Special Topic 3

Computation of Force Fields for Molecular Simulation
Robert Krasny (Michigan)

Outline

• Ewald summation
• treecodes
• Fast Multipole Method
warm-up quiz

“Everyone knows that the real reason universities have students is .......”
warm-up quiz

“Everyone knows that the real reason universities have students is to educate the professors.”

-John Wheeler

(as quoted in *Physics Today*, April 2009)
1. Introduction

Coulomb potential: \( \phi(x) = \frac{1}{4\pi|x|} \)

two properties: singularity at \( x = 0 \), slow decay as \( x \to \infty \)

consider a system of charged particles: \( x_i, q_i, i = 1 : N \)

potential energy: \( V = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} \frac{q_i q_j}{4\pi|x_i - x_j|} : O(N^2) \)

force field: \( F_i = -q_i \sum_{j=1, j \neq i}^{N} q_j \frac{x_i - x_j}{4\pi|x_i - x_j|^3} : O(N^2) \)
2. periodic boundary conditions

potential energy: \[ V = \frac{1}{2} \sum_n \sum_{i,j=1}^N \, ' \, \frac{q_i q_j}{4\pi |x_i - x_j + nL|} \]

cell index: \( n = (n_1, n_2, n_3) \), \( n_i = 0, \pm 1, \ldots \)

/ means omit \( i = j \) for \( n = (0, 0, 0) \)
3. issues with convergence

\[ V = \frac{1}{2} \sum_{n} \sum_{i=1}^{N} \sum_{\substack{j=1 \atop j \neq i}}^{N} \frac{q_i q_j}{4\pi |x_i - x_j|} \]

assume \( \sum_{i=1}^{N} q_i = 0 \) : charge neutrality

\[ \Rightarrow \begin{cases} 
(1) \text{the series for } V \text{ is conditionally convergent} \\
(2) \text{the result depends on the order of summation} \\
(3) \text{regardless of the order, the series converges slowly} 
\end{cases} \]

example

\[ 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} - \frac{1}{6} + \frac{1}{7} - \frac{1}{8} + \cdots = \ln 2 \]

\[ 1 + \frac{1}{3} - \frac{1}{2} + \frac{1}{5} + \frac{1}{7} - \frac{1}{4} + \frac{1}{9} + \frac{1}{11} - \cdots = \frac{3}{2} \ln 2 : \text{reordered} \]

\[ \frac{2}{1.3} + \frac{2}{3.3} + \frac{2}{5.3} + \frac{2}{7.3} + \frac{2}{9.3} + \cdots \cdots \cdots \cdots = \ln 2 : \text{better} \]
4. Ewald summation

\[ V = \frac{1}{2} \sum_n \sum_{i,j=1}^{N} \sum' q_i q_j \frac{4\pi}{4\pi |x_i - x_j + nL|} \]

\[ = V^r + V^k + V^s : \text{one choice} \]

\[ V^r = \frac{1}{2} \sum_n \sum_{i,j=1}^{N} \sum' q_i q_j \frac{\text{erfc}(\alpha |x_i - x_j + nL|)}{4\pi |x_i - x_j + nL|} \quad : \text{real-space term} \]

\[ V^k = \frac{1}{2} \sum_{i,j=1}^{N} q_i q_j \sum_{k \neq 0} e^{-\frac{\pi^2 |k|^2}{L^2\alpha^2} + 2\pi ik \cdot (x_i - x_j)/L} \frac{4\pi^2 L |k|^2}{4\pi^2 L |k|^2} \quad : \text{reciprocal-space term} \]

\[ V^s = -\frac{\alpha}{4\pi^{3/2}} \sum_{i=1}^{N} q_i^2 \quad : \text{self-energy term} \]
5. some ingredients

- \( \text{erf}(r) = \frac{2}{\sqrt{\pi}} \int_0^r e^{-s^2} ds \) :

\[
\text{erfc}(r) = \frac{2}{\sqrt{\pi}} \int_r^\infty e^{-s^2} ds
\]

\[
= 1 - \text{erf}(r)
\]

\( \text{erfc}(0/1/2) = 1.0000/0.1573/0.0047 \) : rapidly decaying

\( f_\alpha(x) = \frac{\alpha^3}{\pi^{3/2}} e^{-\alpha^2|x|^2} \) : Gaussian charge distribution

\[
\int_{\mathbb{R}^3} f_\alpha(x) dx = 1 , \lim_{\alpha \to \infty} f_\alpha(x) = \delta(x) , -\Delta \frac{\text{erf}(\alpha|x|)}{4\pi|x|} = f_\alpha(x)
\]
6. Operation count for Ewald summation

$V^r, V^k$ are evaluated using cutoffs: $|x_i - x_j + nL| < r_c$, $|k| \leq k_c$

large $\alpha$ $\Rightarrow$ \begin{align*} V^r & \text{ converges rapidly} : O(N) \\ V^k & \text{ converges slowly} : O(N^2) \end{align*}

small $\alpha$ $\Rightarrow$ \begin{align*} V^r & \text{ converges slowly} : O(N^2) \\ V^k & \text{ converges rapidly} : O(N) \end{align*}

- Choosing $\alpha, r_c, k_c$ properly, the cost can be reduced to $O(N^{3/2})$ (Fincham, 1993).

