

MD tutorial 3: Ar atoms

I found that the MD simulation using DL-POLY is not the hands-off calculation like a gaussian G03 program. You should make something before and after the simulation. But I like it because it is not blackbox simulation.

Ar atom:

atomic number 18
electron $1s^2 2s^2 2p^6 3s^2 3p^6$

 neutral inert gas

atomic weight 39.948

If there is no interaction between Ar atoms,
they are always in the gas state?

But

from experimental data

Melting point: 83.8 K, Boiling point 87.3 K

Critical temperature 150.8 K

Enthalpy of vaporization 6.5 kJ mol^{-1}

What's the origin of the interatomic force?

- attractive force: van der Waals force from instantaneous dipole-dipole interaction

$$\propto -\frac{c}{r^6}$$

- in the following you can see the origin of vdW interaction and c is given

$$c = \frac{3}{2} \frac{\alpha_1 \alpha_2}{(4\pi\epsilon_0)^2} \frac{I_1 I_2}{I_1 + I_2}$$

- repulsive force : overlapping of wavefunctions

One can assume that

$$\propto \frac{c'}{r^{12}}$$

Origin of the attractive part of Lennard-Jones potential

Masahiro Yamamoto

*Department of Energy and Hydrocarbon Chemistry,
Graduate School of Engineering, Kyoto University, Kyoto 606-8501, Japan*

(Dated: March 4, 2008)

Now we consider two hydrogen atoms. If the two atoms are located far away, the Hamiltonian H_0 and the wavefunction Ψ of the system is given by two independent H atoms,

$$H_0 = -\frac{\hbar^2}{2m}(\nabla_1^2 + \nabla_2^2) - \frac{e^2}{4\pi\epsilon_0 r_1} - \frac{e^2}{4\pi\epsilon_0 r_2} \quad (1)$$

$$\Psi = \psi_1(\mathbf{r}_1)\psi_2(\mathbf{r}_2) = R_{nl}(r_1)Y_{lm}(\hat{\mathbf{r}}_1)R_{nl}(r_2)Y_{lm}(\hat{\mathbf{r}}_2) = |nlm\rangle|nlm\rangle \quad (2)$$

The ground state Ψ_0 is given by $|100\rangle|100\rangle$. If the two H atoms get closer, they interact each other and the potential

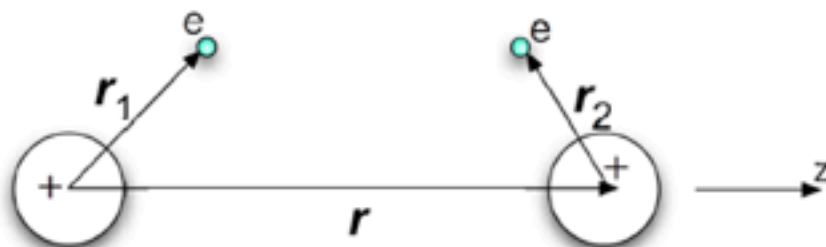


FIG. 1: van der Waals interaction between two hydrogen atom

V (perturbation) is given by

$$V = \frac{e^2}{4\pi\epsilon_0 r} - \frac{e^2}{4\pi\epsilon_0} \frac{1}{|-\mathbf{r}_1 + \mathbf{r}|} - \frac{e^2}{4\pi\epsilon_0} \frac{1}{|\mathbf{r} + \mathbf{r}_2|} + \frac{e^2}{4\pi\epsilon_0} \frac{1}{|-\mathbf{r}_1 + \mathbf{r} + \mathbf{r}_2|} \quad (3)$$

If we assume $r \gg r_1, r_2$, and vector \mathbf{r} is in the z direction. $\mathbf{r} = (0, 0, z)$. If we consider the order up to $r^{-1}(x_1^2/r^2)$ because the $(1/r)(z_1/r)$ order terms are cancelled out. If we use $(1+x)^{-1/2} \simeq 1 - x/2 + 3x^2/8\dots$

$$\begin{aligned} |\mathbf{r} - \mathbf{r}_1|^{-1} &= [(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2]^{-1/2} = r^{-1} \left[\frac{z^2 - 2zz_1 + x_1^2 + y_1^2 + z_1^2}{r^2} \right]^{-1/2} \\ &= r^{-1} \left[1 + \frac{-2zz_1 + x_1^2 + y_1^2 + z_1^2}{r^2} \right]^{-1/2} \simeq r^{-1} \left(1 - \frac{-2zz_1 + x_1^2 + y_1^2 + z_1^2}{2r^2} + \frac{3z_1^2}{2r^2} \right) \\ |\mathbf{r} + \mathbf{r}_2|^{-1} &\simeq r^{-1} \left(1 - \frac{2zz_2 + x_2^2 + y_2^2 + z_2^2}{2r^2} + \frac{3z_2^2}{2r^2} \right) \\ |\mathbf{r} - \mathbf{r}_1 + \mathbf{r}_2|^{-1} &\simeq r^{-1} \left[1 - \frac{-2zz_1 + 2zz_2 + x_1^2 - 2x_1x_2 + x_2^2 + y_1^2 - 2y_1y_2 + y_2^2 + z_1^2 - 2z_1z_2 + z_2^2}{2r^2} + \frac{3(z_2 - z_1)^2}{2r^2} \right] \\ V &\simeq \frac{e^2}{4\pi\epsilon_0 r} \left(1 - 1 + \frac{-2zz_1 + x_1^2 + y_1^2 + z_1^2}{2r^2} - \frac{3z_1^2}{2} - 1 + \frac{2zz_2 + x_2^2 + y_2^2 + z_2^2}{2r^2} - \frac{3z_2^2}{2} \right. \\ &\quad \left. + 1 - \frac{-2zz_1 + 2zz_2 + x_1^2 - 2x_1x_2 + x_2^2 + y_1^2 - 2y_1y_2 + y_2^2 + z_1^2 - 2z_1z_2 + z_2^2}{2r^2} + \frac{3z_1^2 + z_2^2 - 2z_1z_2}{r^2} \right) \\ &= \frac{e^2}{4\pi\epsilon_0 r^3} (x_1x_2 + y_1y_2 + z_1z_2 - 3z_1z_2) = \frac{e^2}{4\pi\epsilon_0 r^3} (x_1x_2 + y_1y_2 - 2z_1z_2) \quad (4) \end{aligned}$$

This is the instantaneous dipole-dipole interaction $1/(4\pi\epsilon_0 r^3)[\vec{\mu}_1 \cdot \vec{\mu}_2 - 3(\vec{\mu}_1 \cdot \hat{\mathbf{r}})(\vec{\mu}_2 \cdot \hat{\mathbf{r}})]$. Here $\vec{\mu}_1 = e(-\mathbf{r}_1)$, $\vec{\mu}_2 = e(-\mathbf{r}_2)$. Using perturbation theory the energy change is given by

$$E_n = E_n^{(0)} + \Delta_n^{(1)} + \Delta_n^{(2)} + \dots \quad (5)$$

$$\Delta_n^{(1)} = \langle n|V|n \rangle = V_{nn} \quad (6)$$

$$\Delta_n^{(2)} = \sum_{k \neq n} \frac{|V_{nk}|^2}{E_n^{(0)} - E_k^{(0)}} \quad (7)$$

For ground state $n = 0$, $E_0^{(0)} = \epsilon_0(\text{atom1}) + \epsilon_0(\text{atom2})$ and

$$\begin{aligned}
\Delta_n^{(1)} &= \langle 0 | \frac{e^2}{4\pi\epsilon_0 r^3} (x_1 x_2 + y_1 y_2 - 2z_1 z_2) | 0 \rangle \\
&= \frac{e^2}{4\pi\epsilon_0 r^3} \langle 1, 0, 0(\text{atom1}) | \langle 1, 0, 0(\text{atom2}) | (x_1 x_2 + y_1 y_2 - 2z_1 z_2) | 1, 0, 0(\text{atom1}) \rangle | 1, 0, 0(\text{atom2}) \rangle \\
&= \frac{e^2}{4\pi\epsilon_0 r^3} \langle 1, 0, 0(\text{atom1}) | x_1 | 1, 0, 0(\text{atom1}) \rangle \langle 1, 0, 0(\text{atom2}) | x_2 | 1, 0, 0(\text{atom2}) \rangle + \dots \\
&= 0
\end{aligned} \tag{8}$$

In the last equation the integral of the product of even, odd, even functions becomes zero. and the second order

$$\begin{aligned}
\Delta_0^{(2)} &= \sum_{k \neq 0} \frac{|V_{0k}|^2}{\epsilon_0(\text{atom1}) + \epsilon_0(\text{atom2}) - \epsilon_k(\text{atom1}) + \epsilon_k(\text{atom2})} \\
&= \frac{e^4}{(4\pi\epsilon_0)^2 r^6} \sum_{k \neq 0} \frac{\langle 0 | x_1 x_2 + y_1 y_2 - 2z_1 z_2 | k \rangle \langle k | x_1 x_2 + y_1 y_2 - 2z_1 z_2 | 0 \rangle}{\epsilon_0(\text{atom1}) + \epsilon_0(\text{atom2}) - \epsilon_k(\text{atom1}) + \epsilon_k(\text{atom2})}
\end{aligned} \tag{9}$$

Using the relation $\sum_k |k\rangle\langle k| = 1 = |0\rangle\langle 0| + \sum_{k \neq 0} |k\rangle\langle k|$, and we assume that the denominator is constant and $\epsilon_0(\text{atom1}) + \epsilon_0(\text{atom2}) - \epsilon_k(\text{atom1}) + \epsilon_k(\text{atom2}) \simeq -(I_1 + I_2)$, here I_1, I_2 is the ionization energy of atom1 and 2.

$$\begin{aligned}
\Delta_0^{(2)} &= -\frac{e^4}{(4\pi\epsilon_0)^2 r^6} \frac{1}{I_1 + I_2} \langle 0 | (x_1 x_2 + y_1 y_2 - 2z_1 z_2)^2 | 0 \rangle \\
&= -\frac{e^4}{(4\pi\epsilon_0)^2 r^6} \frac{1}{I_1 + I_2} [\langle 0(\text{atom1}) | x_1^2 | 0(\text{atom1}) \rangle \langle 0(\text{atom2}) | x_2^2 | 0(\text{atom2}) \rangle \\
&\quad + \langle 0(\text{atom1}) | y_1^2 | 0(\text{atom1}) \rangle \langle 0(\text{atom2}) | y_2^2 | 0(\text{atom2}) \rangle + 4 \langle 0(\text{atom1}) | z_1^2 | 0(\text{atom1}) \rangle \langle 0(\text{atom2}) | z_2^2 | 0(\text{atom2}) \rangle]
\end{aligned}$$

In the same perturbation theory the atomic polarizability is given by

$$\alpha = -2e^2 \sum_{k \neq 0} \frac{\langle 0 | z | k \rangle \langle k | z | 0 \rangle}{E_0^{(0)} - E_k^{(0)}} \simeq -\frac{2e^2}{I} \langle 0 | z^2 | 0 \rangle \tag{10}$$

Here the electric field is in z -direction. Then we can get $\langle 0 | z^2 | 0 \rangle = \langle 0 | x^2 | 0 \rangle = \langle 0 | y^2 | 0 \rangle = -(\alpha I)/(2e^2)$ With this relation

$$\begin{aligned}
\Delta_0^{(2)} &= -\frac{e^4}{(4\pi\epsilon_0)^2 r^6} \frac{1}{I_1 + I_2} \left[\frac{\alpha_1 I_1}{2e^2} \frac{\alpha_2 I_2}{2e^2} + \frac{\alpha_1 I_1}{2e^2} \frac{\alpha_2 I_2}{2e^2} + 4 \frac{\alpha_1 I_1}{2e^2} \frac{\alpha_2 I_2}{2e^2} \right] \\
&= -\frac{3}{2} \frac{\alpha_1 \alpha_2}{(4\pi\epsilon_0)^2 r^6} \frac{I_1 I_2}{I_1 + I_2}
\end{aligned} \tag{11}$$

This equation was derived by F. London (Z. Phys. 63, 245-279, 1930). Since the vdW interaction can not be estimated from usual density functional theory(because we should consider excitation and dynamical dipole-dipole interaction), some semiempirical correction based on Eq.(11) is reported. (e.g. F. Ortman et al. PRB, 73, 205101, 2006)

The final result give the attractive part of the Lennard-Jones potential, $\phi(r) = 4\epsilon[(\sigma/r)^{12} - (\sigma/r)^6]$. In the mixed system one can use the following convention(Lorentz-Berthelot mixing rule)

$$\sigma_{ij} = \frac{\sigma_i + \sigma_j}{2} \quad (12)$$

$$\epsilon_{ij} = \sqrt{\epsilon_i \epsilon_j} \quad (13)$$

In the case of σ_{ij} we can understand the rule from the view point of hard-sphere. In the case of ϵ_{ij} we should consider Eq.(11)

$$\begin{aligned} -4\epsilon_i \sigma_i^6 &= -\frac{3}{2} \frac{\alpha_i^2}{(4\pi\epsilon_0)^2} \frac{I_i}{2}, & -4\epsilon_j \sigma_j^6 &= -\frac{3}{2} \frac{\alpha_j^2}{(4\pi\epsilon_0)^2} \frac{I_j}{2} \\ \epsilon_{ij} &= \sqrt{\epsilon_i \epsilon_j} = \frac{3}{16} \frac{\alpha_i \alpha_j}{(4\pi\epsilon_0)^2} \frac{\sqrt{I_i I_j}}{\sigma_i^3 \sigma_j^3} \end{aligned} \quad (14)$$

From the final equation the approximation of Eq.(13) may be not bad. If one consider the deviation form the standard mixing rule, the modified Lorentz-Berthelot mixing rule can be used. ξ is the deviation parameter.

$$\epsilon_{ij} = \xi \sqrt{\epsilon_i \epsilon_j}, \quad \sigma_{ij} = \frac{\sigma_i + \sigma_j}{2} \quad (15)$$

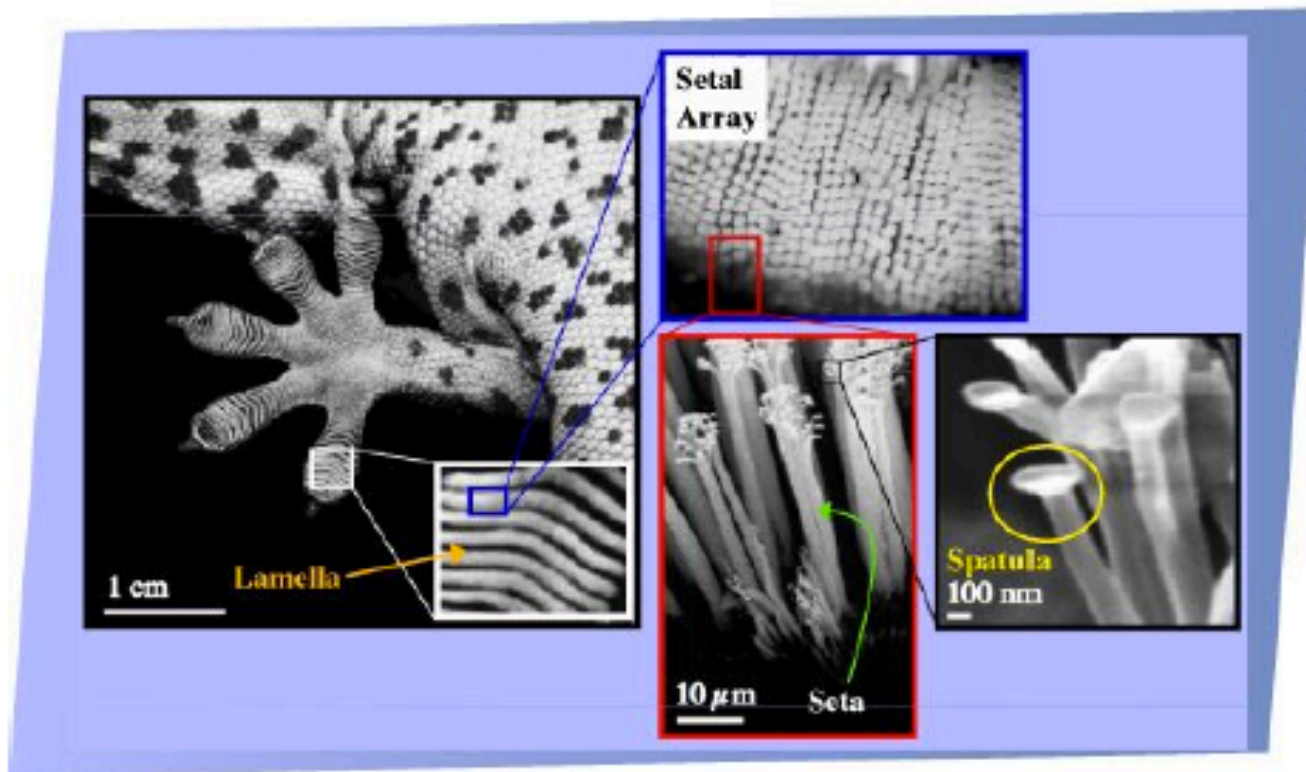


FIG. 2: from http://socrates.berkeley.edu/~peattiea/research_main.html

The Gecko (やもり) can stick to the wall. (both on the hydrophilic and hydrophobic surface and even dead gecko can stick!!) They have 1 billion contacting point per foot and interact via van der Waals force!! The total adhesive stress from Autumn's force measurements is 576 kPa (5.68 atm).

vdW interaction (or Lennard-Jones) is the one of the most important intermolecular interaction and it's always used in MD calculation.

