Neighbor Tables
Long-Range Potentials

Today we learn how we can handle long range potentials.

- Neighbor tables
- Long-range potential
- Ewald sums

Periodic distances

- Minimum Image Convention: take the closest distance

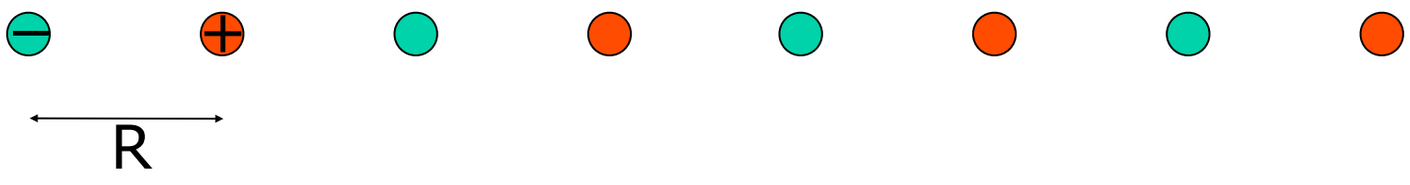
$$|r|_M = \min (r+nL)$$

Potential is cutoff so that V(r)=0 for r > L/2 since force needs to be continuous. Remember perturbation theory.

- Image potential

$$V_I = \Sigma_n \, v(r_i - r_j + nL) \text{ - background (if needed)}$$

For long-range potential (e.g. Coulomb) need the Ewald image potential. You need a background and convergence method.



Perturbation theory

- One can restore a cutoff potential by using a tail correction

$$V_{\text{tail}} = \frac{\rho}{2} \int dr\; g(r)\Delta\phi(r)$$

- To do better one has to find effect of perturbation on g(r). E.g. one can use the RHNC (reference-hypernetted-chain) equation: integral equation involving the potential and g(r).

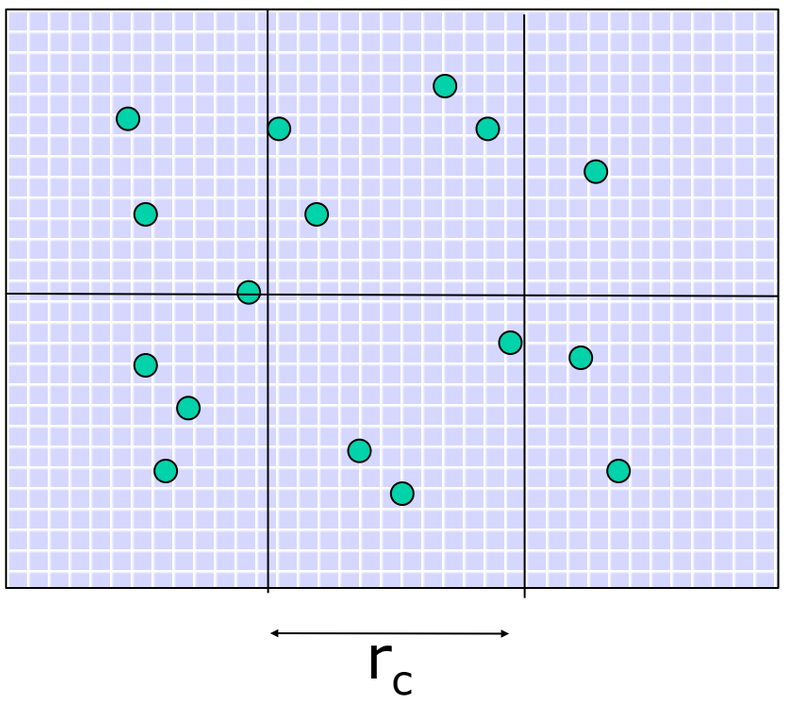
- The Stillinger-Lovett condition says that S(k) at low k for a charged system is different that in a neutral system.

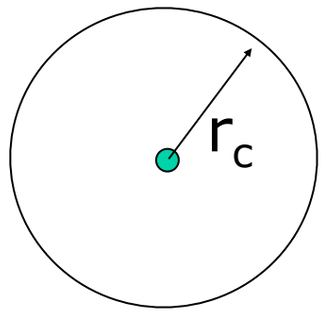
$S(k)=c\,k^2$ for charged system

$S(k)=c$ for uncharged system

Complexity of Force Calculations

- Complexity is the scaling with the number of degrees particles.
- Number of terms in pair potential is $N(N-1)/2 \approx O(N^2)$
- For short-range potential you can use neighbor tables to reduce it to O(N)
  - (Verlet) neighbor list for systems that move slowly.
  - bin sort list (map system onto a $r_c \times r_c$ mesh and find neighbors from the mesh table, only n.n. cells matter).
  - Or both