- For large $\alpha$, the cost of computing $V^k$ can be reduced to $O(N \log N)$ using the FFT-based particle-mesh Ewald method (Darden-York-Pedersen, 1993). PME is the state of the art.

- For small $\alpha$, the cost of computing $V^r$ can be reduced to $O(N \log N)$ using a treecode (Duan-Krasny, 2000).
7. insight into Ewald’s method (1/3)

\[
\sum_n \frac{1}{4\pi |x + nL|} : \text{lattice sum, diverges, but ok as a distribution}
\]

\[-\Delta \frac{1}{4\pi |x|} = \delta(x) \Rightarrow -\Delta \sum_n \frac{1}{4\pi |x + nL|} = \sum_n \delta(x + nL)\]

define \(\rho(x) = \sum_n \delta(x + nL) - \frac{1}{L^3}\), so \(\int_{\text{cell}} \rho(x) dx = 0\)

define \(\psi(x) : -\Delta \psi(x) = \rho(x)\), PBC

Then \(\psi(x)\) is well-defined (up to an additive constant) and pointwise values can be efficiently computed. How?
7. insight into Ewald’s method (2/3)

\[ \rho(x) = \sum_n \delta(x + nL) - \frac{1}{L^3} \]

\[ = \sum_n (\delta(x + nL) - f_\alpha(x + nL)) + \sum_n f_\alpha(x + nL) - \frac{1}{L^3} \]

\[ \rho(x) = \rho_1(x) + \rho_2(x) : \text{both terms are charge neutral} \]

\[-\Delta \psi_1(x) = \rho_1(x), \quad -\Delta \psi_2(x) = \rho_2(x), \quad \text{PBC} \]

then \( \psi(x) = \psi_1(x) + \psi_2(x) \)
7. insight into Ewald’s method (3/3)

- \( \psi_1(x) = \sum_n \frac{\text{erfc}(\alpha|x + nL|)}{4\pi|x + nL|} - \frac{1}{4L^3\alpha^2} \)

- \( \psi_2(x) = \sum_{k \neq 0} \frac{e^{-\frac{\pi^2|k|^2}{L^2\alpha^2}} + 2\pi ik \cdot x/L}{4\pi^2L|k|^2} \quad \text{: Fourier series} \)

These series are rapidly converging and hence \( \psi(x) = \psi_1(x) + \psi_2(x) \) can be efficiently computed.

proof:
7. insight into Ewald’s method (3/3)

- \( \psi_1(x) = \sum_n \frac{\text{erfc}(\alpha|x + nL|)}{4\pi|x + nL|} - \frac{1}{4L^3\alpha^2} \)

- \( \psi_2(x) = \sum_{k \neq 0} \frac{e^{-\frac{\pi^2|k|^2}{L^2\alpha^2}} + 2\pi ik \cdot x/L}{4\pi^2L|k|^2} : \text{ Fourier series} \)

These series are rapidly converging and hence \( \psi(x) = \psi_1(x) + \psi_2(x) \) can be efficiently evaluated.

proof : homework
7. insight into Ewald’s method (3/3)

- $\psi_1(x) = \sum_n \frac{\text{erfc}(\alpha |x + nL|)}{4\pi |x + nL|} - \frac{1}{4L^3\alpha^2}$

- $\psi_2(x) = \sum_{k \neq 0} \frac{e^{-\frac{\pi^2 |k|^2}{L^2\alpha^2}} + 2\pi ik \cdot x / L}{4\pi^2 L |k|^2}$ : Fourier series

These series are rapidly converging and hence $\psi(x) = \psi_1(x) + \psi_2(x)$ can be efficiently evaluated.

proof : homework

now that we understand all about Ewald summation . . .
Is the Ewald summation still necessary? Pairwise alternatives to the accepted standard for long-range electrostatics

Christopher J. Fennell and J. Daniel Gezelter a)
Department of Chemistry Biochemistry, University of Notre Dame, Notre Dame, Indiana 46556

(Received 24 March 2006; accepted 27 April 2006; published online 19 June 2006)