Lennard-Jones Potential

$$V(r) = -\frac{c}{r^6} + \frac{c'}{r^{12}} = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right]$$

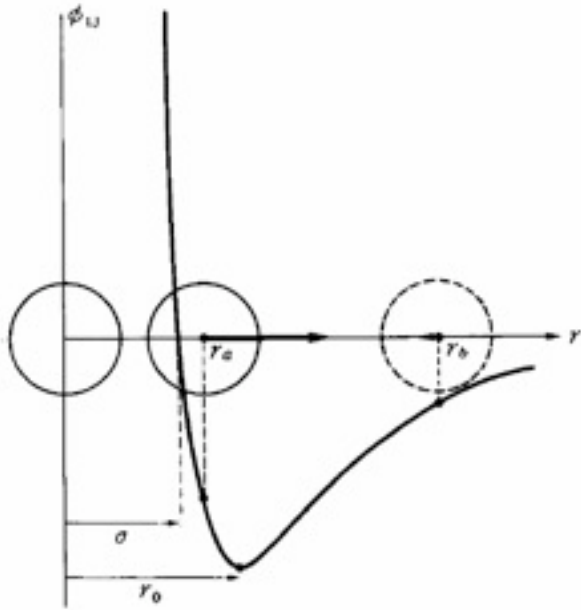


図 2.1 レナード-ジョーンズポテンシャル

$$V(\sigma) = 0$$

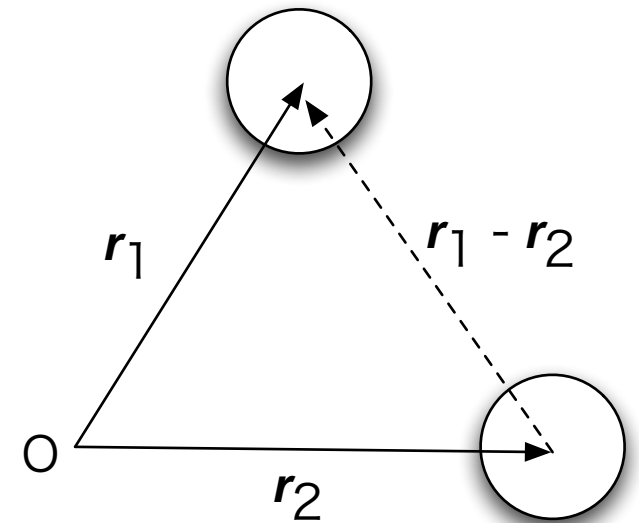
$$V(2^{1/6}\sigma) = 4\epsilon\left(\frac{1}{4} - \frac{1}{2}\right) = -\epsilon, \quad r_0 = 2^{1/6}\sigma$$

Force

$$\begin{aligned} F(r) &= -\frac{dV(r)}{dr} = 4\epsilon \left[12 \left(\frac{\sigma}{r}\right)^{11} (-\sigma r^{-2}) - 6 \left(\frac{\sigma}{r}\right)^5 (-\sigma r^{-2}) \right] \\ &= -\frac{24\epsilon}{r} \left[2 \left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right] \end{aligned}$$

$$F(r_0) = -\frac{24\epsilon}{r} (2/4 - 1/2) = 0, \quad F(\sigma) = -24\epsilon/\sigma$$

If there are two Ar atoms
in the system, the equation of
motion becomes



$$\vec{F}_i = -\nabla_i V, \quad \left[\text{means } (F_i)_x = -\frac{\partial V}{\partial x_i}, (F_i)_y = -\frac{\partial V}{\partial y_i}, (F_i)_z = -\frac{\partial V}{\partial z_i} \right], \quad i = 1, 2$$

$$V = V(|r_1 - r_2|), \quad |r_1 - r_2| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

for example

$$\frac{\partial V}{\partial x_1} = \frac{\partial V}{\partial |r_1 - r_2|} \frac{\partial |r_1 - r_2|}{\partial x_1} = \frac{dV}{dr} \Bigg|_{r=|r_1 - r_2|} \frac{1}{2|r_1 - r_2|} 2(x_1 - x_2) = V'(|r_1 - r_2|) \frac{x_1 - x_2}{|r_1 - r_2|}$$

Then, equation of motion become

$$\begin{aligned} m_1 \frac{d^2}{dt^2} \mathbf{r}_1 &= -V'(|r_1 - r_2|) \frac{x_1 - x_2}{|r_1 - r_2|} \mathbf{i} - V'(|r_1 - r_2|) \frac{y_1 - y_2}{|r_1 - r_2|} \mathbf{j} - V'(|r_1 - r_2|) \frac{z_1 - z_2}{|r_1 - r_2|} \mathbf{k} \\ &= -V'(|r_1 - r_2|) \hat{\mathbf{r}}_{12}, \quad \hat{\mathbf{r}}_{12} = \frac{\mathbf{r}_1 - \mathbf{r}_2}{|r_1 - r_2|} \end{aligned}$$

$$m_2 \frac{d^2}{dt^2} \mathbf{r}_2 = V'(|r_1 - r_2|) \hat{\mathbf{r}}_{12}$$

Lennard-Jones n - l Potential

$$\phi(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^n - \left(\frac{\sigma}{r} \right)^l \right]$$

$$\phi'(r) = \frac{d\phi(r)}{dr} = -4\frac{\epsilon}{r} \left[n \left(\frac{\sigma}{r} \right)^n - l \left(\frac{\sigma}{r} \right)^l \right]$$

$$\phi(\sigma) = 0$$

$$\phi'(r_0) = 0, \quad r_0 = \sigma \left(\frac{n}{l} \right)^{\frac{1}{n-l}}$$

$$\phi(r_0) = 4\epsilon \left[\left(\frac{l}{n} \right)^{\frac{n}{n-l}} - \left(\frac{l}{n} \right)^{\frac{l}{n-l}} \right]$$

$$m_k \frac{d^2}{dt^2} \vec{\mathbf{r}}_k = - \sum_i \left[\phi'(r_{ki}) \frac{x_k - x_i}{r_{ki}} \mathbf{i} + \phi'(r_{ki}) \frac{y_k - y_i}{r_{ki}} \mathbf{j} + \phi'(r_{ki}) \frac{z_k - z_i}{r_{ki}} \mathbf{k} \right]$$

In the case of many Ar atoms:

If the interaction between i and j is pairwise, the potential energy is given by

$$\mathcal{V} = \frac{1}{2} \sum_{i,j} \phi(|\mathbf{r}_i - \mathbf{r}_j|) = \sum_{i < j} \phi(|\mathbf{r}_i - \mathbf{r}_j|)$$

$$\Phi_{ij}(\mathbf{r}_i, \mathbf{r}_j) = \Phi_{ij}(|\mathbf{r}_i - \mathbf{r}_j|)$$

The force acting on the atom k can be written as

$$\vec{\mathbf{f}}_k = -\nabla_k \mathcal{V} = -\left(\mathbf{i} \frac{\partial}{\partial x_k} + \mathbf{j} \frac{\partial}{\partial y_k} + \mathbf{k} \frac{\partial}{\partial z_k} \right) \mathcal{V}$$

$$r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$$

$$\frac{\partial |\mathbf{r}_i - \mathbf{r}_j|}{\partial x_k} = \frac{1}{2} |\mathbf{r}_i - \mathbf{r}_j|^{-1} 2(x_i - x_j)(\delta_{ki} - \delta_{kj})$$

$$\vec{\mathbf{f}}_k = -\frac{1}{2} \sum_{i,j} \left[\mathbf{i} \phi'(r_{ij}) \frac{x_i - x_j}{r_{ij}} + \mathbf{j} \phi'(r_{ij}) \frac{y_i - y_j}{r_{ij}} \right.$$

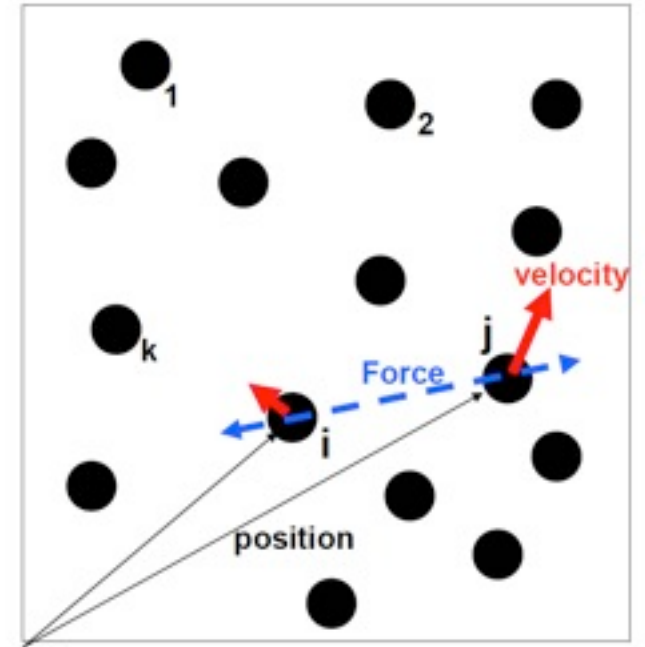
$$\left. + \mathbf{k} \phi'(r_{ij}) \frac{z_i - z_j}{r_{ij}} \right] (\delta_{ki} - \delta_{kj})$$

$$= -\sum_i \left[\phi'(r_{ki}) \frac{x_k - x_i}{r_{ki}} \mathbf{i} + \phi'(r_{ki}) \frac{y_k - y_i}{r_{ki}} \mathbf{j} \right.$$

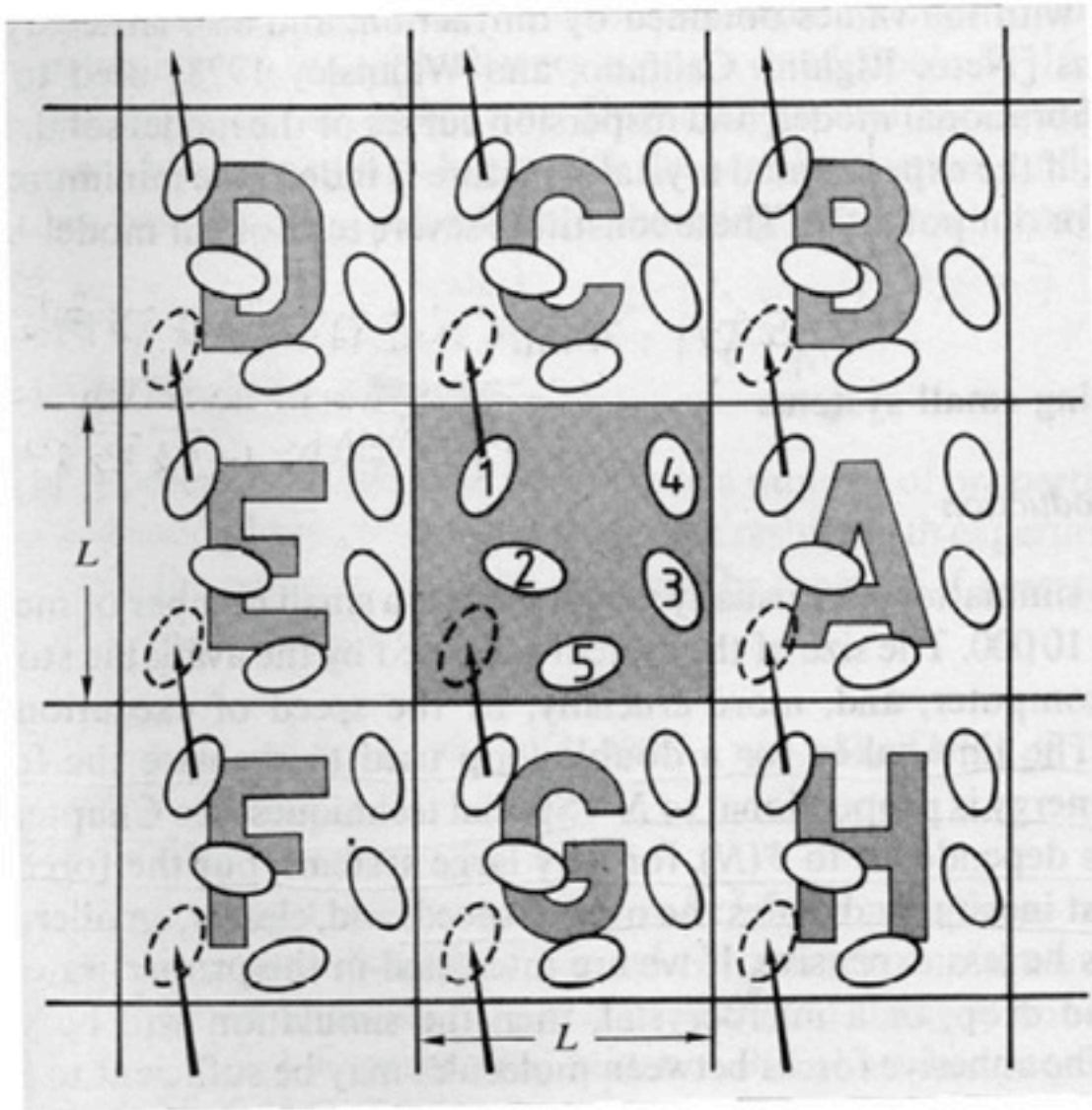
$$\left. + \phi'(r_{ki}) \frac{z_k - z_i}{r_{ki}} \mathbf{k} \right]$$

$$m_k \frac{d^2}{dt^2} \vec{\mathbf{r}}_k = -\sum_i \left[\phi'(r_{ki}) \frac{x_k - x_i}{r_{ki}} \mathbf{i} + \phi'(r_{ki}) \frac{y_k - y_i}{r_{ki}} \mathbf{j} \right.$$

$$\left. + \phi'(r_{ki}) \frac{z_k - z_i}{r_{ki}} \mathbf{k} \right]$$



Periodic Boundary Condition



Allen and Tildesley, Computer simulation of liquids, Oxford

```
c --periodic boundary condition
axh=ax/2.0d0
ayh=ay/2.0d0
azh=az/2.0d0
do i=1, natom
  if (x(i) > axh) x(i)=x(i)-ax
  if (x(i) < -axh) x(i)=x(i)+ax
  if (y(i) > ayh) y(i)=y(i)-ay
  if (y(i) < -ayh) y(i)=y(i)+ay
  if (z(i) > azh) z(i)=z(i)-az
  if (z(i) < -azh) z(i)=z(i)+az
enddo
```

To neglect the effect of wall, PBC is used.

When the PBC is set up, the potential should be cut-off, because a molecule in the center cell can interact with the self-image in the neighboring image cell.