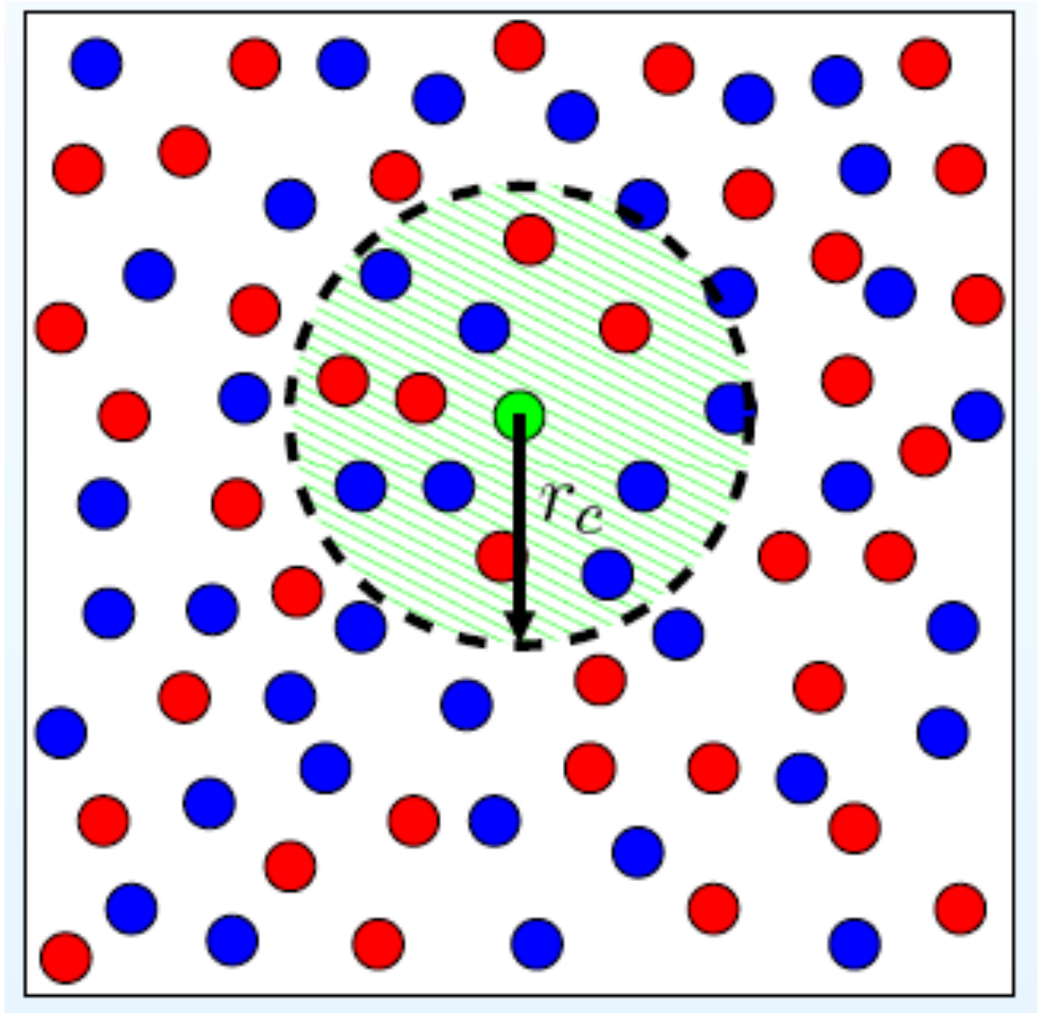




If $v(r > r_c) = 0$, force calculations are O(N) with M neighbors

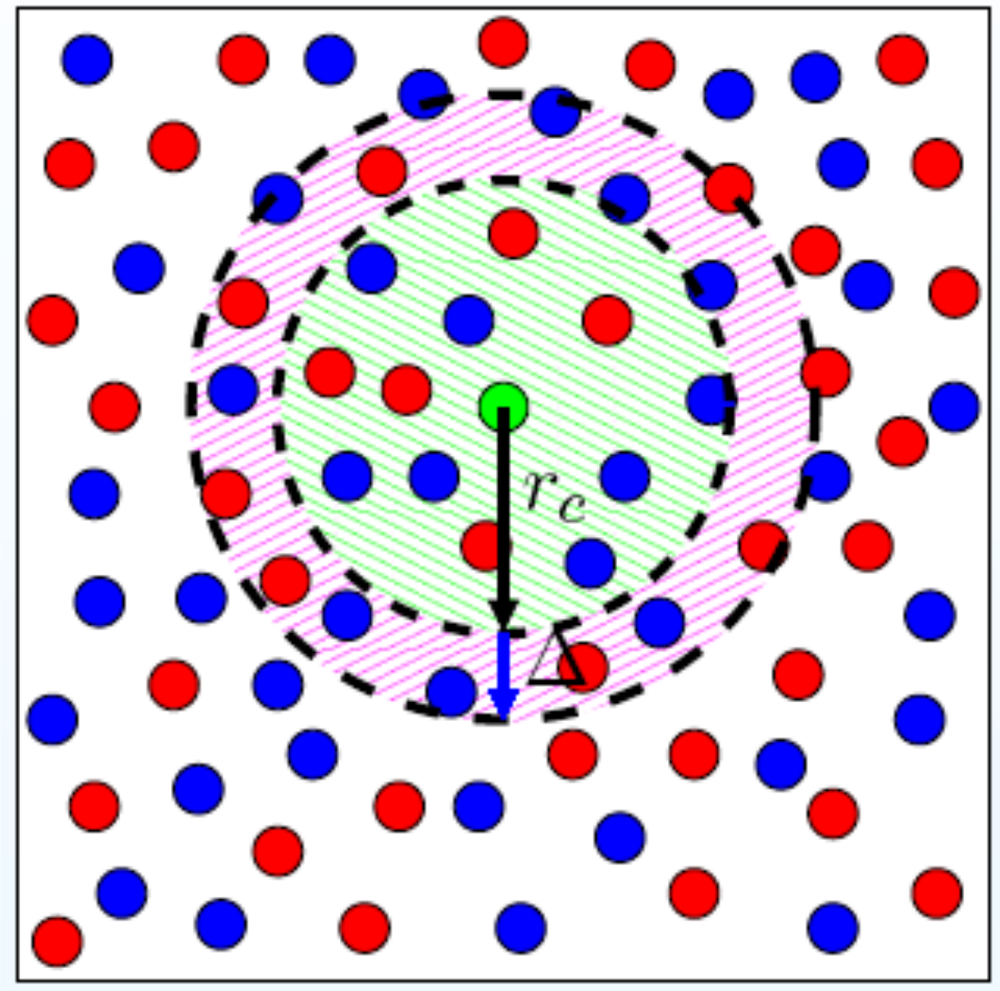
Neighbor Lists: $O(N^2)$ to $O(N)$

- Pair Potentials: If $V_c(r) = 0$ for $r > r_c$ (i.e., short-ranged), then
  - Number of computations of the potential is mN, where m is average number of neighbors inside r_c and is independent of system size!
  - So, once we have the neighbor table, the calculation is $O(N)$!
- Constructing Neighbor Tables:
- $O(N^2)$, must check if particles fall inside each other's radius.
- only have to reconstruct the table occasionally.