We investigate pairwise electrostatic interaction methods and show that there are viable computationally efficient (\(O(N)\)) alternatives to the Ewald summation for typical modern molecular simulations. These methods are extended from the damped and cutoff-neutralized Coulombic sum originally proposed by Wolf et al. [J. Chem. Phys. 110, 8255 (1999)]. One of these, the damped shifted force method, shows a remarkable ability to reproduce the energetic and dynamic characteristics exhibited by simulations employing lattice summation techniques. Comparisons were performed with this and other pairwise methods against the smooth particle-mesh Ewald summation to see how well they reproduce the energetics and dynamics of a variety of molecular simulations. © 2006 American Institute of Physics. [DOI: 10.1063/1.2206581]
8. free-space boundary conditions

\[ V = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{j \neq i}^{N} \frac{q_i q_j}{|x_i - x_j|} : \text{potential energy} \]

- direct summation
  - particle-particle, \(O(N^2)\)
- treecode
  - Barnes-Hut (1986), particle-cluster, \(O(N \log N)\)
- Fast Multipole Method
  - Greengard-Rokhlin (1987), cluster-cluster, \(O(N)\) claimed

DISCLOSURE: I work on treecodes. -RK
9. treecode

\[
V = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{j \neq i} q_i q_j \frac{1}{|x_i - x_j|} = \frac{1}{2} \sum_{i=1}^{N} q_i \sum_{c} \sum_{y_j \in c} \frac{q_j}{|x_i - y_j|}
\]

BH used a monopole approximation for the particle-cluster interaction.

\[
\sum_{y_j \in c} \frac{q_j}{|x_i - y_j|} \approx \frac{Q_c}{|x_i - y_c|}, \quad Q_c = \sum_{y_j \in c} q_j
\]
10. multipole approximation for particle-cluster interaction

\[ \sum_{y_j \in c} \frac{q_j}{|x_i - y_j|} = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \frac{M_{l}^m}{r_i^{l+1}} Y_l^m(\theta_i, \phi_i) : \text{multipole expansion} \]

\[ M_{l}^m = \sum_{y_j \in c} q_j r_j^l Y_l^{-m}(\theta_j, \phi_j) : \text{cluster moments} \]

**Cartesian coordinates**

see posters of Boateng, Krasny for

\[ \phi(x) = \frac{\text{erfc}(\alpha |x|)}{|x|}, \quad \frac{e^{-\kappa |x|}}{|x|} \]
11. outline of treecode

- program main
  construct tree
  do $i = 1, N$
    \texttt{compute-potential}(x_i, root)
  end program

- subroutine \texttt{compute-potential}(x, c)
  if particle $x$ and cluster $c$ are well-separated
    compute and store moments of $c$ (unless already available)
    compute particle-cluster interaction by multipole approximation
  else
    if $c$ is a leaf
      compute particle-cluster interaction by direct summation
    else
      call \texttt{compute-potential}(x, c\%child) for each child of $c$
  end subroutine
12. treecode performance for random particles in a cube

(a) relative error, $E_2$

(b) memory usage ratio

CPU (sec)

memory usage (MB)
13. two types of clusters

- **uniform cubes**

- **adapted rectangular boxes**

level 0

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>n</td>
<td>n</td>
<td>i</td>
<td>i</td>
</tr>
<tr>
<td>n</td>
<td>b</td>
<td>n</td>
<td>i</td>
<td>i</td>
</tr>
<tr>
<td>n</td>
<td>n</td>
<td>n</td>
<td>i</td>
<td>i</td>
</tr>
<tr>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
</tr>
</tbody>
</table>

level 1

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
</tr>
<tr>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
</tr>
<tr>
<td>i</td>
<td>i</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>i</td>
<td>i</td>
<td>n</td>
<td>b</td>
<td>n</td>
</tr>
<tr>
<td>i</td>
<td>i</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
<td>i</td>
</tr>
</tbody>
</table>

b : box , n : neighbor , i : interaction list

tools

• shift the center of a multipole expansion (M2M)
• convert a multipole expansion into a local expansion (M2L)
• shift the center of a local expansion (L2L)
15. outline of FMM algorithm

**upward pass**

- form multipole expansion of each box at finest level of tree
- form multipole expansion of each parent by shifting multipole expansions of children to center of parent using M2M

**downward pass**

- starting at coarsest level of tree, convert multipole expansion of each box \(b\) into local expansion about center of each box \(i\) in \(b\)'s interaction list using M2L and add to local expansion in box \(i\)
- starting at coarsest level of tree, shift local expansion of parent to center of each child using L2L and add to local expansion of child
- at finest level of tree, evaluate local expansion at each particle and add to the direct sum with neighbor particles
16. variations of FMM (partial list)

- Poisson integral formula: Anderson (1992)
- plane-wave expansions: Greengard-Rokhlin (1997)

17. other fast summation methods

- multilevel summation: Brandt-Lubrecht (1990)
  Skeel-Tezcan-Hardy (2002)
17. conclusions

• I discussed Ewald summation, treecodes, and the FMM.

• There are several options available for reducing the cost of computing long-range electrostatic interactions.

• There is a need for:
  - more comparison of different methods
  - open source versions
  - better understanding the range of application of PBC
  - automatic tuning of parameters
  - parallel/GPU implementations