To avoid that the total energy is not conserved by introduction of the cutoff, the potential is shifted

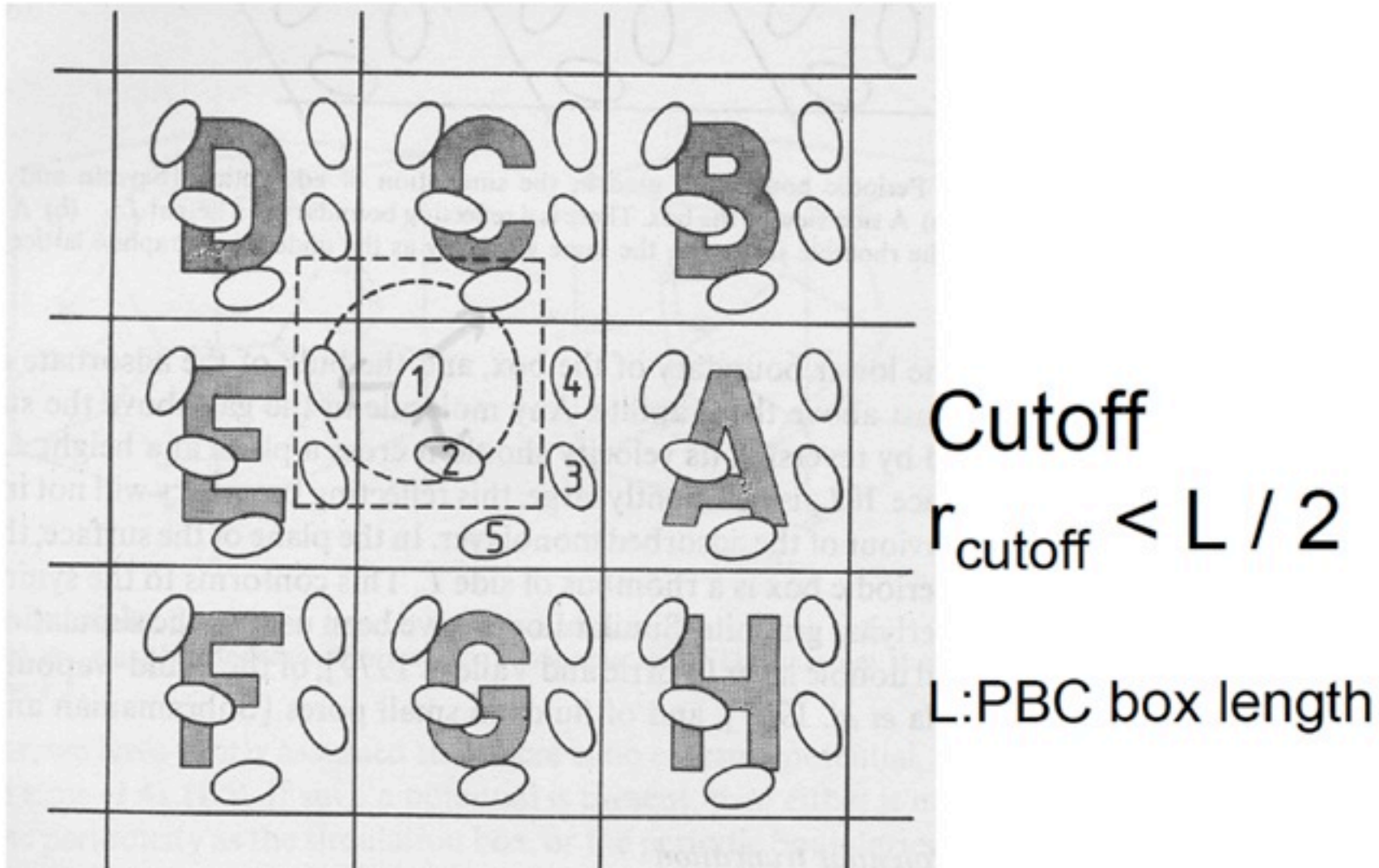
$$v^S(r) = \begin{cases} v(r) - v(r_c) & r \leq r_c \\ 0 & r > r_c \end{cases}$$

To avoid the discontinuity of the force at the cutoff, the force-shifted potential is sometimes introduced

$$v^{SF}(r) = \begin{cases} v(r) - v(r_c) - \frac{dv(r)}{dr} \Big|_{r_c} (r - r_c) & r \leq r_c \\ 0 & r > r_c \end{cases}$$

The correction by the introduction of cutoff should be done.

Minimum image convention



Allen and Tildesley, Computer simulation of liquids, Oxford

Should consider the interaction between 1(central cell) and 4 (cell E)

minimum image convention in the program (by MY)

For fortran study I strongly recommend 1) “JIS FORTRAN入門(上) 森口繁一 東大出版”

and 2) F90 <http://ace.phys.h.kyoto-u.ac.jp/~tomita/education/fortran90/sec0.html> by 富田博之.

```
c ----- calculate potential -----
c ---i-j pair include minimum image --
do i=1,natom-1
  do j=i+1, natom
    xij=x(i)-x(j)
    yij=y(i)-y(j)
    zij=z(i)-z(j)
    if (abs(xij) > rcut)
      &      xij=xij-sign(1.0,xij)*ax ! when |x_i - x_j| > r_cut
      if (abs(xij) > rcut) cycle ! check the image cell x_i - x_j - sign(xij)*ax
      ! sign(1.0,x) = 1.0 when x > 0, = -1.0 when x < 0
    if (abs(yij) > rcut)
      &      yij=yij-sign(1.0,yij)*ay
    if (abs(yij) > rcut) cycle
    if (abs(zij) > rcut)
      &      zij=zij-sign(1.0,zij)*az
    if (abs(zij) > rcut) cycle ! cycle means skip this do-loop
    r2ij=xij*xij+yij*yij+zij*zij
    if (r2ij > r2cut) cycle
    rij=sqrt(r2ij)
    .....
  enddo
enddo
```

寄り道: Programming のおすすめ again

特にSEで就職する方には必携かもね！

For Fortran study I strongly recommend

1) “JIS FORTRAN入門(上) 森口繁一 東大出版”

(It takes just one day) and in the second step

2) F90 <http://ace.phys.h.kyoto-u.ac.jp/~tomita/education/fortran90/sec0.html> by 富田博之. (this is a kind of dictionary.)

- FORTRANが生き残っているのには理由がある。:特に数値計算分野
- <http://www.komoto.org/fortran/>
- <http://members.at.infoseek.co.jp/kitaurawa/fortran.html>

- 標準的なC言語でもよい。 <http://www9.plala.or.jp/sgwr-t/index.html>
- or 速習C言語入門—脳に定着する新メソッドで必ず身につく (単行本) 荻原朋子 (著)

- <http://www.komoto.org/fortran/> より

Fortranを使おう

「いまさらFortran!？」などと言われそうですが、誤解を恐れずに、あえて書いたら、現在普及している高級言語の中でFortranは最も先進的な言語です(ただし数値計算分野に限る)。

簡単にFortranの歴史を書いておきます。

Fortranの生まれは古く、1957年、IBMのJohn Backusにより世界初の高級言語として開発されました。FORmula TRANslatorの名が示すように、Fortranは当初から数値計算を目的としていました。

その後、FORTRAN66、FORTRAN77規格ができ、FORTRAN77は数値計算用言語として多くの研究者に愛用されてきました。そのため、数値計算関連の書籍やライブラリでは、FORTRAN77で書かれた物が数多くあります。

最近、FORTRAN77の問題点を改善し、さらに数値計算特有の機能を強化した Fortran90及びFortran95ができました。ちょっと普及が遅れている気がしますけど。

FORTTRAN77

Fortranと言うと、未だにFORTRAN77の印象が強いようです。しかし、ここではっきりと書いておきます。FORTRAN77は既に時代遅れです。もっとひどく言うなら、「古典だ」と言ってしまっても差し支えないでしょう。

根本的に、構造化プログラミングをしたくてもできない言語仕様に問題があります。具体的に問題点を指摘すると、以下のような点が挙げられると思います。

- * 行番号なしでプログラミングできない
- * ループブロックを記述できない(goto でループを作るしかない)
- * 構造体がない

いいかげん、FORTRAN77はやめて、Fortran90を使いましょう。

Fortran90/Fortran95

Fortran90では、上記のようなFORTRAN77の欠点がほぼ全て解消されています。それに加えて、数値計算用にベクトル化や並列化に適した構文が追加されています。どんな特徴があるのか、C/C++と比較してみましょう。

配列演算を書きやすい

たぶん、他の言語と比べた時の、Fortranの最も大きな特徴でしょう。特に、多次元配列を扱おうとすると、他の言語を使うのがばからしくなります。

並列性を明示的に書くことができる

FortranのForAll文や、配列演算文を使うと、コンパイラは演算に並列性があることを簡単に解析できます。そのため、コンパイラは自動ベクトル化や自動並列化を行いやすくなります。

可読性が高い（Cは特に読みにくいのでコメントを多く：MY註）

"FORmula TRANslator"と言うぐらいですので、元の数式に近い形でプログラムを書けます。また、配列演算構文を使うことで、何重ものループを書いたりする必要が無くなり、ループだらけで何をやっているのかよくわからない、といったことにもなりません。

豊富な数値計算用ライブラリ

数値計算に使うライブラリ群は、大抵FortranとC向けに作られています。NetLib などを見ると、いろいろなライブラリが提供されています。

オブジェクト指向は実装されていない

C++では、オブジェクト指向が使えるようになっています。一方、FortranではFortran2000でオブジェクト指向を実装しようという動きがありますが、まだ標準化されていませんし、対応コンパイラもありません。

オブジェクト指向を使えないのは確かにデメリットですが、そもそも数値計算の場合はオブジェクト指向を使うメリットがあまりないので、あまり気になることは無いように思います。

試しに、 $f(x,y,z) = 2g^2(x,y,z) - 3g(x,y,z) + 4$ を FortranとCで書いてみましょう。

Fortran

```
f(:, :, :) = 2.0*g(:, :, :)**2 - 3.0*g(:, :, :) + 4.0
```

C

```
for(z=0; z<N; z++){  
    for(y=0; y<N; y++){  
        for(x=0; x<N; x++){  
            f[z][y][x]=2.0*g[z][y][x]*g[z][y][x] - 3.0*g[z][y][x] + 4.0;  
        }  
    }  
}
```

Fortranでは、配列演算文をつかって簡単に書けます。このような簡単な式では、Fortranのメリットは感じにくいかもしれませんが、式が複雑になるにしたがって、Fortranの方が書きやすくなります。

また、Fortranの配列演算文では、各配列要素を並列に演算可能であることを示します。上記の例では、コンパイラはN³の並列性があることを簡単に解析できます。

- <http://members.at.infoseek.co.jp/kitaurawa/fortran.html> より

F O R T R A N の 歴 史

1 9 5 0 年 代

F O R T R A N は、米 I B M 社 の John. W. Backus 氏 が 中 心 と な っ て、1 9 5 4 年 か ら 1 9 5 7 年 に け っ け I B M 社 の I B M 7 0 4 計 算 機 に 対 し て 開 発 さ れ た。F O R T R A N と い う 名 は、F O R m u l a T R A N s l a t i n g s y s t e m (数 式 変 換 シ ス テ ム) の 略 称 か ら 付 け ら れ た。世 界 初 の 成 功 し た 高 級 言 語 と 言 わ れ て い る。

当 時 は、高 級 言 語 と い う も の に 関 し て 懐 疑 的 な 見 方 が 強 く、そ の よ う な も の は 非 現 実 的 な も の と 考 え ら れ て い た。そ の 根 拠 と し て、1) コ ン パ イ ラ 自 身 が 貴 重 な 計 算 機 資 源 を 消 費 す る こ と、2) 生 成 さ れ る コ ー ド の 効 率 性 が ア セ ン ブ リ 言 語 に よ る も の ほ ど 期 待 で き な い こ と、な ど が あ げ ら れ て い た。し か し な が ら、F O R T R A N は こ れ ら の 問 題 を 解 決 し、す ば や く 効 率 的 な 実 行 コ ー ド を 生 成 す る こ と に 成 功 し た。こ の 成 功 以 降、高 級 言 語 に 対 す る 懐 疑 は 消 え、幾 多 の 高 級 言 語 が 登 場 す る こ と に な る。盟 友 C O B O L も 1 9 5 0 年 代 末 に は 登 場 し、異 端 児 L I S P も 同 じ 頃 F O R T R A N に よ っ て 書 か れ た。

1 9 6 0 年 代

F O R T R A N は 使 い や す い 言 語 で も あ っ た た め 利 用 者 に 好 ま れ、I B M 社 以 外 の 計 算 機 に も 広 く 普 及 し、各 社 独 自 の F O R T R A N が 乱 立 す る に 至 っ た。そ の 結 果、1 9 6 0 年 代 に 言 語 の 統 一 規 格 の 制 定 が 求 め ら る よ う に な っ た。結 局 I B M 社 の F O R T R A N I V を 元 に、1 9 7 2 年、世 界 初 の 計 算 機 言 語 規 格 F O R T R A N 6 6 が 制 定 さ れ た。こ の 結 果、F O R T R A N の 普 及 は い っ そ う 進 ん だ。

1 9 7 0 年 代

し か し な が ら、こ の こ ろ 計 算 機 学 科 方 面 か ら、F O R T R A N の 言 語 と し て の 欠 陥 に 対 す る 攻 撃 が は じ ま る。1 9 6 8 年 に は、有 名 な 論 文 『G O T O S t a t e m e n t C o n s i d e r e d H a r m f u l (G O T O 文 は 有 害 な り)』が 現 れ、大 論 争 が 巻 き 起 こ る。そ し て 『s t r u c t u r e d p r o g r a m m i n g (構 造 化 プ ロ グ ラ ミ ン グ)』が 一 世 を 風 靡 す る 流 行 語 と な る。ま た、こ れ 以 降 『G O T O 文』を 否 定 し P A S C A L を 礼 賛 す る こ と で 通 人 ぶ る 風 俗 が 登 場 す る。

こ れ に 対 し、F O R T R A N 陣 営 は、G O T O の か わ り に 『C O M E F R O M 文』を 提 案 し て み た り、冗 談 で 対 抗 し た。1 9 8 0 年 に 制 定 さ れ た F O R T R A N 7 7 規 格 は、プ ロ グ ラ ム 制 御 に 関 し て は B L O C K I F 文 す な わ ち 『I F ... T H E N ... E L S E ... E N D I F』と い う 構 造 化 命 令 だ け を 取 り 入 れ た 冗 談 の よ う な 規 格 で あ る。こ れ だ け で は 『G O T O 文』を 排 除 で き な い の で、こ れ 以 降 も 『G O T O 文』を 満 喫 す る 事 に 成 功 す る。

1980年代

ところが、そうこうするうち計算機学科ではPASCAL対Cの内ゲバがはじまっており、FORTRAN陣営が『Real Programmers Don't Use PASCAL (本物のプログラマーはPASCALを使わない)』などとほざいているうちに、いつの間にかFORTRANは相手にされないようになっていた。1981年にはC言語のBrian W. Kernighanが『Why Pascal is Not My Favorite Programming Language (なぜPASCALはわたしの好きなプログラミング言語じゃないのか)』を出す。これ以降C派がクーデターに成功し、世界がUNIX WS (ワークステーション) とCまみれになって行くことが運命づけられる。

しかしながら、FORTRAN陣営は世間の流れにも気づかず、小文字を軽蔑しつつメインフレーム上での楽しい世界にうつつを抜かしていた。さらに1980年代のスーパーコンピュータはベクトル計算機であり、この世に存在するベクトル化コンパイラーはFORTRANだけであったので、何の不安もなく田舎地主のごとく気ままに暮らしていられた。

一方、このころFORTRAN77の中途半端さを憂い、新しい言語規格FORTRAN8x制定に向けての会議が開かれるが、守旧派對革新派、IBM対DEC、その他諸々の対立軸でもめにもめ、ずるずると時は流れ、いつの間にか1980年代は終わってしまう。結局、1991年になって自民党的な素敵な解決策によって新規格FORTRAN90がまとめ上げられることになる。すなわち、革新派にしたがって、構造化プログラミングの必要要素をすべて取り入れると同時に、守旧派にしたがって、これまでのFORTRAN77規格をすべて包括することになった。

1990年代

FORTRANはFORTRAN90となってModule概念を取り入れた構造化言語へと進化し1970年代の言語水準に到達した。FORTRAN90は、理念的だった初期の構造化言語と異なり、様々な実地経験に基づいて規格され、FORTRAN利用者の様々な不満が解決されるように拡張された。言語の抽象度は上がり、概念をコード化するのが以前より容易になった。しかし、守旧派の多いFORTRAN陣営では未だFORTRAN66に固執するものいて、遅々として普及は進まなかった。(但しFORTRAN66は、ほぼFORTRAN77に包括され、FORTRAN77は完全にFORTRAN90に包括されるので、FORTRAN66プログラムは、すなわちほぼFORTRAN90プログラムであるが。)1997年には、FORTRAN90の規格の不備を埋めるBug Fix的規格としてFORTRAN95規格が制定され、より完成度を高めている。

2000年代

FORTRANは、今後もしぶとく生き残ると見られている。その理由は、FORTRAN使用者の間ではプログラマーが即そのプログラムのユーザであることが多い、数学概念との親和性が高い、配列操作に長けている、1950年代のコードが未だに残るほど物持ちが良い、ユーザに守旧派思考が強く他の言語やプログラミング概念を知らないし知ろうともしない、などがあげられている。

現在FORTRANは、オブジェクト指向の概念を取り入れたFORTRAN2000規格を制定中である。

LJ systems

For Ar atoms, $m = 6.63 \times 10^{-26}$ kg, $\epsilon / k_B = 119.8$ K, and $\sigma = 3.405 \times 10^{-10}$ m.

For Ne atoms, $m = 3.35 \times 10^{-26}$ kg, $\epsilon / k_B = 47.0$ K, and $\sigma = 2.72 \times 10^{-10}$ m.