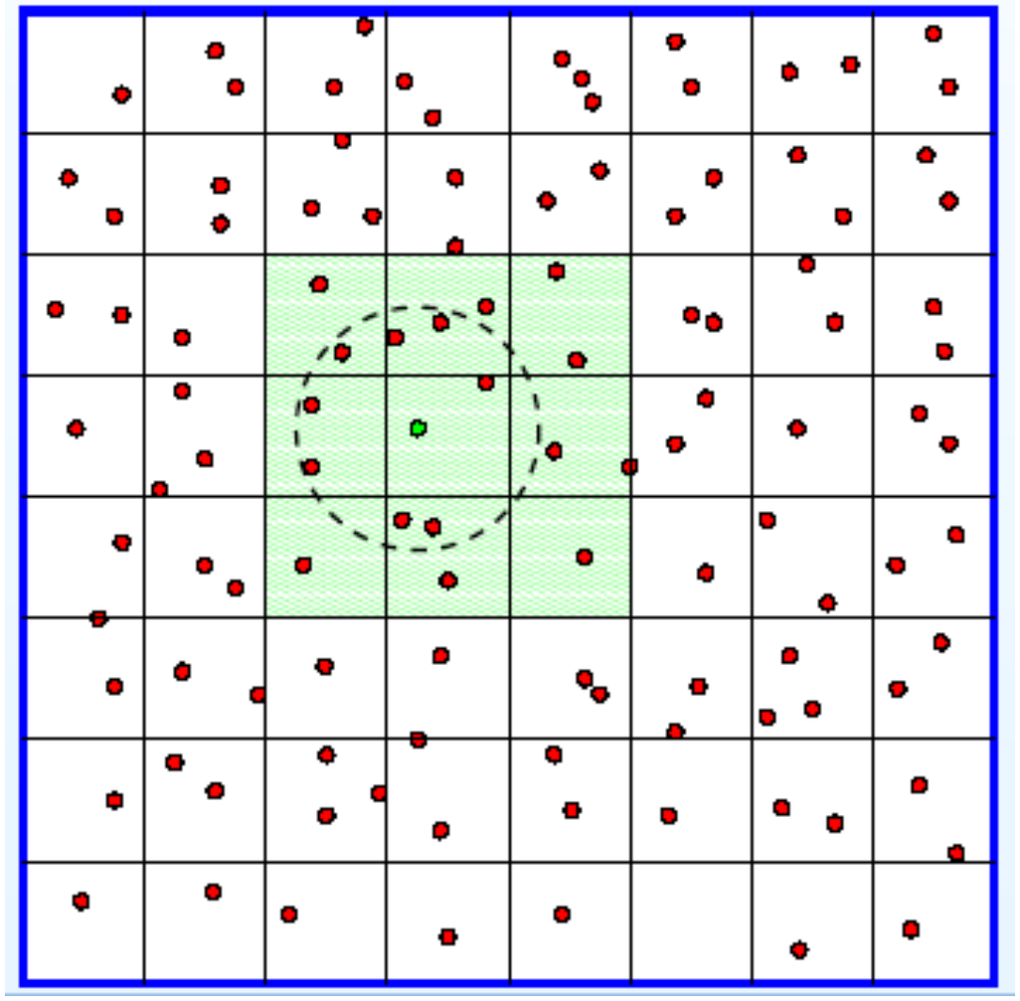
Maintaining neighbor Lists: use skin depth

- Cut off table at $r_c + \Delta$ (skin depth).
  - allows a particle to move a while before new table needed.
- Keep track of the two largest distances traveled, d_1 and d_2,
- when $d_1+d_2 \geq \Delta$ get new table.
- Choose Δ to optimize efficiency
- As Δ decreases, fewer neighbors and force evaluations.
- As Δ increases, need to calculate neighbor table less often. $O(N^2)$ operation.
- Dynamically optimize Δ by fitting $t(\Delta)$ to a polynomial and minimizing.



Improvements: the Cell Method

- divide box into cells of size $L > r_c + \Delta$.
  - Need only particle's own and neighboring cells.
- table construction is $O(N)$!
- To find neighbors use a linked list
- Memory is also N (not mN)



Charged systems

How can we handle charged systems?

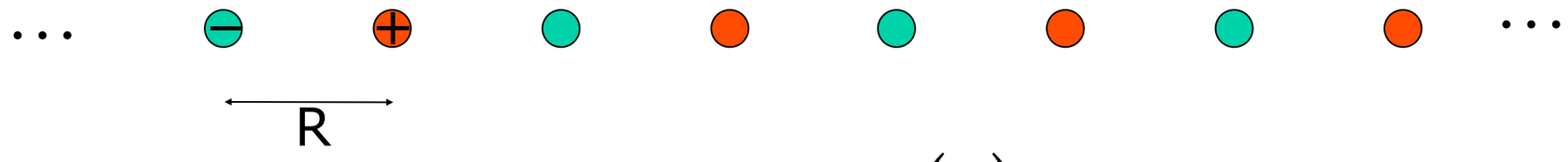
- Treat like short-ranged potential: cutoff potential at r>L/2.
  Problems:
  - Effect of discontinuity never disappears ((1/r) (r^2) gets bigger.
  - Will violate Stillinger-Lovett conditions because Poisson equation is not satisfied.
  - Even a problem with dipolar forces.
- Image potential solves this:

$$V_I = \Sigma_n v(r_i - r_j + nL)$$

  - But summation diverges!

    We need to resum: use the Ewald image potential.
  - For one component system, need to add a background to make it neutral.

