

Computation of non-bonded interactions: Part 2

I. Periodic boundary conditions

In most cases the purpose of simulations is to study the properties of an infinite molecular system at a given concentration, temperature etc. The computational power of modern computers allows us to simulate the systems of up to about millions of degrees of freedom. Although this number is large, it is still well below the thermodynamic limit (\sim Avogadro number 10^{23}), which would correspond to the bulk properties. Therefore, without using special tricks the simulation system would be confined to the finite volume, if the hard walls surround the system, or appear as a bubble of atoms in vacuum. In general, the fraction of atoms on the surface relative to the total number of atoms scales as $N^{-1/3}$. Therefore, only for the system of 10^6 atoms this fraction is of an order of 0.01.

The walls confining the system lead to a number of so called finite size effects that distort bulk properties. For example, solvated protein in a cavity with inert walls would experience hydrophobic adsorption on the walls. The common way to avoid finite size effects is to consider periodic boundary conditions (PBC). Consider a molecular system (e.g., protein and solvent) in the cubic box with the dimensions $LxLxL$ (Figure 3.2 in Frenkel and Smit book). Replicate the system and place its exact copies in all the directions to fill the entire 3D space. The original system is called unit cell, all other cells are called images. The atom in a unit cell may now interact with other atoms from the unit cell as well as the atoms from system's images. The total non-bonded energy is given by

$$E_{tot} = \frac{1}{2} \sum_{i,j,\vec{n}} V(|\vec{r}_{ij} + \vec{n}L|),$$

where the sum is taken over all pairs of atoms i and j as well as all images. Each image is specified by the vector \vec{n} (with the integer components). In practice, the cut-offs in the non-bonded interactions limit the sum over \vec{n} to nearest images.

There are few important rules that must be followed when using PBC for proteins:

1. In order to prevent self-interaction of a protein (between protein copies in neighboring images), the sufficient number of water layers must be used. The number of water molecules between the protein and the unit cell boundary must be at least 3 (distance of ~ 10 Å). This will create a minimum water buffer of about 20 Å between protein images.
2. The size of the unit cell L must be more than twice the cut-off distance r_c . This condition eliminates correlated fluctuations that atoms may experience due to simultaneous interactions with two images of a given particle.

The most common geometry of the unit cell is cubic, but more complicated shapes are also used, such as truncated octahedron, hexagonal prism etc. In general, the larger the number of facets and the closer the shape is to a spherical one, the more efficient solvation of a protein becomes. Note that the sphere itself cannot be used for a unit cell. It is also important to be aware that PBC introduce spurious correlated fluctuations with the wavelength of the order of L . Furthermore, the PBC lead to anisotropic radial distributions of densities, which are manifested in the radial correlation functions $g(r)$.

The example of a unit cell for solvated A β 11-42 peptide dimer is displayed in Fig. 1.