For Kr atoms, $m = 13.92 \times 10^{-26}$ kg, $\epsilon / k_B = 164.0$ K, and $\sigma = 3.83 \times 10^{-10}$ m.

$$k_B = 1.3806503 \times 10^{-23} \text{ J K}^{-1}$$

In the usual LJ systems, one can use the reduced units,
because the system is universal.

In the DL-POLY the following units are used.

1.3.10 Units

Internally all DL_POLY_2 subroutines and functions assume the use of the following defined *molecular units*:

1. The unit of time (t_o) is 1×10^{-12} seconds (i.e. picoseconds).
2. The unit of length (ℓ_o) is 1×10^{-10} metres (i.e. Ångstroms).
3. The unit of mass (m_o) is $1.6605402 \times 10^{-27}$ kilograms (i.e. atomic mass units).
4. The unit of charge (q_o) is $1.60217733 \times 10^{-19}$ coulombs (i.e. unit of proton charge).
5. The unit of energy ($E_o = m_o(\ell_o/t_o)^2$) is $1.6605402 \times 10^{-23}$ Joules (10 J mol^{-1}).
6. The unit of pressure ($\mathcal{P}_o = E_o\ell_o^{-3}$) is 1.6605402×10^7 Pascal (163.882576 atm).
7. Planck's constant (\hbar) which is $6.350780668 \times E_o t_o$.

In addition the following conversion factors are used:

The coulombic conversion factor (γ_o) is:

$$\gamma_o = \frac{1}{E_o} \left[\frac{q_o^2}{4\pi\epsilon_o\ell_o} \right] = 138935.4835$$

such that:

$$U_{MKS} = E_o\gamma_o U_{Internal}$$

Where U represents the configuration energy.

The Boltzmann factor (k_B) is $0.831451115 E_o K^{-1}$, such that:

$$T = E_{kin}/k_B$$

represents the conversion from kinetic energy (in internal units) to temperature (in Kelvin).

Note: In the DL_POLY_2 CONTROL and OUTPUT files, the pressure is given in units of kilo-atmospheres (katm) at all times. The unit of energy is either DL_POLY_2 units specified above, or in other units specified by the user at run time. The default is DL_POLY units.

Lennard-Jones system: phase diagram

相律
 $F = C - P + 2$

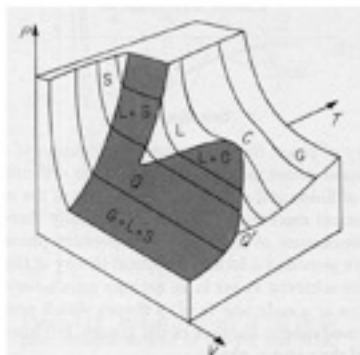


Figure 5.1 PVT surface of a realistic single-component system. The coexistence regions are shown shaded, where G, L and S refer to gas, liquid and solid phases respectively. C and Q are the critical and triple points.

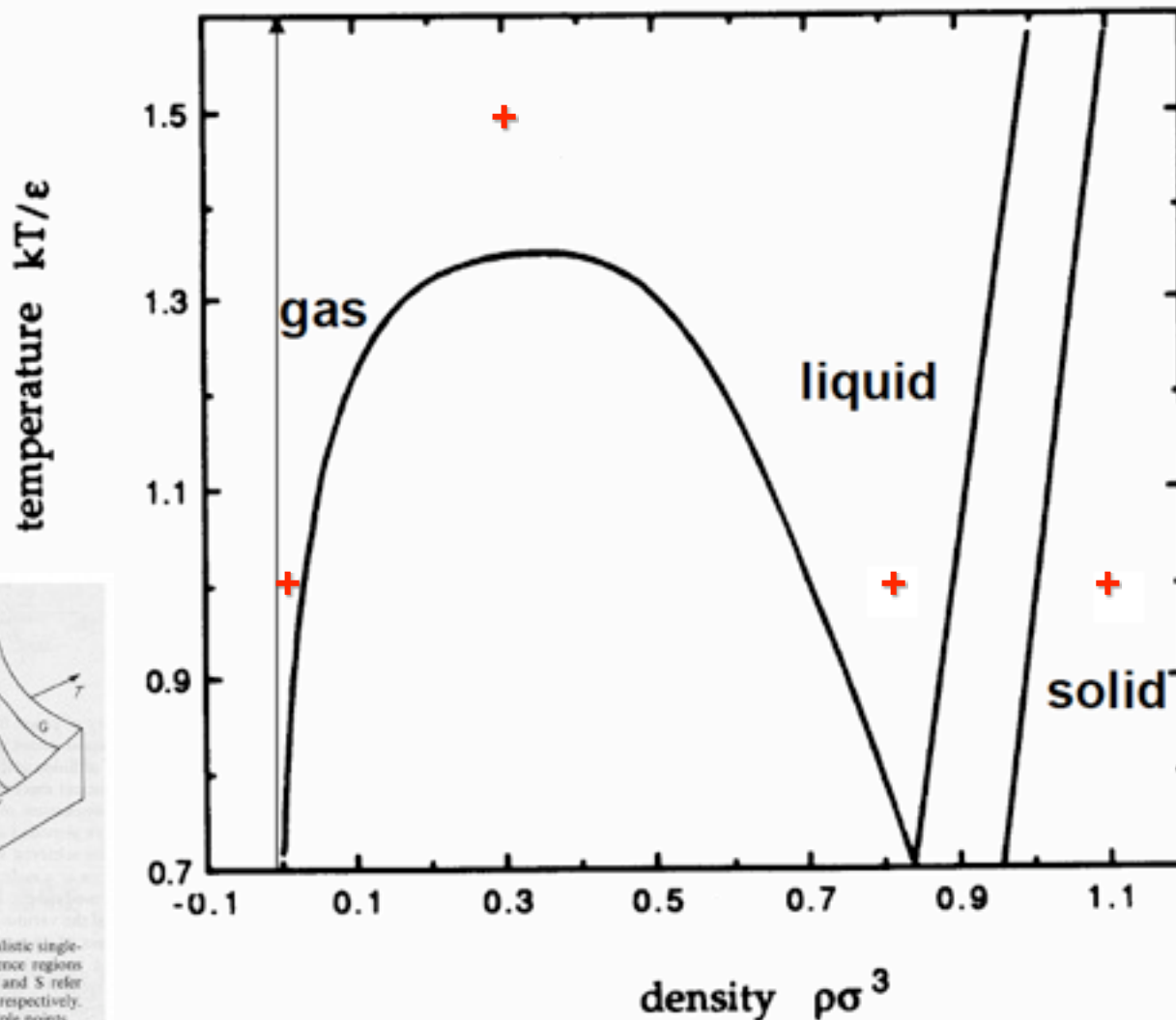
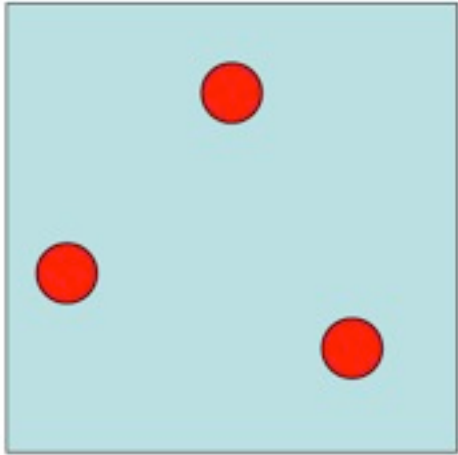


FIGURE 5.7 Temperature–density phase diagram for the pure Lennard-Jones (12, 6) substance. The vapor–liquid saturation curve was computed using the Nicolas et al. [3] equation of state. The melting lines were taken from Hansen and McDonald [10].

Molecular Dynamics Simulation, J. M. Haile, John-Wiley 1992

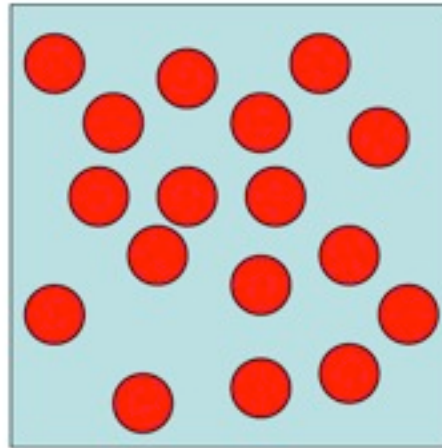
LJ system is universal: two parameters

$$\rho\sigma^3, k_B T/\epsilon$$



gas

$$\rho\sigma^3 = 0.01, k_B T/\epsilon = 1.0$$

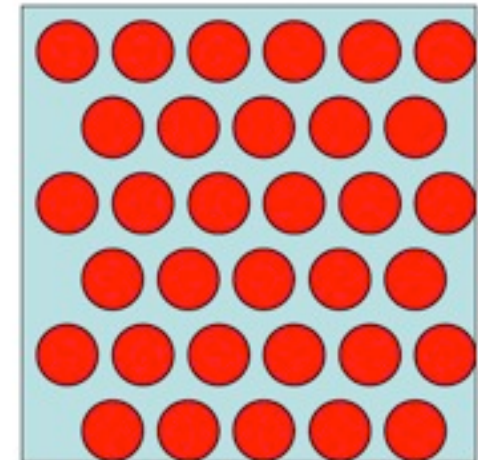


liquid

$$\rho\sigma^3 = 0.8, k_B T/\epsilon = 1.0$$

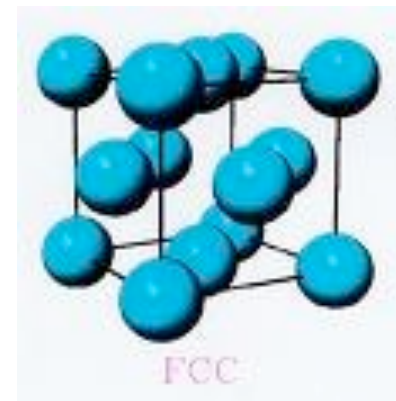
$$\rho\sigma^3 = 1.1, k_B T/\epsilon = 1.0$$

solid



initial positions of Ar atoms

- First we should determine the numbers of atoms in the cell. (e.g. 864 atoms in liquid and solid phase.)
- In gas phase the density is low, you can put the Ar atoms randomly using random number generator.* (*The routine is very important, you should use good one in Wall-Street if you don't want to lose your money.)
- In solid phase it is known that they make a face centered cubic(fcc) lattice. So we should put Ar atoms on the fcc lattice points. It is easy to do that with your own program.
- In liquid phase we can start from the fcc solid.



fcc lattice generation Fortran program by MY (I hope you can understand the language.) Of course you can make it by hand if you like.(laughter)

```

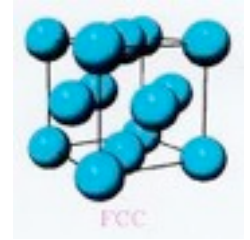
c --- fcc lattice generation Fortran program by MY
c # of atoms:natom
  nx=6
  ny=6
  nz=6
  natom=nx*ny*nz*4
  write (6,*) 'natom=', natom
c --- volume
  sigma=3.405d0
  rho_real= 1.1d0/sigma/sigma/sigma
  volume_real=real(natom)/rho_real
  pow=1.0d0/3.0d0
  box_real=volume_real**pow
  ax=box_real
  ay=box_real
  az=box_real
  volume=ax*ay*az
  write (6,*) 'ax,ay,az,volume=',ax,ay,az,volume
c half of box size
  axh=ax/2.0d0
  ayh=ay/2.0d0
  azh=az/2.0d0
c primitive cubic cell
  box_nx=box_real/real(nx)
  box_ny=box_real/real(ny)
  box_nz=box_real/real(nz)

```

```

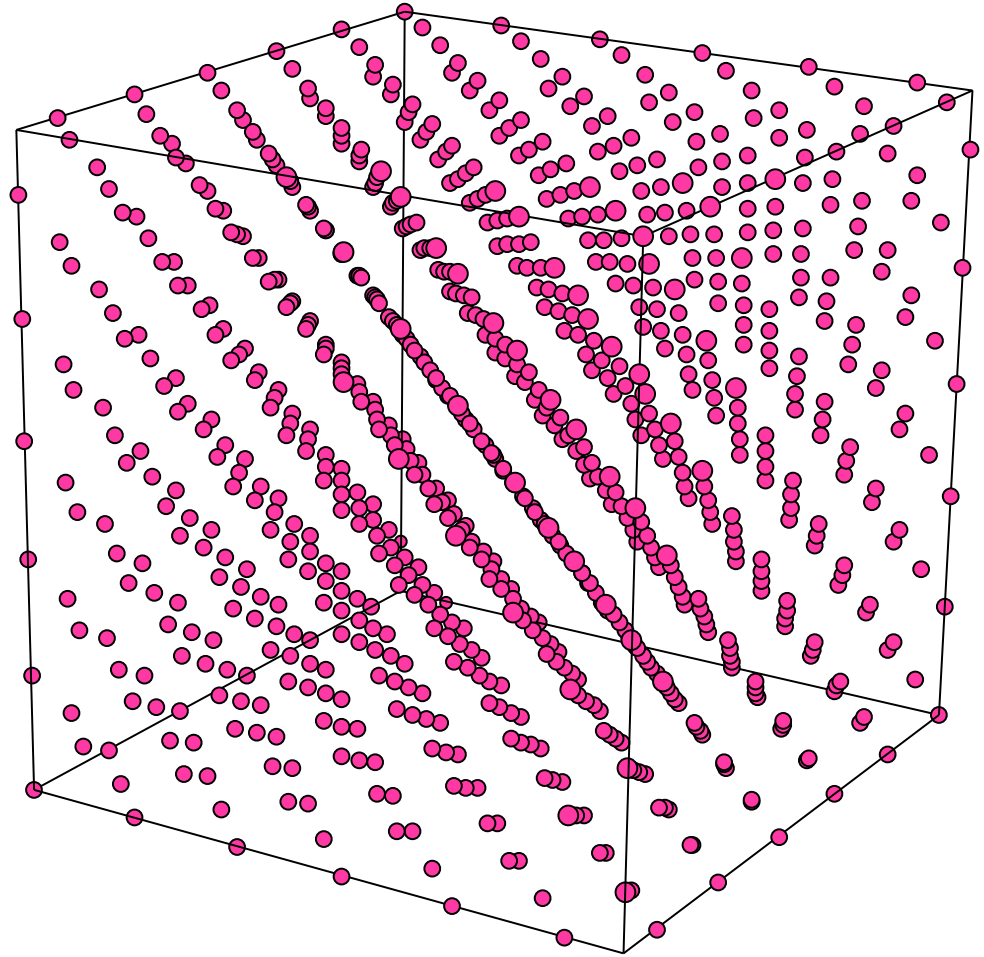
c ---- 4 atoms in cubic cell: fcc
c (0,0,0), (1/2,1/2,0), (1/2,0,1/2), (0,1/2,1/2,1/2)
do i=1, nx
do j=1, ny
do k=1, nz
  icount=icount+1
  x(icount)=real(i-1)
  y(icount)=real(j-1)
  z(icount)=real(k-1)
  write (6,*) icount,x(icount),y(icount),z(icount)
  icount=icount+1
  x(icount)=real(i-1)+0.5d0
  y(icount)=real(j-1)+0.5d0
  z(icount)=real(k-1)
  write (6,*) icount,x(icount),y(icount),z(icount)
  icount=icount+1
  x(icount)=real(i-1)+0.5d0
  y(icount)=real(j-1)
  z(icount)=real(k-1)+0.5d0
  write (6,*) icount,x(icount),y(icount),z(icount)
  icount=icount+1
  x(icount)=real(i-1)
  y(icount)=real(j-1)+0.5d0
  z(icount)=real(k-1)+0.5d0
  write (6,*) icount,x(icount),y(icount),z(icount)
enddo
enddo
enddo
if (icount .ne. natom) stop 999
c ----- fcc lattice ---
c box shift to -axh to axh and so on.
do i=1, natom
  x(i)=x(i)*box_nx-axh
  y(i)=y(i)*box_ny-ayh
  z(i)=z(i)*box_nz-azh
  write (6,'(3f15.7)') x(i),y(i),z(i)
enddo

```



- After the atomic positions are given, you can calculate the forces if you know the inter-atomic potential.
- And if you can integrate the equation of motion numerically, you can calculate the trajectory.
- But we should define the velocity (If we select an option, we do not need velocity and force. But I think it is better to prepare them in this simplest case.)

fcc lattice that
contains 864 atoms



velocity (momentum)



temperature of the system

$$\left\langle \sum_{i=1}^N \frac{m_i v_i^2}{2} \right\rangle = \left\langle \sum_{i=1}^N \frac{p_i^2}{2m_i} \right\rangle = \frac{3Nk_B T}{2}$$