1-D Madelung Sum: Prelude to Ewald Sum



Madelung constant: $$\frac{\alpha}{R} = \sum_{j \neq i} \frac{(\pm)}{R\, p_{ij}} \qquad r_{ij} = R\, p_{ij}$$

The value of α will depend on whether it is define in terms of the lattice constant R. Start on negative ion, summing (left and right)

$$\alpha = \sum_{j \neq i} \frac{(\pm)}{p_{ij}} = 2\left[\frac{1}{1} - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots\right]$$ Sum is conditionally convergent

$$= 2 \ln 2$$

Note: $$\ln(1+x) = \left[x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots\right]$$

$V_{\text{electrostatic}} \sim \alpha / R$

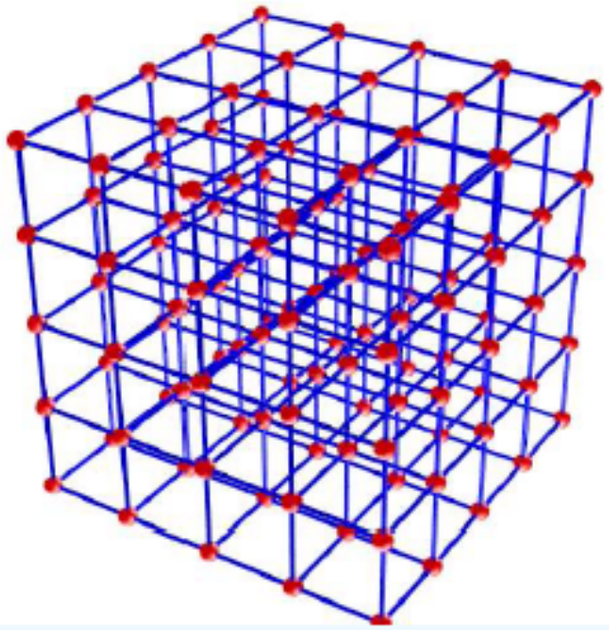
Structure	α
NaCl	1.747565
CsCl	1.762675
ZnS	1.6381

In 3D the series presents greater difficulty. Series will not converge unless successive terms in the series are arranged so that + and − terms nearly cancel.

Powerful methods were developed by Ewald (Ann. Physik **64**, 253 (1921), and Evjen (Phys. Rev. **39**, 675 (1932) and Frank (Phil. Mag. **41**, 1287 (1950) .

Long-Ranged Potentials

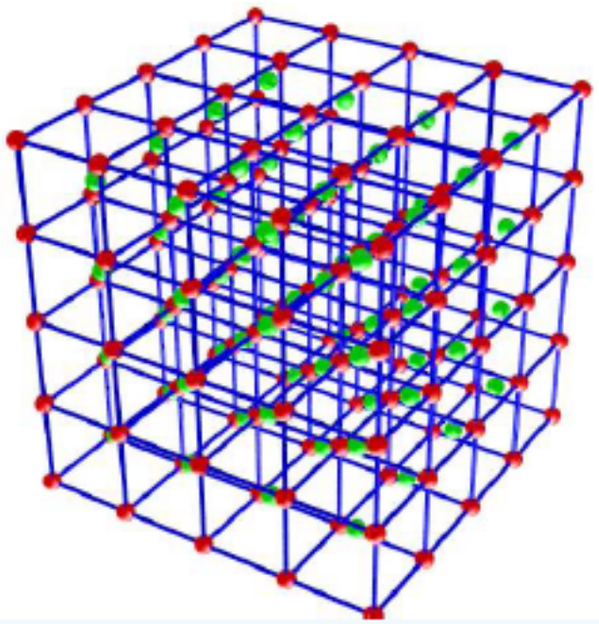
- Why make the potential long ranged?
  - Consider a cubic lattice with +1 charges, and its Coulomb potential.



$$V(r_i) = \sum_{L \neq 0} \frac{1}{|r_i - L|} \approx \int_0^\infty dr\, 4\pi r^2 \frac{\rho}{r}$$

- Approximate integral diverges!
  - Correct! Non-neutral system with infinite charge has infinite potential.

  - Consider a cubic lattice with charge neutrality, i.e. with ±1 charges.



$$V_{\text{cell}} = \frac{1}{2} \sum_{j \neq i} \sum_{L} \frac{Z_i Z_j}{\left|r_i - r_j - L\right|}$$

- Again need convergent lattice sum.
  - Energy is finite in charge neutral cell

What is Long-Ranged Potential?

- A potential is long-ranged if the real-space lattice sum does not (naively) converge.
  - In 3D, a potential is long-ranges if converges at rate < r^{-3}.
  - In 2D, a potential is long-ranges if converges at rate < r^{-2}.
  - In practice, we often use techniques for potentials that are not strictly long-ranged.

- MOTIVATION for bothersome math: Most interesting systems contain charge:
  - Any atomic system at the level of electrons and nuclei.
  - Any system with charged defects (e.g., Frenkel defects)
  - Any system with dissolved ions (e.g. biological cases)
  - Any system with partial charges (e.g. chemical systems)

Use Fourier Transform: large r=small k

If $f(r)$ is a continuous periodic function such that $f(r+L)=f(r)$, with $L_n = n_x L_x x + n_y L_y y + n_z L_z z$, then

$$f(r) = \sum_k e^{ik\cdot r} f_k \; ; \quad k = m_x \frac{2\pi}{L_x}\hat{x} + m_y \frac{2\pi}{L_y}\hat{y} + m_z \frac{2\pi}{L_z}\hat{z}$$

$$f_k = \frac{1}{\Omega}\int_{\text{all space}} dr\, e^{-ik\cdot r} f(r)$$

k's are discrete!

