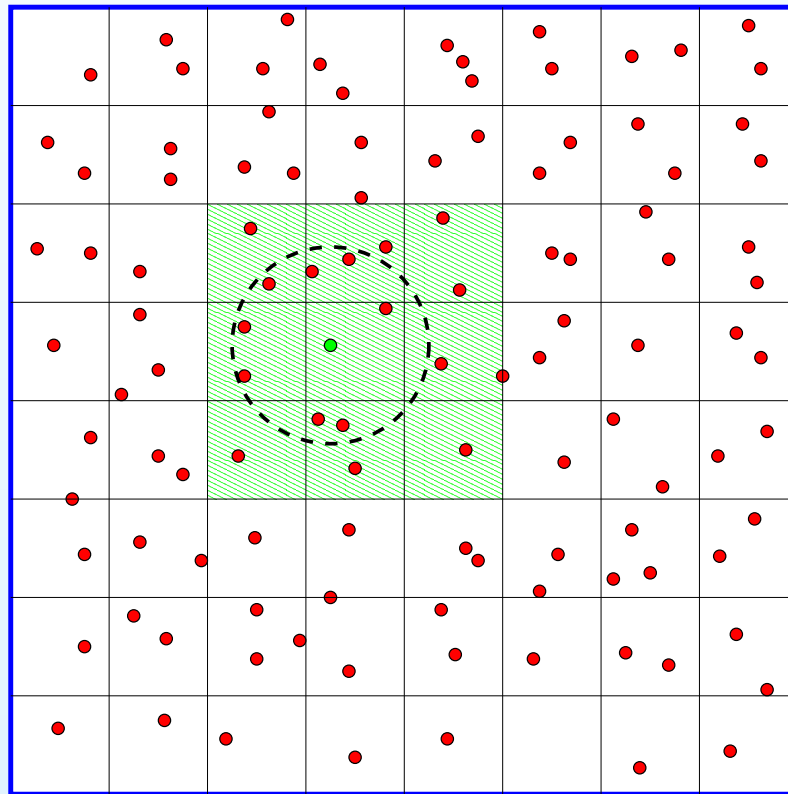
## Further Improvements: The Cell Method

- When computing the neighbor table, divide box into cells with size  $L > r_c + \Delta$ .
- Need only consider a ptcl's own and neighboring cells.



## Further Improvements: The Cell Method

- When computing the neighbor table, divide box into cells with size  $L > r_c + \Delta$ .
- Need only consider a ptcl's own and neighboring cells.
- Hence, construction of neighbor table is  $\mathcal{O}(N)$ .



Thursday

**No class:**

**Meet with your teams!**

**Homework 3 is up**