Maxwell-Boltzmann distribution

$$\vec{v}_i = (v_{ix}, v_{iy}, v_{iz})$$

$$\rho(v_{ix}) = \left(\frac{m_i}{2\pi k_B T} \right)^{1/2} \exp \left(-\frac{m_i v_{ix}^2}{2k_B T} \right)$$

v_{iy}, v_{iz} is the same

momentum of the system $\rightarrow 0$

$$\sum_{i=1}^N \vec{p}_i = \sum_{i=1}^N m_i \vec{v}_i = 0$$

```

c ----- MD simulation parameter ----
temp=300.0d0      ! 300 K
bk=1.3806503d-23  !kb J K-1
war= 6.63d-26     !m_Ar kg
c----- initial velocity -----
c      M.B. Distrubution function
c      gasdev is random number generator
c      Normal distribution:mean=0,dispersion=1
c      1/(sigma)/(2pi)**0.5 exp(-(x-<x>)**2/2/sigma**2)
cc=sqrt(bk * temp / war)
sumx=0.0d0
sumy=0.0d0
sumz=0.0d0
do i=1, natom
vxx=gasdev(idum)
vyy=gasdev(idum)
vzz=gasdev(idum)
vx(i)=vxx*cc
vy(i)=vyy*cc
vz(i)=vzz*cc
write (6,*) vx(i),vy(i),vz(i)
sumx=sumx+vx(i)
sumy=sumy+vy(i)
sumz=sumz+vz(i)
enddo
c---- total momentum => zero-----
c      psum is kinetic energy of the system
psum=0.0d0
sumx=sumx/real(natom)
sumy=sumy/real(natom)
sumz=sumz/real(natom)
do i=1, natom
vx(i)=vx(i)-sumx
vy(i)=vy(i)-sumy
vz(i)=vz(i)-sumz
psum=psum+war*(vx(i)**2+vy(i)**2+vz(i)**2)/2.0d0
enddo
tempinst=2.0d0*psum/3.0d0/real(natom)/bk
write (6,*) psum,tempinst

c      write (6,*) itime
write (6,*) psum,tempinst

```

```

c-----
c gasdev is random number generator
c Normal distribution:mean=0,dispersion=1
c-----

```

```

FUNCTION gasdev(idum)
  INTEGER idum
  REAL*8 gasdev
CU  USES ran1
Cran2 use ran2
  INTEGER iset
Cran2 REAL*8 fac,gset,rsq,v1,v2,ran1
  REAL*8 fac,gset,rsq,v1,v2,ran2
  SAVE iset,gset
  DATA iset/0/
  if (iset.eq.0) then
Cran2 I v1=2.*ran1(idum)-1.
Cran2 v2=2.*ran1(idum)-1.
  I v1=2.*ran2(idum)-1.
  v2=2.*ran2(idum)-1.
  rsq=v1**2+v2**2
  if(rsq.ge.1..or.rsq.eq.0.)goto 1
  fac=sqrt(-2.*log(rsq)/rsq)
  gset=v1*fac
  gasdev=v2*fac
  iset=1
  else
  gasdev=gset
  iset=0
  endif
  return
END

```

You can get these codes
from “Numerical recipe” by Press et al.

<http://www.nr.com/>

The above codes are modified by MY.

```

c-----
c ran2 is uniform random number generator
c between 0 and 1.
c ran2 random deviate by L'Ecuyer long period plus shuffle
c-----

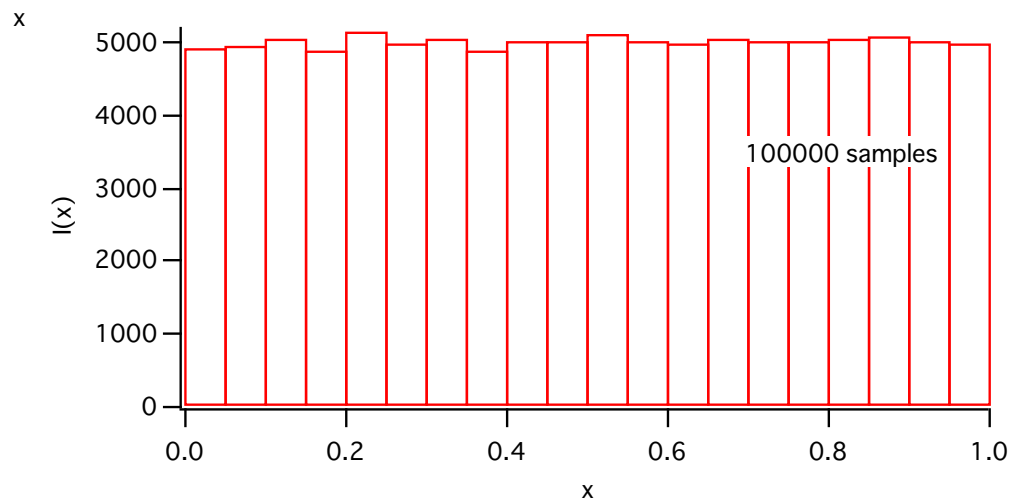
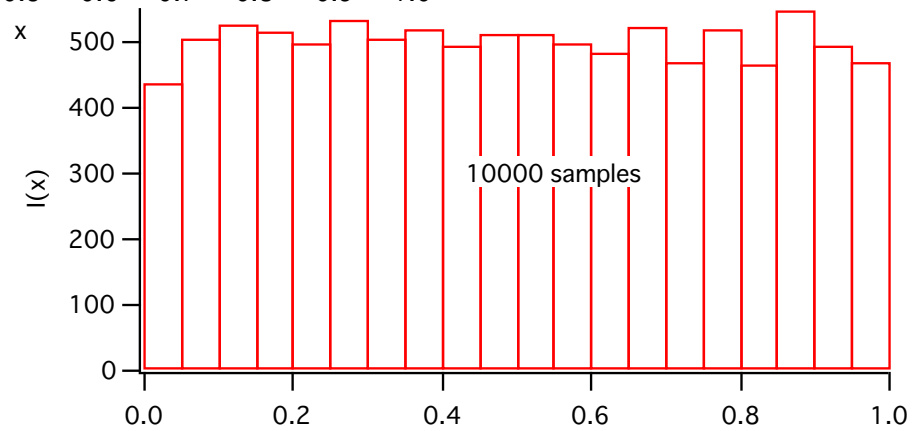
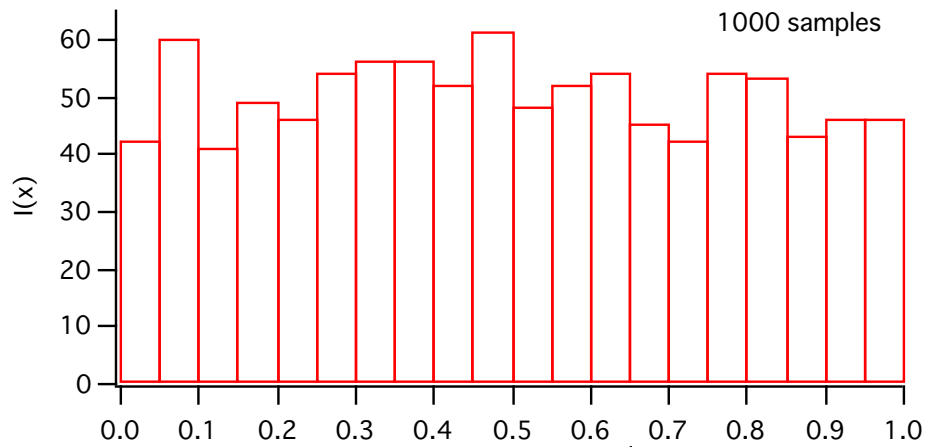
```

```

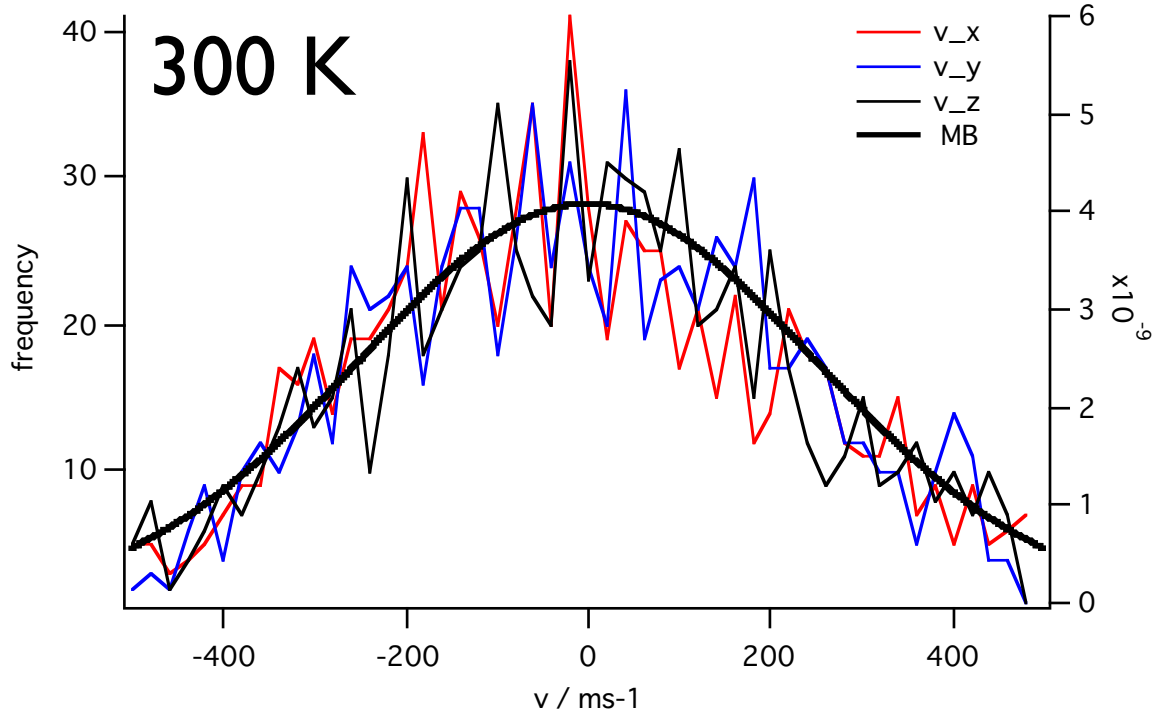
FUNCTION ran2(idum)
  INTEGER idum,IM1,IM2,IMM1,IA1,IA2,IQ1,IQ2,IR1,IR2,NTAB,NDIV
  REAL*8 ran2,AM,EPS,RNMX
  PARAMETER (IM1=2147483563,IM2=2147483399,AM=1./IM1,IMM1=IM1-1,
*IA1=40014,IA2=40692,IQ1=53668,IQ2=52774,IR1=12211,IR2=3791,
*NTAB=32,NDIV=1+IMM1/NTAB,EPS=1.2e-7,RNMX=1.-EPS)
  INTEGER idum2,j,k,iv(NTAB),iy
  SAVE iv,iy,idum2
  DATA idum2/123456789/, iv/NTAB*0/, iy/0/
  if (idum.le.0) then
  idum=max(-idum,1)
  idum2=idum
  do 11 j=NTAB+8,1,-1
  k=idum/IQ1
  idum=IA1*(idum-k*IQ1)-k*IR1
  if (idum.lt.0) idum=idum+IM1
  if (j.le.NTAB) iv(j)=idum
11 continue
  iy=iv(1)
  endif
  k=idum/IQ1
  idum=IA1*(idum-k*IQ1)-k*IR1
  if (idum.lt.0) idum=idum+IM1
  k=idum2/IQ2
  idum2=IA2*(idum2-k*IQ2)-k*IR2
  if (idum2.lt.0) idum2=idum2+IM2
  j=1+iy/NDIV
  iy=iv(j)-idum2
  iv(j)=idum
  if(iy.lt.1)iy=iy+IMM1
  ran2=min(AM*iy,RNMX)
  return
END

```


random number generator check: ran2

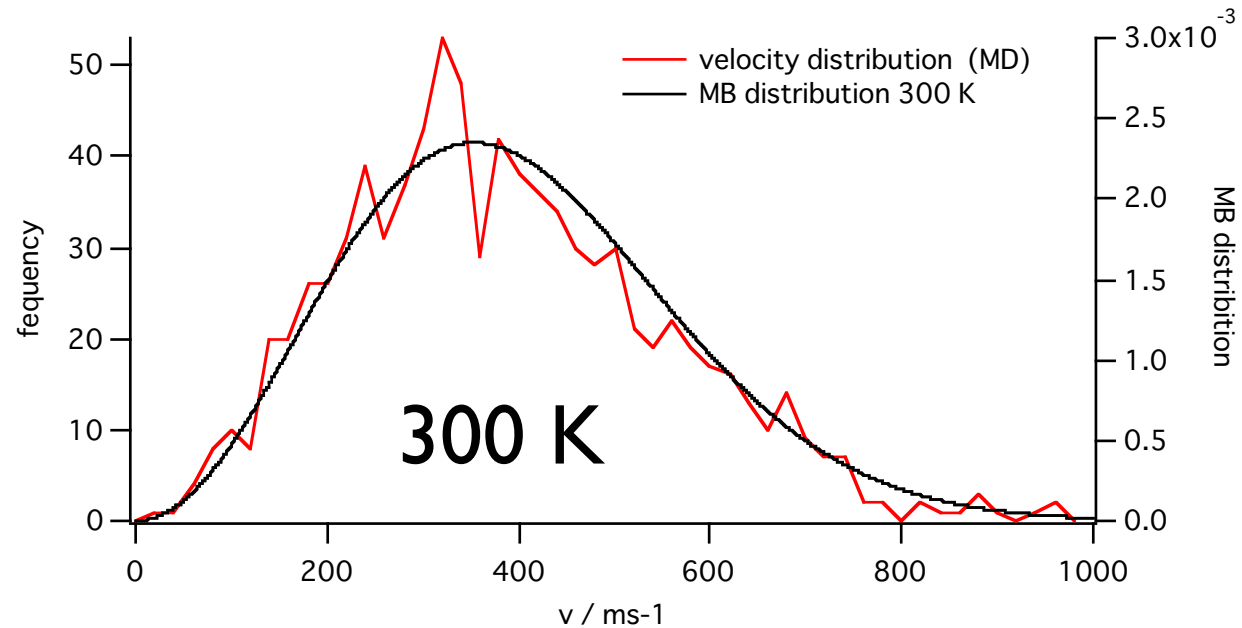


$$\rho(v_{ix}), \rho(v_{iy}), \rho(v_{iz})$$



comparison between the
velocity distribution
from random number
generator and
M-B distribution

$$\rho(\vec{v})dv_xdv_ydv_z \rightarrow 4\pi v^2 \rho(v)dv$$



$$1 \text{ m s}^{-1} = 10^{10} \text{ \AA} \times 10^{-12} \text{ ps}^{-1} = 0.01 \text{ \AA ps}^{-1}$$

Equation of motion: how to solve it

Now we know $\{\vec{r}_i\}$, $\{\vec{v}_i\}$, $\{\vec{F}_i\}$ at time t .

How can we know $\{\vec{r}_i\}$, $\{\vec{v}_i\}$, $\{\vec{F}_i\}$ at time $t + \delta t$.