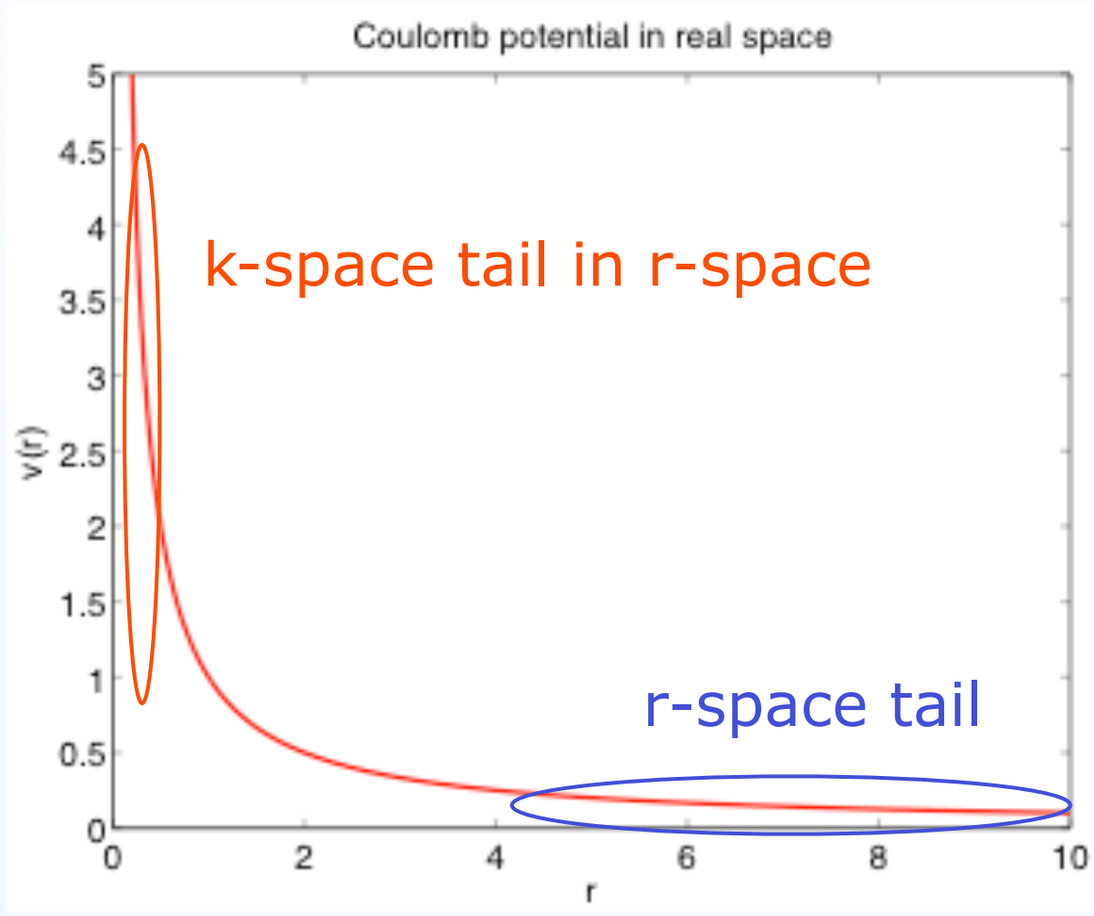
With $f(r) \sim v(r) = \dfrac{e^2}{r}$,

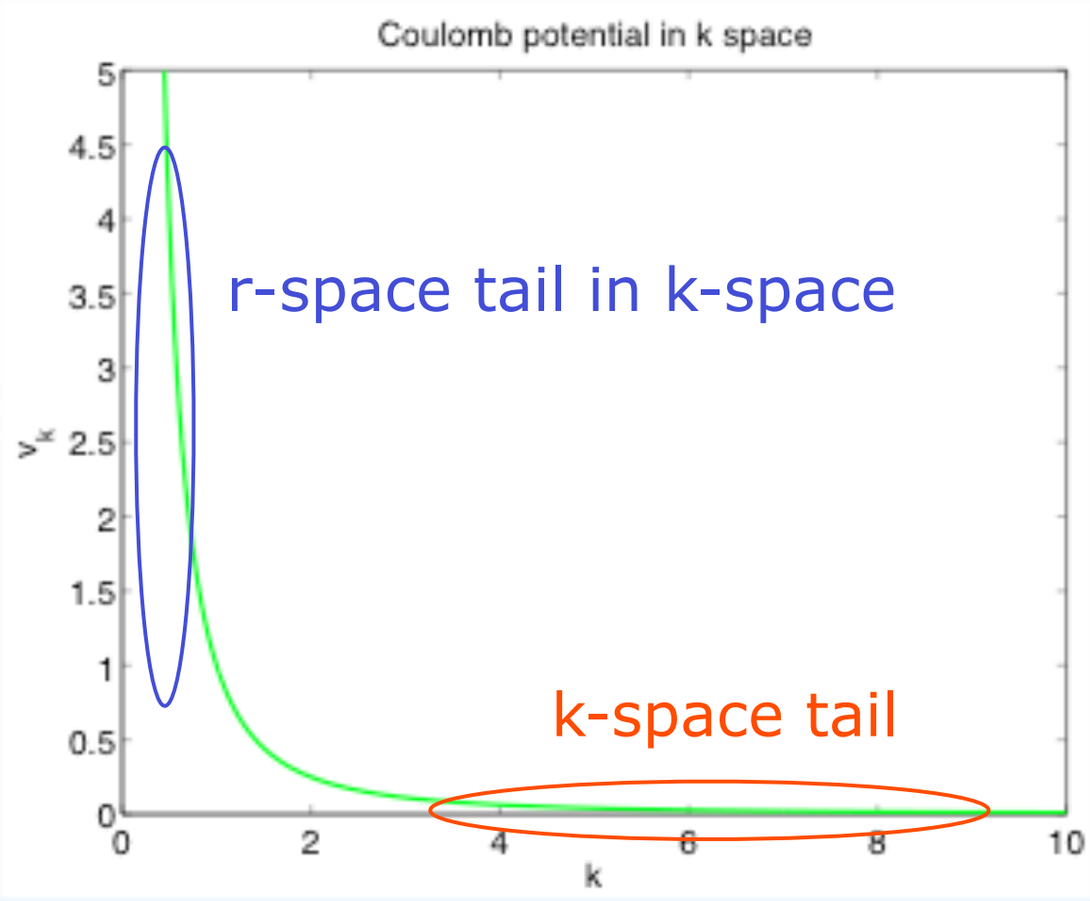
$$v_k = \frac{1}{\Omega}\int dr\, e^{-ik\cdot r} v(r) = \frac{4\pi e^2}{k^2}$$

Potential decays slowly in k-space also.