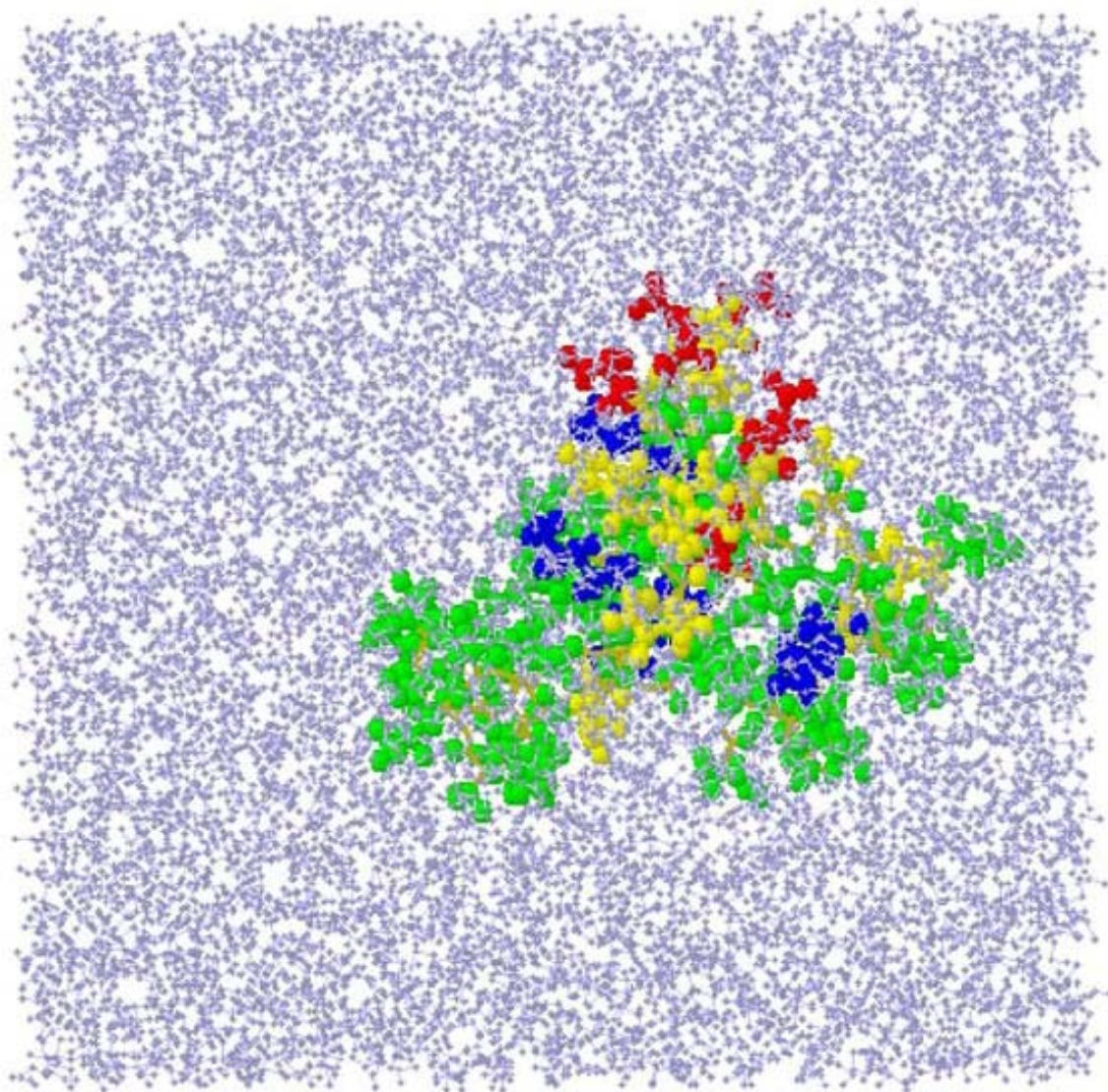


Fig. 1 Solvated A β 11-42 peptide dimer in a large unit cell. The waters colored in blue solvate a peptide dimer shown in green, dark blue, yellow, and red. The dimensions of the unit cell are 57.8 Åx57.8 Åx57.8 Å. It contains 5957 molecules (mostly water)-18643 atoms.

The NAMD configuration file is shown in Box 1. Two keywords determine the set up of the unit cell. `cellBasisVector1(2,3)` are the components of the three basis vectors, which specify the geometry of the unit cell and its dimensions; `cellOrigin` is the position of the unit cell center.

II. Ewald method for computation electrostatic interactions.

Ewald method provides the opportunity to compute electrostatic interactions without using cut-off distances and still avoiding explicit enumeration of all atom pairs i and j . The general idea of the method and its implementation details are as follows. Consider the system of N point charges (net charge is zero) with periodic boundary conditions (assume cubic unit cell with the dimensions $LxLxL$). The electrostatic energy is given by

$$E_{EL} = \frac{1}{2} \sum_i z_i \phi(\vec{r}_i), \quad (1)$$

where

$$\phi(\vec{r}_i) = \sum_{j, \vec{n}} \frac{z_j}{|\vec{r}_{ij} + \vec{n}L|} \quad (2)$$

is the electrostatic potential created by the charges j distributed on all unit cell images (excluding the case $i=j$). It is known that such infinite series as in Eq. (2) are poorly converging. Ewald method gets around this problem by considering two sets of spherically symmetric charge clouds. The screening charge clouds have the signs opposite to the charges z_i and are centered at the position of the charges z_i . The compensating charge clouds have exactly the same charge distribution as the screening ones do, but are assigned the opposite sign. For simplicity, the distributions are assumed to be narrow Gaussians. In this case, the screening clouds partially compensate the point charges at the point r and ensure fast decay of total electrostatic potential.

The idea of the Ewald method is to compute the electrostatic energy as the sum of three components. The first is the interaction of point charges with the compensating charge clouds. Because the distribution of compensating charge clouds is periodic, this term is calculated using Fourier transform in k space. The second term is associated with the interaction of point charges with their own compensating charge clouds. The third term is the interactions of point charges with the other point charges partially screened by the screening charge clouds. The second and the third terms are computed in real space (no Fourier transform involved). Because the introduced charge clouds are narrow, all terms are converging. The final expression for E_{EL} takes on the form

$$E_{EL} = E_{EL}^F - E_{EL}^{self} + E_{EL}^R, \quad (3)$$

where the Fourier part (interactions with compensating clouds)