If the classical trajectory is continuous, the estimate of the positions at $t \pm \delta t$ may be given by

$$\vec{r}_i(t + \delta t) = \vec{r}_i(t) + \delta t \vec{v}_i(t) + \frac{(\delta t)^2}{2} \vec{a}_i(t) + \frac{(\delta t)^3}{6} \vec{b}_i(t) \dots$$

$$\vec{r}_i(t - \delta t) = \vec{r}_i(t) - \delta t \vec{v}_i(t) + \frac{(\delta t)^2}{2} \vec{a}_i(t) - \frac{(\delta t)^3}{6} \vec{b}_i(t) \dots$$

By adding and subtracting, we can get

$$\vec{r}_i(t + \delta t) + \vec{r}_i(t - \delta t) = 2\vec{r}_i(t) + (\delta t)^2 \vec{a}_i(t) + O[(\delta t)^4]$$

$$\vec{r}_i(t + \delta t) - \vec{r}_i(t - \delta t) = 2\delta t \vec{v}_i(t) + \frac{(\delta t)^3}{3} \vec{b}_i(t)$$

Verlet equation (not recommended)

$$\begin{aligned} \vec{r}_i(t + \delta t) &= 2\vec{r}_i(t) - \vec{r}_i(t - \delta t) + (\delta t)^2 \vec{a}_i(t) + O[(\delta t)^4] \\ &= 2\vec{r}_i(t) - \vec{r}_i(t - \delta t) + (\delta t)^2 \frac{\vec{f}_i(t)}{m_i} + O[(\delta t)^4] \end{aligned}$$

The velocities do not appear in the above formula. The velocities are not needed to compute the trajectories, but they are useful for estimating kinetic energy. They may be obtained by

$$\vec{v}_i(t) = \frac{\vec{r}_i(t + \delta t) - \vec{r}_i(t - \delta t)}{2\delta t} + O[(\delta t)^2]$$

Leapfrog (LF) also used in the electromagnetic wave simulation FDTD Yee algorithm

In this method the mid-step velocity and the positions are updated separately. Using the Taylor expansions

$$\vec{v}_i(t - \frac{\delta t}{2}) = \vec{v}_i(t) - \frac{\delta t}{2} \vec{a}_i(t) + \frac{(\delta t)^2}{8} \vec{b}_i(t) + O[(\delta t)^3]$$

$$\vec{v}_i(t + \frac{\delta t}{2}) = \vec{v}_i(t) + \frac{\delta t}{2} \vec{a}_i(t) + \frac{(\delta t)^2}{8} \vec{b}_i(t) - O[(\delta t)^3]$$

$$\vec{v}_i(t + \frac{\delta t}{2}) - \vec{v}_i(t - \frac{\delta t}{2}) = \delta t \vec{a}_i(t) + O[(\delta t)^3]$$

$$\vec{v}_i(t + \frac{\delta t}{2}) + \vec{v}_i(t - \frac{\delta t}{2}) = 2\vec{v}_i(t) + \frac{(\delta t)^2}{4} \vec{b}_i(t)$$

$$\vec{r}_i(t + \delta t) = \vec{r}_i(t) + \delta t \vec{v}_i(t) + \frac{(\delta t)^2}{2} \vec{a}_i(t) + \frac{(\delta t)^3}{6} \vec{b}_i(t) \dots$$

Then we have

$$\vec{\mathbf{v}}_i(t + \frac{\delta t}{2}) = \vec{\mathbf{v}}_i(t - \frac{\delta t}{2}) + \delta t \vec{\mathbf{a}}_i(t) + O[(\delta t)^3] = \vec{\mathbf{v}}_i(t - \frac{\delta t}{2}) + \delta t \frac{\vec{\mathbf{f}}_i(t)}{m_i} + O[(\delta t)^3]$$

$$\vec{\mathbf{r}}_i(t + \delta t) = \vec{\mathbf{r}}_i(t) + \delta t \underbrace{\left(\vec{\mathbf{v}}_i(t) + \frac{\delta t}{2} \vec{\mathbf{a}}_i(t) + \frac{(\delta t)^2}{6} \vec{\mathbf{b}}_i(t) \right)}_{\simeq \vec{\mathbf{v}}_i(t + \frac{\delta t}{2})} + O[(\delta t)^4]$$

$$\simeq \vec{\mathbf{v}}_i(t + \frac{\delta t}{2}) = \vec{\mathbf{v}}_i(t) + \frac{\delta t}{2} \vec{\mathbf{a}}_i(t) + \frac{(\delta t)^2}{8} \vec{\mathbf{b}}_i(t)$$

$$= \vec{\mathbf{r}}_i(t) + \delta t \vec{\mathbf{v}}_i(t + \frac{\delta t}{2}) + O[(\delta t)^4]$$

$$\vec{\mathbf{v}}_i(t) = \frac{\vec{\mathbf{v}}_i(t + \frac{\delta t}{2}) + \vec{\mathbf{v}}_i(t - \frac{\delta t}{2})}{2} + O[(\delta t)^2]$$

The calculation procedure is

$$\left\{ \begin{array}{c} \vec{\mathbf{v}}_i(t - \delta t/2) \\ \vec{\mathbf{f}}_i(t) \end{array} \right\} \Rightarrow \vec{\mathbf{v}}_i(t + \delta t/2), \quad \left\{ \begin{array}{c} \vec{\mathbf{r}}_i(t) \\ \vec{\mathbf{v}}_i(t + \delta/2) \end{array} \right\} \Rightarrow \vec{\mathbf{r}}_i(t + \delta t), \quad \left\{ \begin{array}{c} \vec{\mathbf{v}}_i(t - \delta/2) \\ \vec{\mathbf{v}}_i(t + \delta/2) \end{array} \right\} \Rightarrow \vec{\mathbf{v}}_i(t)$$

Velocity Verlet!!

The velocity Verlet method is recommended because of the memory storage, its numerical stability.

First, the new velocities at mid-step time $t + \delta t/2$ are calculated in the following way.

$$\vec{v}_i(t + \frac{\delta t}{2}) = \vec{v}_i(t) + \frac{\delta t}{2} \vec{a}_i(t) = \vec{v}_i(t) + \frac{\delta t}{2} \frac{\vec{f}_i(t)}{m_i} + O[(\delta t)^2]$$

The new positions are calculated using these velocities at the mid-time-step.

$$\begin{aligned} \vec{r}_i(t + \delta t) &= \vec{r}_i(t) + \delta t \vec{v}_i(t) + \frac{(\delta t)^2}{2} \vec{a}_i(t) + O[(\delta t)^3] \\ &= \vec{r}_i(t) + \delta t \vec{v}_i(t + \frac{\delta t}{2}) \\ &= \vec{r}_i(t) + \delta t \left[\vec{v}_i(t) + \frac{\delta t}{2} \frac{\vec{f}_i(t)}{m_i} \right] + O[(\delta t)^3] \end{aligned}$$

The forces and accelerations ($\vec{f}_i = m_i \vec{a}_i$) at time $t + \delta t$ are then computed from $\vec{r}_i(t + \delta t)$, and the velocity at time $t + \delta t$ is estimated by

$$\begin{aligned} \vec{v}_i(t + \delta t) &= \vec{v}_i(t + \frac{\delta t}{2}) + \frac{\delta t}{2} \vec{a}_i(t + \delta t) \\ &= \vec{v}_i(t + \frac{\delta t}{2}) + \frac{\delta t}{2} \frac{\vec{f}_i(t + \delta t)}{m_i} \\ &= \vec{v}_i(t) + \frac{\delta t}{2m_i} [\vec{f}_i(t + \delta t) + \vec{f}_i(t)] + O[(\delta t)^2] \end{aligned}$$

The calculation procedure is

$$\left\{ \begin{array}{l} \vec{\mathbf{r}}_i(t) \\ \vec{\mathbf{v}}_i(t) \\ \vec{\mathbf{f}}_i(t) \end{array} \right\} \Rightarrow \vec{\mathbf{r}}_i(t + \delta t) \Longrightarrow \vec{\mathbf{f}}_i(t + \delta t) \rightarrow \left\{ \begin{array}{l} \vec{\mathbf{v}}_i(t) \\ \vec{\mathbf{f}}_i(t) \\ \vec{\mathbf{f}}_i(t + \delta) \end{array} \right\} \Rightarrow \vec{\mathbf{v}}_i(t + \delta t)$$

To avoid the round off error , small number amounts should be added separately.

$$\vec{\mathbf{r}}_i(t + n\delta t) = \vec{\mathbf{r}}_i(t) + \delta t \sum_{k=1}^n \vec{\mathbf{v}}_i(t + (k-1)\delta t) + \frac{(\delta t)^2}{2m_i} \sum_{k=1}^n \vec{\mathbf{f}}_i(t + (k-1)\delta t)$$

$$\vec{\mathbf{v}}_i(t + n\delta t) = \vec{\mathbf{v}}_i(t) + \frac{\delta t}{2m_i} \sum_{k=1}^n \left[\vec{\mathbf{f}}_i(t + k\delta t) + \vec{\mathbf{f}}_i(t + (k-1)\delta t) \right]$$

Ar atoms MD simulation input files

- The followings are the examples. microcanonical ensemble NVE
- I think it is very important to “do it yourself”

TABLE I: Translation of reduced units to real units for Lennard-Jones Argon. $m = 6.63 \times 10^{-26}$ kg, $\epsilon/k_B = 119.8$ K, and $\sigma = 3.405 \times 10^{-10}$ m.

Quantity	Reduced units	Real units
length	$\bar{r} = 1$	$\leftrightarrow r = \sigma = 3.405 \times 10^{-10}$ m
energy	$\bar{E} = 1$	$\leftrightarrow E = \epsilon = 1.654 \times 10^{-21}$ J
time	$\bar{t} = 1$	$\leftrightarrow t = \tau = 2.156 \times 10^{-12}$ s
velocity	$\bar{v} = 1$	$\leftrightarrow v = \sigma/\tau = 157.95$ m/s
temperature	$\bar{T} = 1$	$\leftrightarrow T = \epsilon/k_B = 119.8$ K
density	$\bar{\rho} = 1$	$\leftrightarrow \rho = \sigma^{-3} = 2.533 \times 10^{28}$ m ⁻³
pressure	$\bar{P} = 1$	$\leftrightarrow P = \epsilon/\sigma^3 = 41.9$ MPa
force	$\bar{f} = 1$	$\leftrightarrow f = \epsilon/\sigma = 4.858$ pN

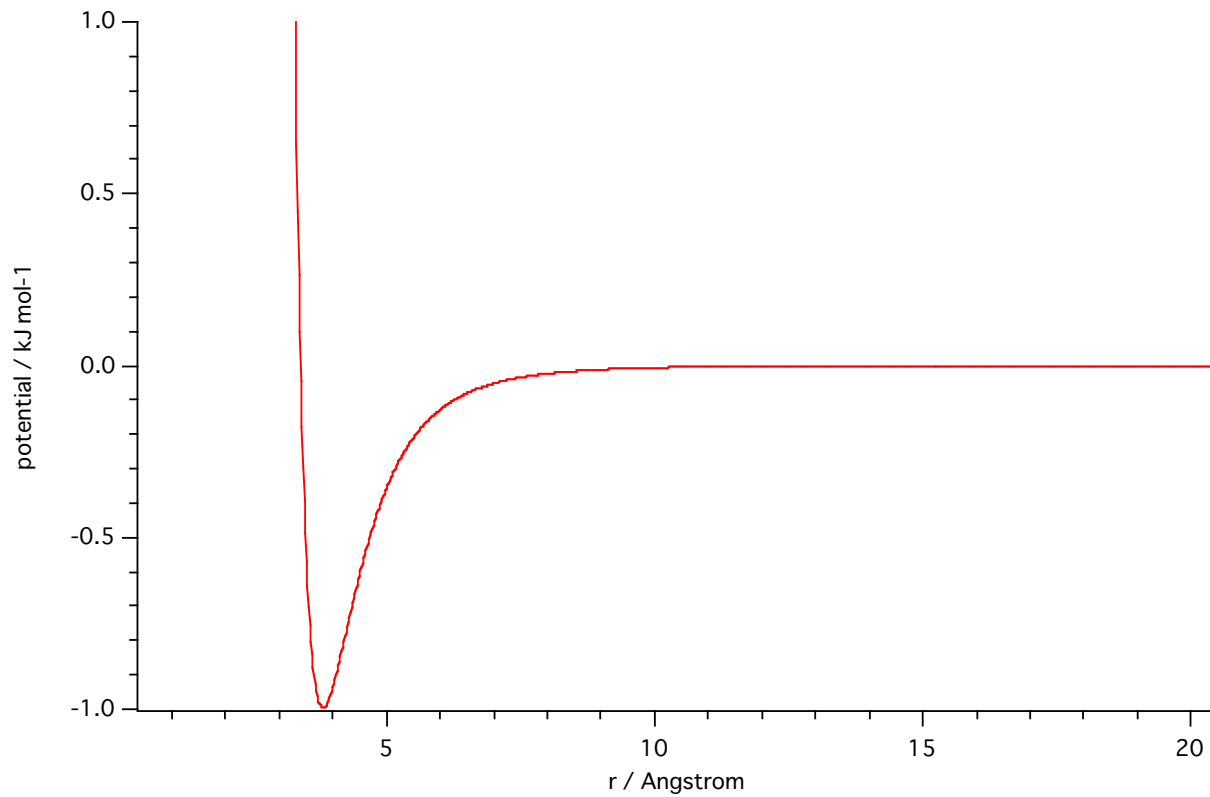
XVI. REDUCED UNITS IN THE L-J POTENTIAL

Now the Lennard-Jones pair potential is described as

$$\phi(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^n - \left(\frac{\sigma}{r} \right)^l \right]$$

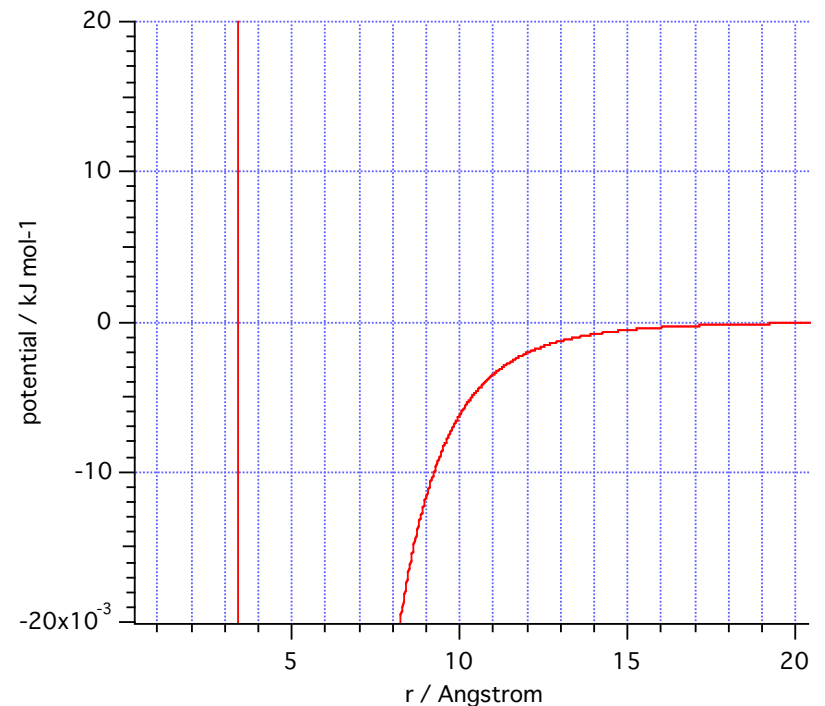
We define the dimensionless \bar{r} and \bar{t} in the following way

$$\begin{aligned} r &= \sigma \bar{r}, & \frac{d}{dr} &= \frac{1}{\sigma} \frac{d}{d\bar{r}} \\ t &= \tau \bar{t}, & \frac{d^2}{dt^2} &= \left(\frac{1}{\tau} \right)^2 \frac{d^2}{d\bar{t}^2} \\ \bar{\phi}(\bar{r}) &\equiv \phi(\bar{r})/\epsilon = 4(\bar{r}^{-n} - \bar{r}^{-l}) \\ \frac{d\bar{\phi}(\bar{r})}{d\bar{r}} &= -4n(\bar{r}^{-n-1} - \frac{l}{n}\bar{r}^{-l-1}) \end{aligned}$$



potential min. /100
 $\Rightarrow r = 10$ Angstrom
 potential min. /1000
 $\Rightarrow r = 15$ Angstrom
 $\Rightarrow r_{\text{cut}}$

rcut = 8 Å is enough



CONTROL

LJ Argon

integrator velocity verlet

temperature 120.00

ensemble nve

steps 5000

equilibration 1000

multiple step 1

scale 10

print 50

stack 100

stats 10

rdf 1

timestep 0.004312

cutoff 8.00

delr width 0.5000

rvdw cutoff 8.00

no electrostatics

shake tolerance 1.0E-5

quaternion tolerance 1.0E-5

print rdf

job time 600.00

close time 100.00

finish

FIELD

```
LJ Ar
units kJ
molecular types 1
Argon
nummols 864
atoms 1
Ar    39.948    0.0    1
finish
vdw   1
  Ar   Ar   lj  0.99607    3.405
close
```

CONFIG

Ar LJ system solid 120K rho1.1

2	1		
31.4164760186	0.0000000000	0.0000000000	
0.0000000000	31.4164760186	0.0000000000	
0.0000000000	0.0000000000	31.4164760186	
Ar 1			
-15.7082380093	-15.7082380093	-15.7082380093	
-1.4841599823	-1.3546851956	0.3078685086	
0.0000000000	0.0000000000	0.0000000000	
Ar 2			
-13.0901983411	-13.0901983411	-15.7082380093	
-1.1985052697	1.1185526954	-0.1253282262	
0.0000000000	0.0000000000	0.0000000000	
Ar 3			
-13.0901983411	-15.7082380093	-13.0901983411	
-0.2902447339	-1.6010119044	-1.5683169323	
0.0000000000	0.0000000000	0.0000000000	
Ar 4			
-15.7082380093	-13.0901983411	-13.0901983411	
0.2025070266	2.6507153672	-1.8006851740	
0.0000000000	0.0000000000	0.0000000000	
Ar 5			
-15.7082380093	-15.7082380093	-10.4721586729	
1.6752475218	3.0135226089	-2.3565592798	
0.0000000000	0.0000000000	0.0000000000	
Ar 6			
-13.0901983411	-13.0901983411	-10.4721586729	
-1.8792819470	0.5181167823	-2.2284424691	
0.0000000000	0.0000000000	0.0000000000	
Ar 7			
-13.0901983411	-15.7082380093	-7.8541190046	
-1.9572185128	-2.5524163519	-1.6063832355	
0.0000000000	0.0000000000	0.0000000000	
Ar 8			
-15.7082380093	-13.0901983411	-7.8541190046	
-3.0249123171	0.5990315258	-1.6698931477	
0.0000000000	0.0000000000	0.0000000000	
Ar 9			
-15.7082380093	-15.7082380093	-5.2360793364	
-1.1606271365	1.9297868123	-0.8672368852	
0.0000000000	0.0000000000	0.0000000000	
Ar 10			
-13.0901983411	-13.0901983411	-5.2360793364	
1.2247598317	1.0607088519	0.3416609074	
0.0000000000	0.0000000000	0.0000000000	

Now we can play
around!!