Origin of Convergence Problem

$$\text{With} \quad f(r) \sim v(r) = \frac{e^2}{r}\,, \quad v_k = \frac{1}{\Omega}\int dr\, e^{-ik\cdot r} v(r) = \frac{4\pi e^2}{k^2}$$





- r-space convergence issue comes from r→∞ (k→0).
- k-space convergence issue comes from k→∞ (r→0).

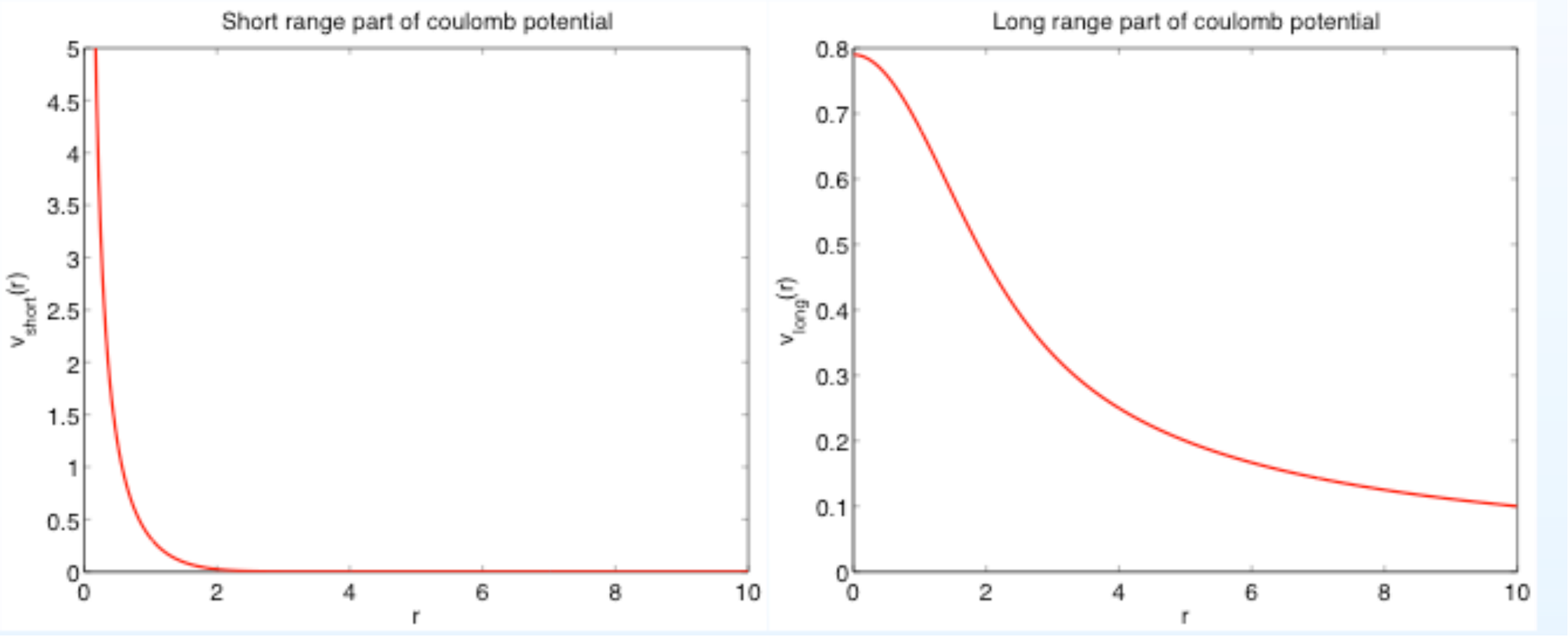
Ewald summation method

- Key idea: Split potential into k-space and real-space parts. We can do since Fourier Transform is linear.

$$V = \sum_{i<j,n} \phi(r_i - r_j + nL) \qquad V = \sum_{k} \phi_k(|\rho_k|^2 - N)$$

$$\rho_k = \sum_{r} e^{ik\cdot r} \qquad \phi_k = \frac{1}{\Omega}\int dr; e^{ik\cdot r}\phi(r) = \frac{4\pi e^2}{k^2}$$

- SLOW convergence at large r (in r-space) and slow at small k (in k-space)



Classic Ewald

- Split up via Gaussian charge distribution

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z dt\, e^{-t^2}$$

$$v_{\mathrm{short}}(r) = \frac{q_1 q_2}{r} \mathrm{erfc}(\kappa\, r) \qquad v_{\mathrm{long}}(r) = \frac{q_1 q_2}{r} - v_{\mathrm{short}}(r) = \frac{q_1 q_2}{r} \mathrm{erf}(\kappa\, r)$$

decays fast for large r

$$v_{\mathrm{long},k} = \frac{4\pi q_1 q_2}{\Omega k^2} e^{-k^2/4\kappa^2}$$

decays fast for large k

- Choose Ewald parameter κ such that $v_{\mathrm{short}}(r) = 0$ at r=L/2.
- Need only one image in real space: min. image potential.