$$E_{EL}^F = \frac{1}{2} \sum_{\vec{k} \neq 0} \frac{4\pi V}{k^2} |\rho(\vec{k})|^2 e^{-\frac{k^2}{4\alpha}} \text{ and } \rho(\vec{k}) = \frac{1}{V} \sum_{i=1}^N z_i e^{i\vec{k}\vec{r}_i}, \quad (4)$$

the self-interaction term is

$$E_{EL}^{self} = \sqrt{\frac{\alpha}{\pi}} \sum_{i=1}^N z_i^2, \quad (5)$$

and the real-space term (interactions with partially screened charges) is

$$E_{EL}^R = \frac{1}{2} \sum_{i \neq j} \sum_{\vec{n}} z_i z_j \frac{\text{erfc}(\sqrt{\alpha} |\vec{r}_{ij} + \vec{n}L|)}{|\vec{r}_{ij} + \vec{n}L|}. \quad (6)$$

In Eqs. (4-6) α determines the width of Gaussian distribution of screening and compensating charge clouds, V is the volume of the unit cell.

The value of α also determines the convergence of sums over \vec{n} and \vec{k} in Eqs. (4,6). The larger α becomes, the faster real-space term converges (for large α only $n=0$ component survives). However, α has the opposite effect on Eq. (4). Large α causes slower convergence in the sum over \vec{k} . The accuracy of third term computation is determined by α and its typical compromise value is about $(10/L)^2$. Also the trigonometric functions in Eq. (4) are usually computed using fast interpolation methods, which require the evaluation of the function on the grid points within the interval L , the total number of which should be equal to the product of small integer numbers (2, 3 or 5). The rule of thumb is to choose the grid in such a way that the interval between grid points is less than 2 \AA . For example, in Box 1 for $L=57.8 \text{ \AA}$ the number of grid points is 2^5 , which gives the grid interval of $L/2^5 \approx 1.8 \text{ \AA}$. After proper selection of α and mesh interpolation the final scaling of Ewald method is $N \log N$.

NAMD offers Ewald method for computing electrostatic interactions. The basic keywords are as follows (Box 1). To turn on the Ewald sums the keyword `PME` must be set to `on`. The keyword `PMETolerance` sets the accuracy in the computation of real-space term (Eq. (6)). The keywords `PMEGridSizeX(Y,Z)` specify the number of grid points in three directions. The keyword `PMEInterpOrder` determines the quality of function interpolation between grid points. In most cases, only `PMEGridSizeX(Y,Z)` should be adjusted depending on the system size, all other parameters may be set at their default values that should provide reasonable performance.

As described in the previous lecture the keyword `switchdist` and `cutoff` determine the interval, over which the direct computation of electrostatic interactions is replaced with Ewald sum.

Box 1. Example of the NAMD configuration file for A β 11-42 dimer

```
# input system.....
structure ab_m2_dimer_solv.psf
coordinates ./output/ab_quench04149.coor
bincoordinates ./output/abi_quench04149.coor
binvelocities ./output/abi_quench04149.vel

#..force field.....
paratypecharm on
parameters par_all22_na.inp
parameters par_all22_prot.inp
exclude scaled1-4
1-4scaling 1.0
dielectric 1.0
switching on
switchdist 8.0
cutoff 12.0
pairlistdist 13.5
margin 0.0
stepspercycle 20
rigidBonds all
rigidTolerance 0.00001
rigidIterations 100

# Ewald EL.....
PME on # turning on PME
PMEtolerance 0.000001 # accuracy of computing real-space term
PMEGridSizeX 32 # grids for fast evaluation in Fourier sum
PMEGridSizeY 32 # note that 32=25
PMEGridSizeZ 32 #

#integrator .....
timestep 1.0
fullElectFrequency 4
```

```

#output.....
outputenergies 1000
outputtiming 1000
binaryoutput no
outputname output/ab_quench04150
restartname output/abi_quench04150
restartfreq 10000
binaryrestart yes
DCDfile output/ab_quench04150.dcd
dcdfreq 1000

#MD protocol.....
seed 32204150
numsteps 40000
#temperature 300
#rescaleFreq 1
#rescaleTemp runtemp
#reassignFreq 1
#reassignTemp 300
#reassignIncr 0.001
#reassignHold 300

# periodic boundary conditions.....
cellBasisVector1 57.8 0.0 0.0 # components of the unit cell basis
vectors
cellBasisVector2 0.0 57.8 0.0 #
cellBasisVector3 0.0 0.0 57.8 #
cellOrigin 0.0 0.0 0.0 # position of the unit cell center
wrapWater on

```

Reference: Annual Review of Biophysics and Biomolecular Structure **28**, 155 (1999).