Exercise

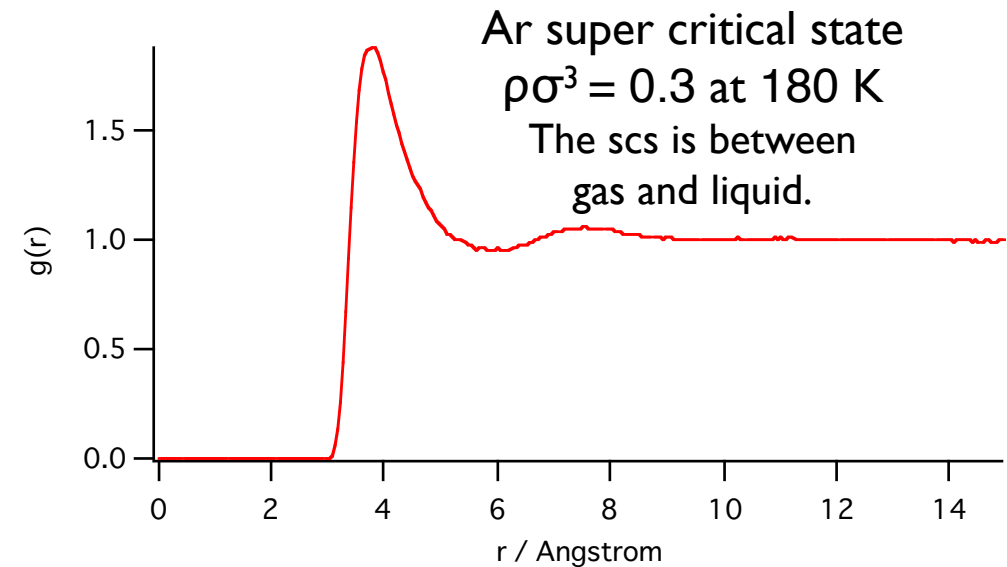
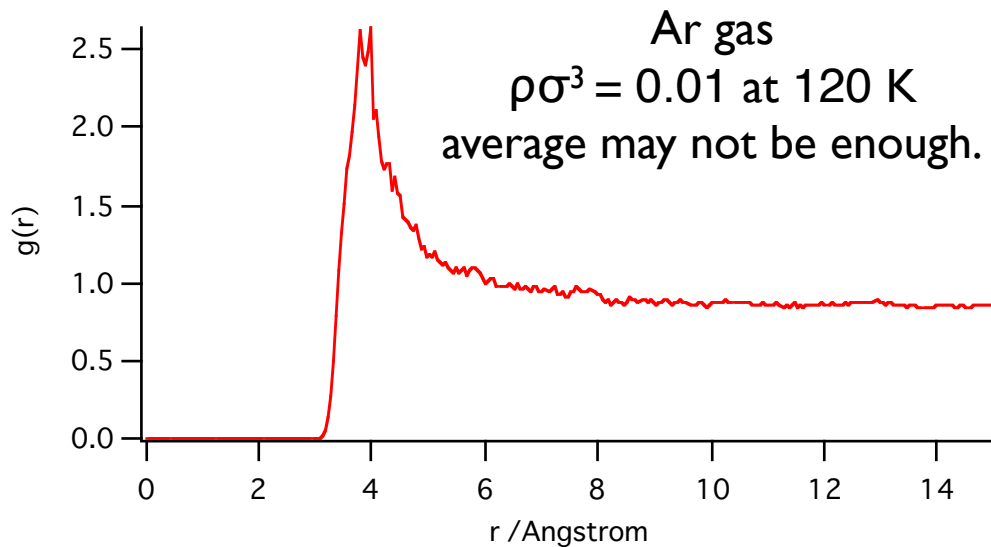
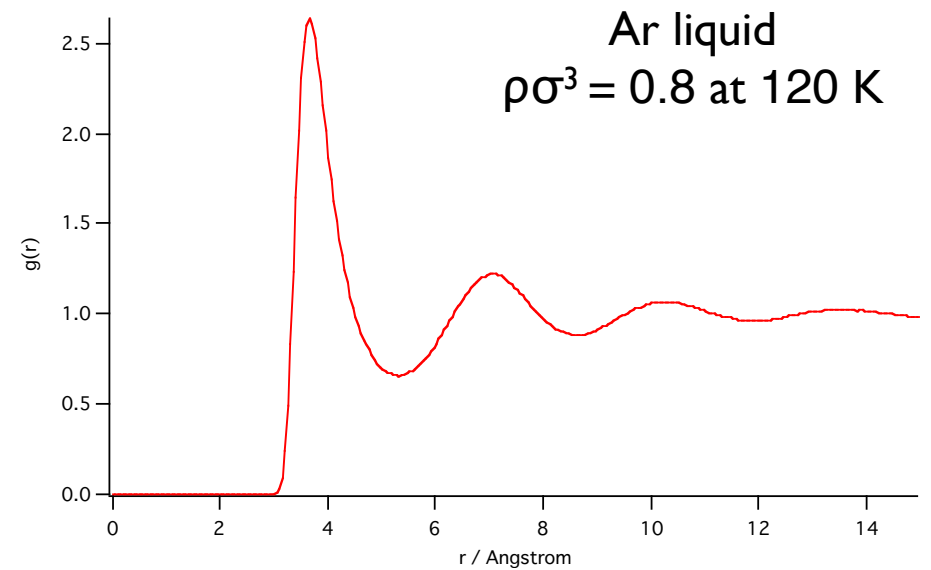
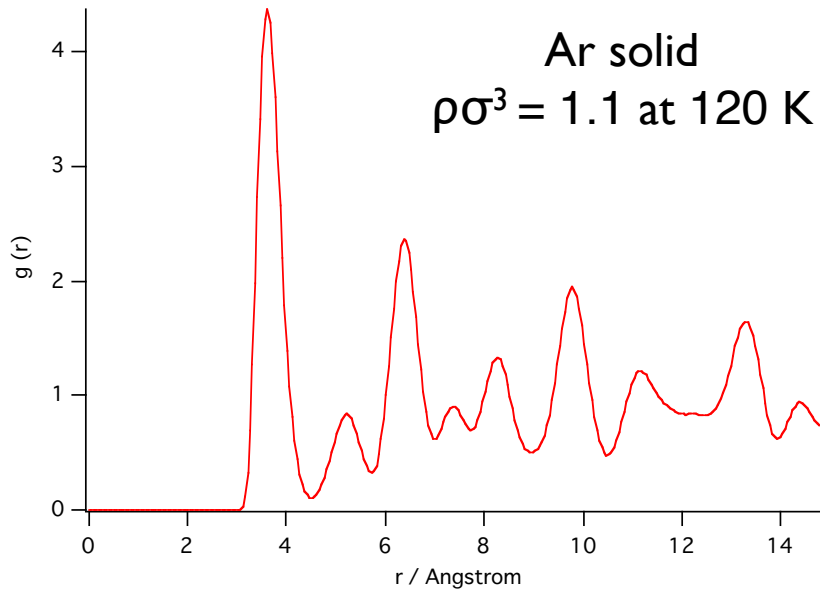
0) The three input files and input-file-make-fortran source codes DLPOLY_Ljinput.f can be downloaded from

http://www.chem.konan-u.ac.jp/applphys/DL_POLY/

From the phase diagram

- 1) DO the MD simulation of Ar solid $\rho\sigma^3 = 1.1$ at 120 K and get radial distribution function (print rdf option in CONTROL file).
- 2) DO the MD simulation of Ar liquid $\rho\sigma^3 = 0.8$ at 120 K and get the radial distribution function.
- 3) DO the MD simulation of Ar gas $\rho\sigma^3 = 0.01$ at 120 K and get the radial distribution function.
- 4) DO the MD simulation of Ar super critical state $\rho\sigma^3 = 0.3$ at 180 K and get the radial distribution function.

Structure



The pair distribution function $g(r)$ gives a probability of finding a pair of atoms a distance r apart, relative to the probability expected for a completely random distribution at the same density.

$$g(r) = \frac{V}{N^2} \left\langle \sum_i \sum_{j \neq i} \delta(\mathbf{r} - \mathbf{r}_{ij}) \right\rangle$$

Dynamics: root of mean square displacement

The mean-square-displacement(MSD) is defined by

$$\langle |\mathbf{r}_i(t + t_0) - \mathbf{r}_i(t_0)|^2 \rangle \equiv \langle |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \rangle$$

The MSD may be independent on the t_0 in the thermal equilibrium condition.

For the short t,

$$\begin{aligned} \mathbf{r}_i(t + t_0) - \mathbf{r}_i(t_0) &\simeq \mathbf{v}(t_0)t \\ \langle |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \rangle &\simeq \langle \mathbf{v}_i^2(0) \rangle t^2 = \frac{3k_B T}{m} t^2 \end{aligned}$$

In the longer time limit, the MSD becomes

$$\langle |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \rangle \simeq 6Dt$$

In the NVE-MD calculation the center of the mass of the system is fixed, so each particles do not diffuse freely. By this constraint the MSD deviates from the above linear relation at the longer t limit.

The diffusion constant is also given by

$$D = \frac{1}{3} \int_0^\infty dt \langle \mathbf{v}_i(t) \cdot \mathbf{v}_i(0) \rangle$$

In practice these averages would be computed for each of the N particles in the simulation, the results added together, and divided by N . Note that in the computation of MSD, it is important to use the trajectory which have not been subjected to PBC.

In the OUTPUT you can find,
the diffusion constant is given.

Be careful that MSD should be
linear with time!! In these
case no checks were done!!

Ar solid $\rho\sigma^3 = 1.1$ at 120 K

Approximate 3D Diffusion coefficients ($10^{-9} \text{ m}^2 / \text{ s}$)
0.0000E+00

Ar liquid $\rho\sigma^3 = 0.8$ at 120 K

Approximate 3D Diffusion coefficients ($10^{-9} \text{ m}^2 / \text{ s}$)
4.0951E+00

Ar gas $\rho\sigma^3 = 0.01$ at 120 K

Approximate 3D Diffusion coefficients ($10^{-9} \text{ m}^2 / \text{ s}$)
4.2605E+02

Ar supercritical state $\rho\sigma^3 = 0.3$ at 180 K

Approximate 3D Diffusion coefficients ($10^{-9} \text{ m}^2 / \text{ s}$)
3.1635E+01

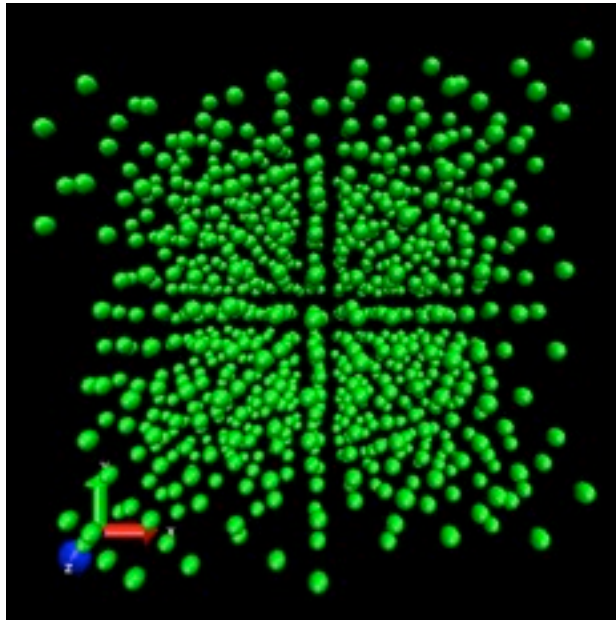
structure and dynamics

to see the **Movie**: I like the software such as
Jmol <http://jmol.sourceforge.net>

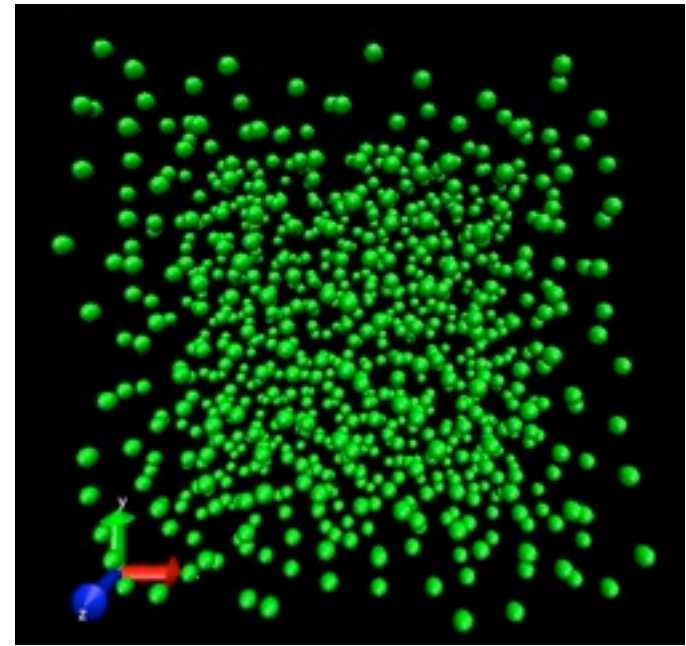
and/or

VMD <http://www.ks.uiuc.edu/Research/vmd/>

and If we use VMD, it can read HISTORY file
directory.

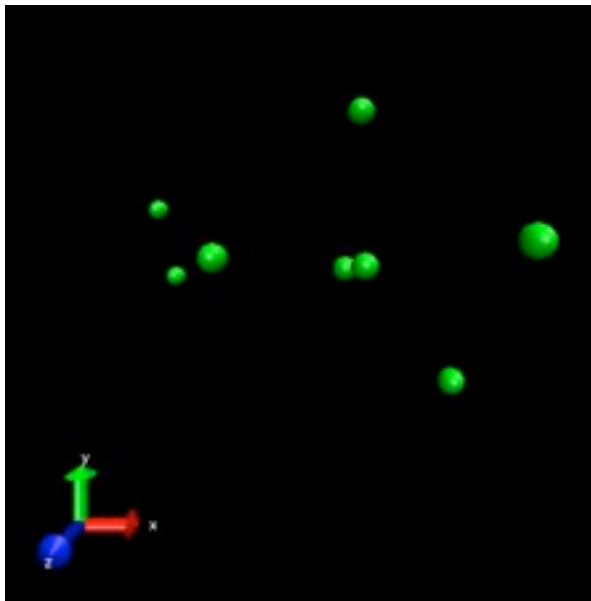


Solid $\rho\sigma^3 = 1.1$ at 120 K



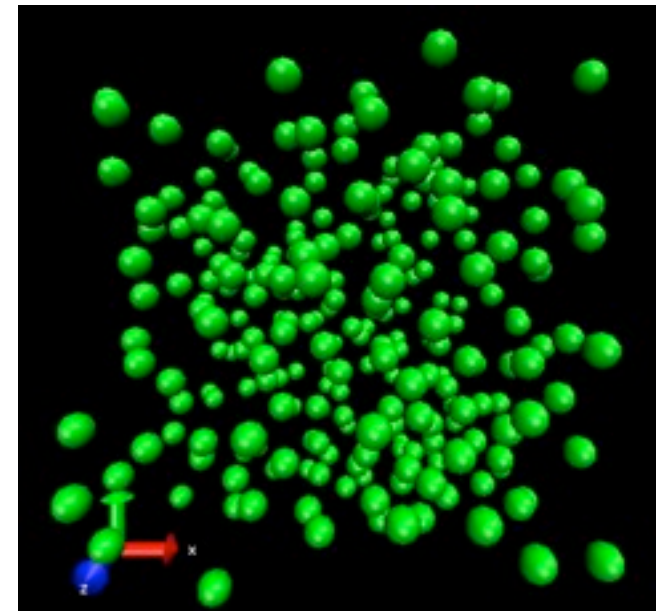
liquid $\rho\sigma^3 = 0.8$ at 120 K

Snapshot



Gas. $\rho\sigma^3 = 0.01$ at 120 K

It is like “Hotaru” firefly



supercritical state $\rho\sigma^3 = 0.3$ at 180 K

Animations (so what's?)