- Total Potential is then:

$$V = C + \sum_{j \neq i} \left[v_{\mathrm{short}}(r_{ij}) + \sum_{|k|<k_c} e^{ik \cdot r_{ij}}\, v_{\mathrm{long},k} \right], \qquad r_{ij} = \min_L |r_i - r_j - L|$$

- Extra term for insulators:

$$V_{\mathrm{dipole}} = \frac{2\pi}{(2\varepsilon + 1)\Omega} \left| \sum_i \mu_i \right|^2$$

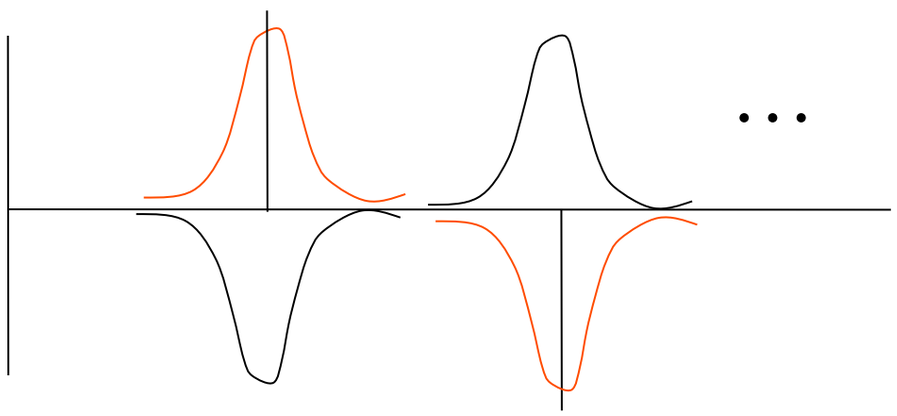
Classic Ewald

- In this conventional break up,

$$v_{\text{short}}(L/2) = \frac{2q_1q_2}{L}\text{erfc}(\kappa L/2) \neq 0$$ **leads to numerical error**

- Summation in k-space is truncated at desired accuracy.
- Adjust κ to minimize total error!

- Physical "trick": Poisson Eq. is linear so we are free to add and subtract charge that conserves system neutrality, making sure charges screen out long-range part.



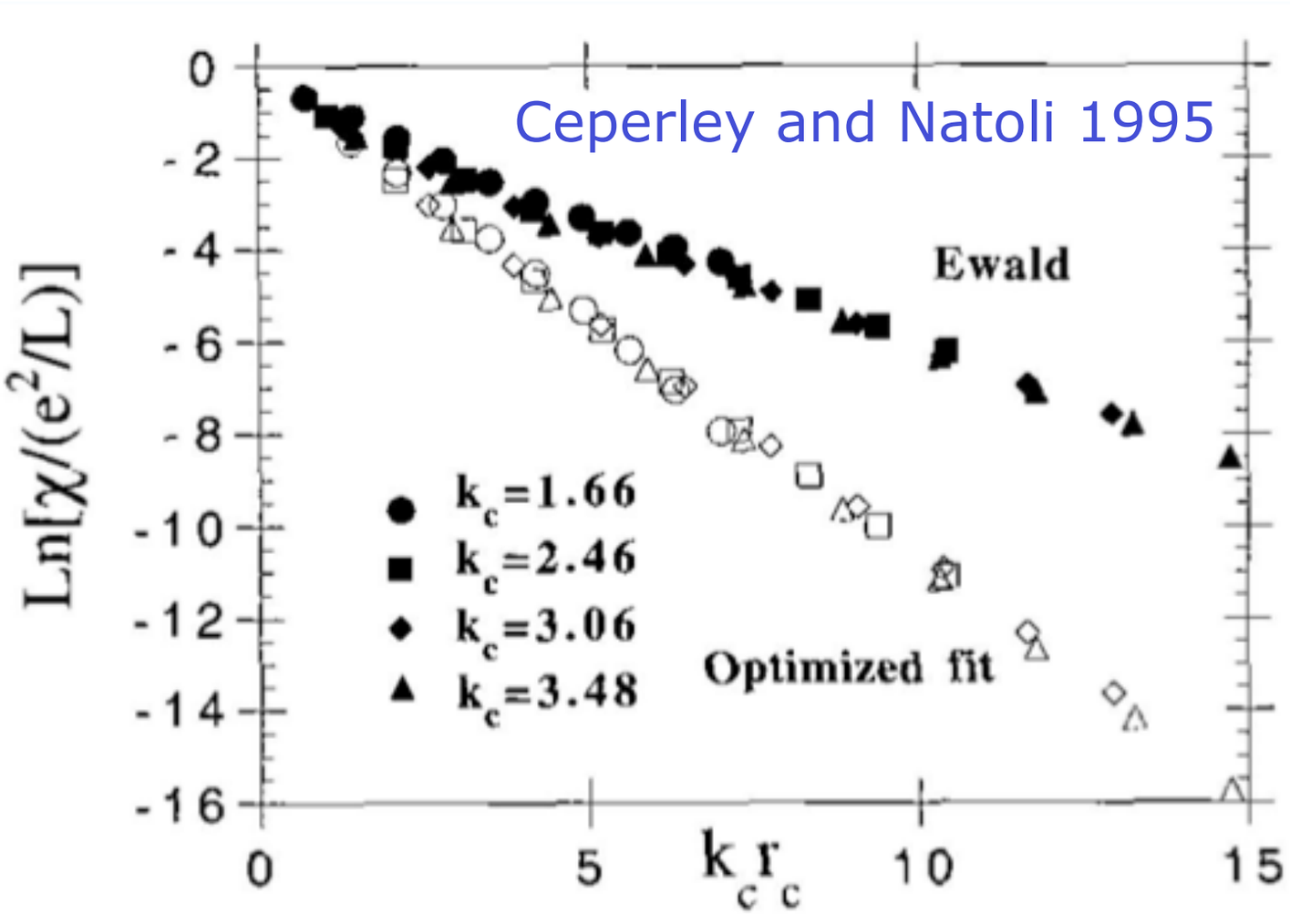
Gaussian charge (black) screens pt. charge.
Gaussian charge (red): Ewald sum!