VMD (visual MD) is used here.

HISTORY file can be easily visualised.

Please visit

<http://www.ks.uiuc.edu/Research/vmd/>

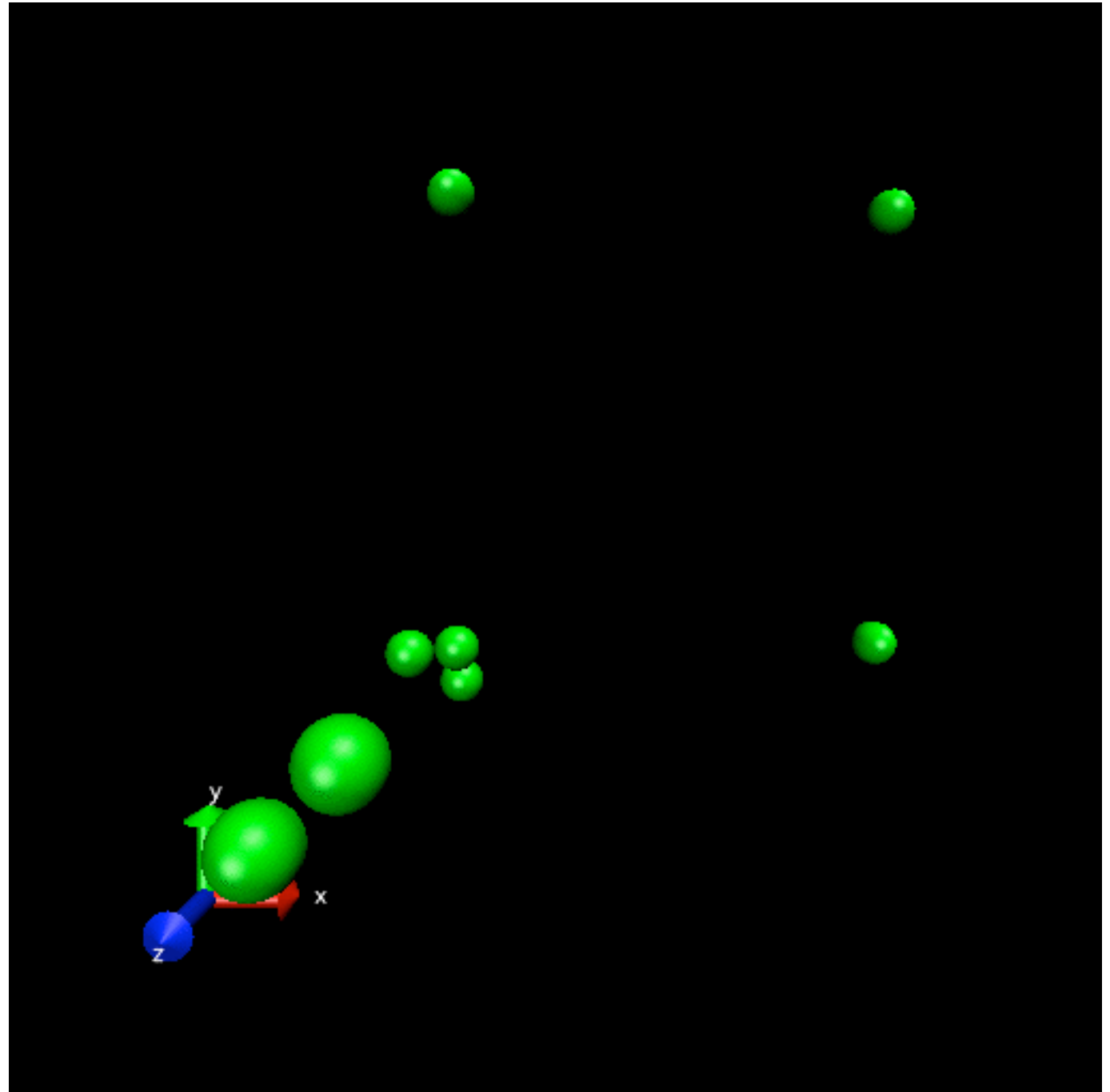
and can export animated GIF by imagemagik!

animation

Gas. $\rho\sigma^3 = 0.01$ at 120 K
It is like “Hotaru” firefly

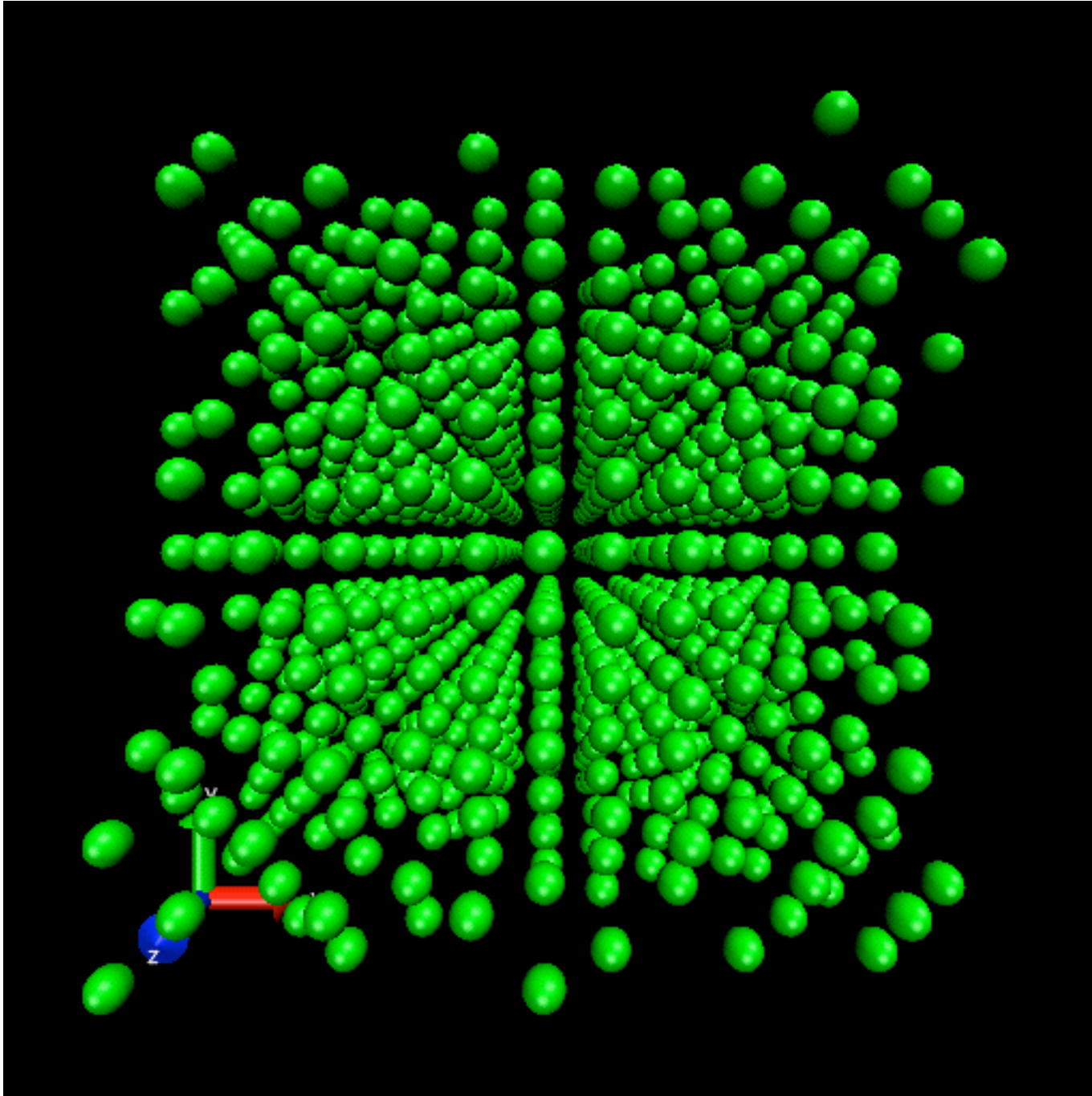
animation

Gas. $\rho\sigma^3 = 0.01$ at 120 K
It is like “Hotaru” firefly



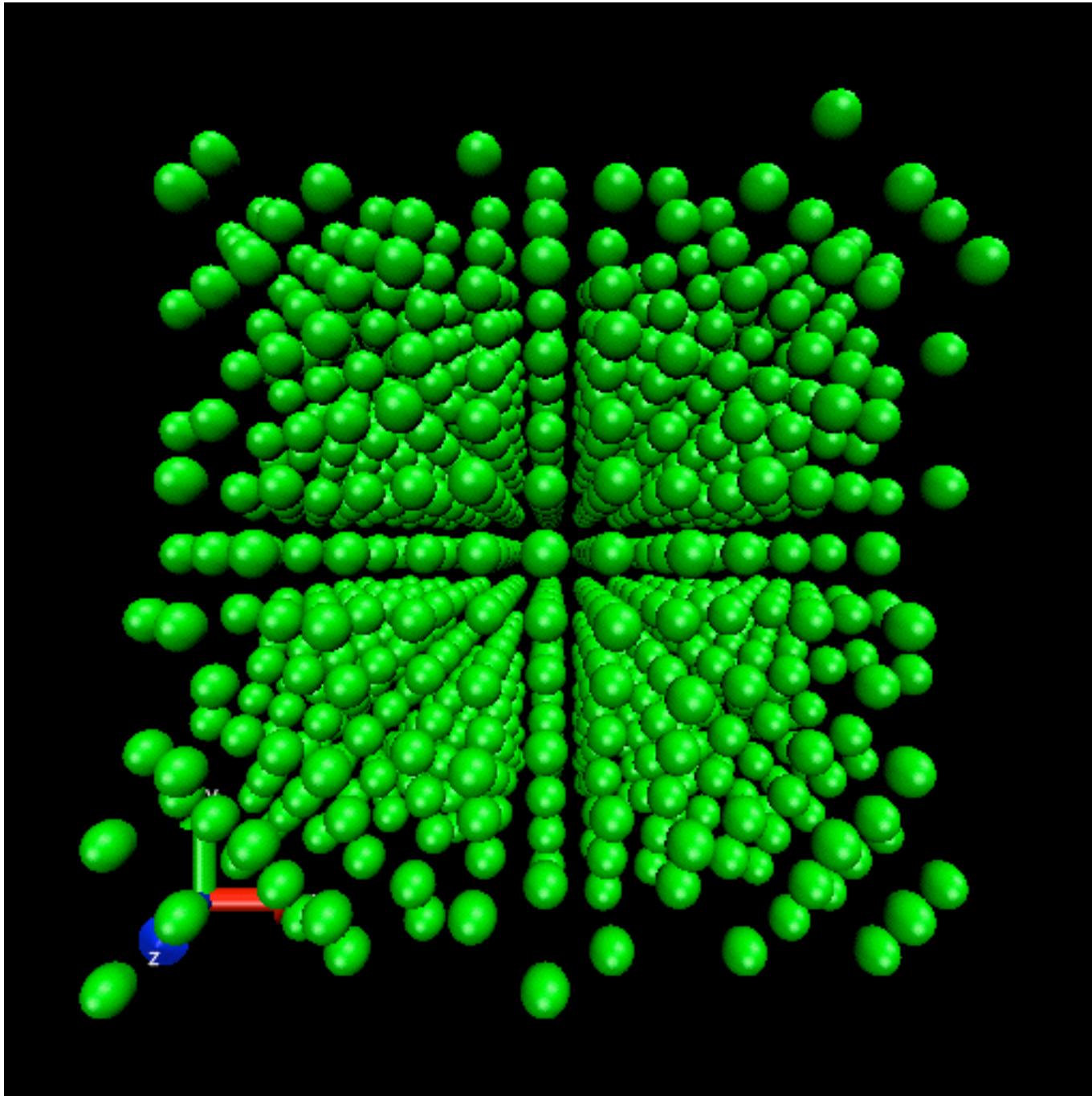
liquid $\rho\sigma^3 = 0.8$ at 120 K

liquid $\rho\sigma^3 = 0.8$ at 120 K



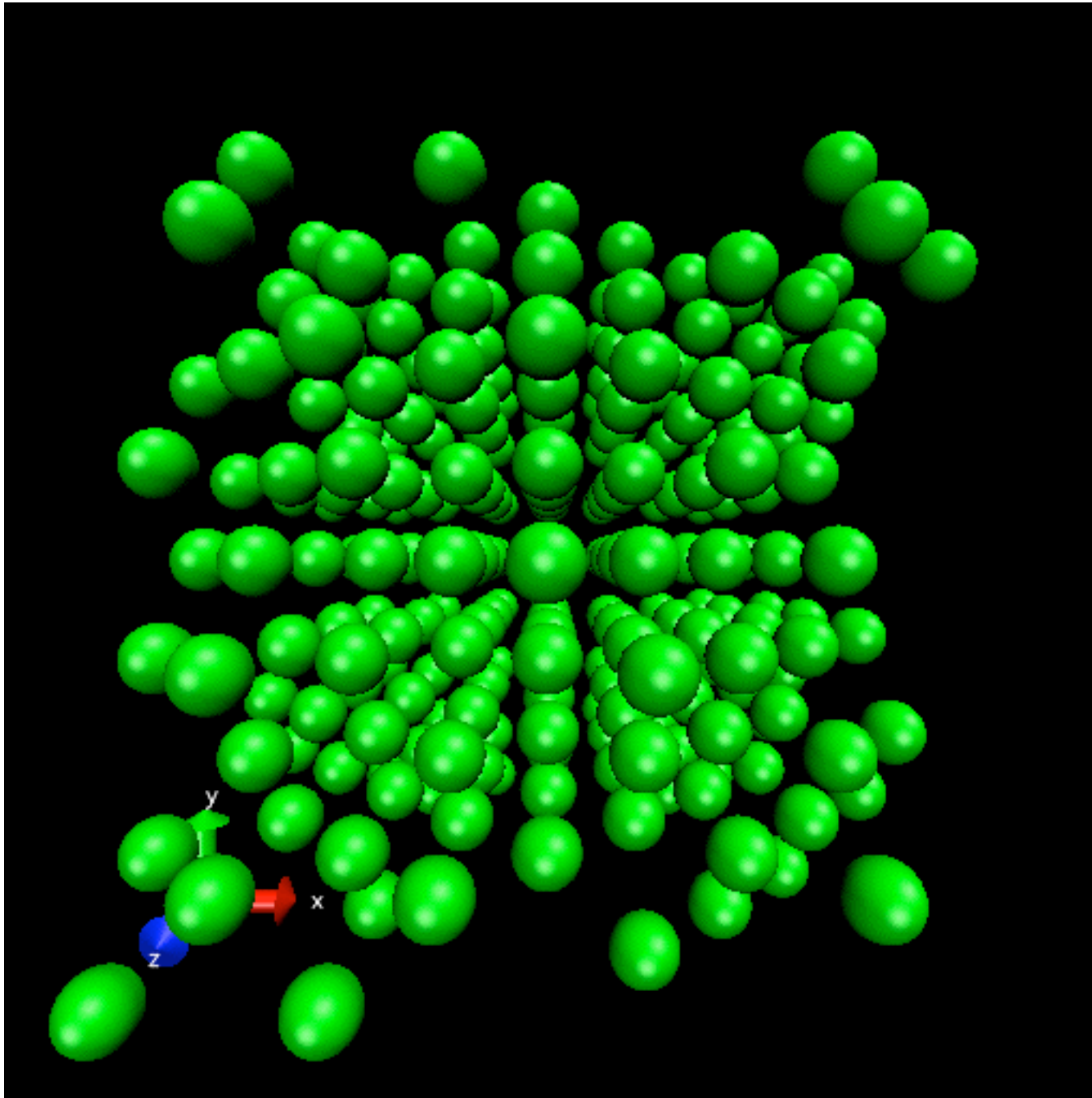
Solid $\rho\sigma^3 = 1.1$ at 120 K

Solid $\rho\sigma^3 = 1.1$ at 120 K



supercritical state $\rho\sigma^3 = 0.3$ at 180 K

supercritical state $\rho\sigma^3 = 0.3$ at 180 K



This is the end.

I think it takes time to make a input files if you make by your-self even in this simple system.

- In the next tutorial I will try to include long-range coulomb interaction. I will do a high-temperature molten salts (like LiCl and/or NaCl) MD simulatiuon.