Ewald: optimizing breakup

- Improve the conventional break up, by

$$v_{\text{short}}(r) = \sum_n c_n \, h_n(r) \quad \text{or} \quad v_{\text{long}}(r) = \sum_n c_n \, h_n(r)$$

- h_n's satisfy B.C.
- Choose k-space cutoff, k_c.
- Write error that comes from neglecting higher k's.
- Minimize error w.r.t. c_n's.



Look at k-space: the algorithm

$$V_{\text{long}} = \frac{1}{2}\sum_{j\neq i}\sum_{k} e^{ik\cdot(r_i - r_j)} v_k = \frac{1}{2}\sum_{ij}\sum_{k} e^{ik\cdot(r_i - r_j)} v_k - \frac{1}{2}\sum_{k} v_k$$

$$= \frac{1}{2}\sum_{k}\left[\sum_{i} e^{ik\cdot r_i}\right]\left[\sum_{j} e^{-ik\cdot r_j}\right] v_k + C = \frac{1}{2}\sum_{k} \rho_k \rho_{-k} v_k + C$$

- $\rho_k = \rho^*_{-k}$ so only have to compute one of them.
- Computation of $\rho_k \sim NM_k$, with M_k the number of k vectors.

Algorithm for computing k-space sums:

```
for k in k-vector-list:
  V_long = V_long +  ρ_k  ρ_-k  v_k
```

Changes due to moving a few particles can be calculated more quickly

Look at k-space: the algorithm

$$V_{\text{long}} = \frac{1}{2}\sum_{k} \rho_k \rho_{-k} v_k + C$$

Algorithm to compute ρ_k

- Complex multiply is much faster that complex exponentiation.
  Use sin's and cos's!

$$e^{ik\cdot r} = \left[e^{i\frac{2\pi}{L_x}r_x}\right]^{m_1}\left[e^{i\frac{2\pi}{L_y}r_y}\right]^{m_2}\left[e^{i\frac{2\pi}{L_z}r_z}\right]^{m_3}$$

Look at k-space: the algorithm

Algorithm for quickly computing ρ_k:

```
Create list of k and (m1 m2 m3) indices
rho = 0       # zero out matrix
for i in particles:
  for j in [1,2,3]:
    Cj[j] = exp(I r[i].b[j])
    for m in range(-mmax,mmax):
      Cjm[j,m] = Cj[j]^m
  for m in mindex_list:
    eikr = Cjm[1,m[1]]*Cjm[2,m[2]]*Cjm[3,m[3]]
    rho[k] = rho[k] + eikr
```

- Use neighbor tables and optimize κ, Ewald is $O(N^{3/2})$
- If we do not reoptimize, then $O(N^2)$
- With efficient code, prefactor is small.

How to do it

- r-space part same as short-ranged potential — $O(N^{3/2})$

- k-space part:

1. Compute $\exp(ik_0x_i) = (\cos(k_0x_i), \sin(k_0x_i))$, $k_0 = 2\pi/L_i$. — $O(N)$
2. Compute powers $\exp(i2k_0x_i) = \exp(ik_0x_i)*\exp(ik_0x_i)$ etc. — $O(N^{3/2})$
   Get all values of $\exp(ik\cdot r_i)$ with just multiplications.
3. Sum over particles to get ρ_k all k. — $O(N^{3/2})$
4. Sum over k to get the potentials. — $O(N^{1/2})$
5. Forces can also be done by taking gradients. — $O(N^{3/2})$

- Constant terms to be added. — $O(1)$
- Checks: perfect cubic lattice: V = -1.4186487/a.

Complexity of Fast Multipole

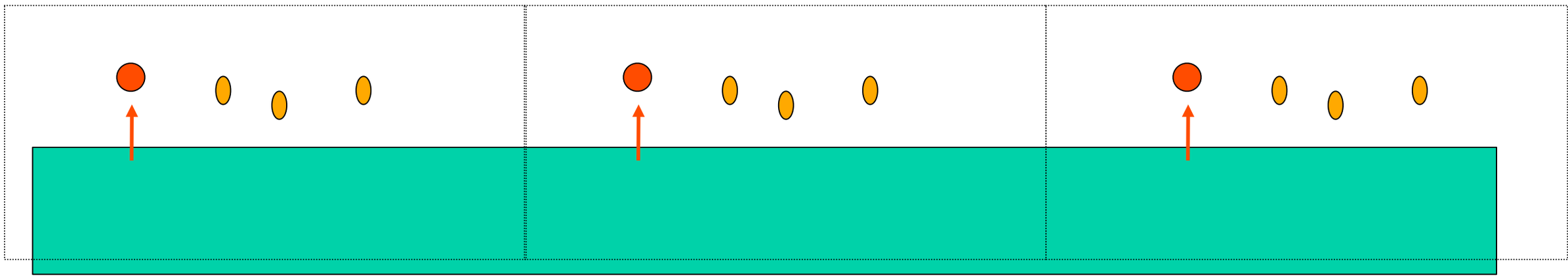
- Coulomb potentials with Ewald sums are $O(N^{3/2})$ if you adjust κ and use neighbor tables.

- Fast Multipole Methods are O(N) for large N.
  - Divide space into cells recursively
  - Find dipole moment of each cell
  - Find rules for how dipole moments for supercells are related to moments for smaller cells.
  - Effective for large systems for molecular dynamics

- Other related method: Particle cell methods (Hockney)
  - Compute the k-space parts on a grid with FFTs.
  - N ln(N)

Problems with Image potential

- Introduces a lattice structure which may not be appropriate.
- Example: a charge layer.



  - We assume charge structure continues at large r.
  - Actually nearby fluid will be anticorrelated.
  - This means such structures will be penalized.

- One should always consider the effects of boundary conditions, particularly when electrostatic forces are around!