The Influence of Numerical Calculation Methods on the Molecular Dynamics Simulation Results

Dong-yu Xia

Yanshan University，Hebei，066004，China

E-mail：s8055798@163.com

Abstract. In this paper, three kinds of algorithms usually used in molecular simulation are introduced in detail. In order to compare the results of the fluid transport coefficients among different algorithms, uniform fluid 3-D model calculated. With the aim of comparing the temperature adaptability, crystal copper nano-bar model is simulated with MD method. It is very helpful for the choice of different calculating aim.

1. Introduction

As an auxiliary method of molecular behavior research, molecular dynamics simulation has been widely used in nanoscale fluid and solid model research [1][2][3]. The MDS method is based on the Newton's law of motion, approximates the differential equation by the finite difference method, and makes the motion of the particle transitions from the description of continuous variables and differential operators to the description of discrete variables and finite difference operators. For numerical solution on the computer, there are many finite difference algorithms that can solve Newton's equations of motion, and they are given in detail in many literatures [4][5][6]. Molecular dynamics requires high accuracy in time and space, and it is necessary to select an appropriate time integration algorithm according to the purpose of calculation.

Verlet L. presented the famous Verlet algorithm [6] as early as 1967. This algorithm is based on the Taylor expansion and is widely used as a method for calculating the particle displacement, velocity, and acceleration in the simulation. The displacement truncation error of the basic form of the Verlet algorithm is $O(\Delta t^4)$, and the truncation error of the velocity calculation is $O(\Delta t^2)$. The algorithm requires that when calculating the velocity at time t, the position at $t-\Delta t$ and $t+\Delta t$ must be known in advance. That is, the calculation of velocity at t is performed after the position at $t+\Delta t$ is determined. This results in velocity and displacement calculation are not synchronized. When energy calculation is performed, kinetic energy and potential energy cannot be calculated at the same time, which increases the occupation of memory storage space. For a constant temperature system, when a speed scale is used for temperature calibration, since the displacement calculation is not limited by the speed, the temperature correction cannot affect the displacement calculation result, this causes the speed calculation to be decoupled from the displacement calculation, it is not suitable with Verlet algorithm for direct velocity scaling method.

Then in 1982, Velocity-Verlet algorithm [7] was proposed as a further improvement of the Verlet algorithm by Swope W. C. et al as follows:

$$\mathrm{r}(t+\Delta t)=\mathrm{r}(t)+\mathrm{v}(t)+\frac{f(t)}{2m}\Delta t^2 \tag{1}$$

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

$$\mathrm{v}\left(t+\Delta t\right)=\mathrm{v}\left(t\right)+\frac{f\left(t+\Delta t\right)+f\left(t\right)}{2m}\Delta t \tag{2}$$

By formula (1) (2), we can see that compared with the Verlet algorithm, the advantage of this method is that it can obtain displacement, velocity and acceleration at the same time, and there is an explicit speed item. On the premise that the displacement truncation error is not changed, the velocity truncation error is reduced to O (Δt^4), and the displacement and velocity calculation results can be obtained at the same time. The stored variables are the same as the Verlet algorithm, 9N variable storage spaces are needed.

Another evolutionary algorithm form was proposed based on the Verlet algorithm by Beeman [8] in 1976. Although the displacement truncation error is O (Δt^4), the speed truncation error reduces to O (Δt^3), but the storage space required for its calculation increases to 12N.

$$\vec{r}\,(t+\Delta t)=\vec{r}\,(t)+\Delta t\,\vec{V}\,(t)+\Delta t^2\,\frac{4\,\vec{a}\,(t)-\vec{a}\,(t-\Delta t)}{6} \tag{3}$$

$$\vec{V}\,(t+\Delta t)=\vec{V}\,(t)+\Delta t\,\frac{2\,\vec{a}\,(t+\Delta t)+5\,\vec{a}\,(t)-\vec{a}\,(t-\Delta t)}{6} \tag{4}$$

In 1971, Gear proposed the prediction-correction integration method [9]. The algorithm can select different highest order micro-scores, which is correction factor orders, according to different calculation requirements, so that the truncation error progression can be appropriately adjusted.

$$r^p(t+\Delta t)=r(t)+v(t)\Delta t+\frac{1}{2}a(t)\Delta t^2+\frac{1}{6}\Delta t^3 b(t)+\ldots \tag{5}$$

$$v^p(t+\Delta t)=v(t)+a(t)\Delta t+\frac{1}{2}b(t)\Delta t^2+\ldots \tag{6}$$

$$a^p(t+\Delta t)=a(t)+b(t)\Delta t+\ldots \tag{7}$$

A set of vectors is defined by the above formula, where r_0 is the position of the atom:

$$r_1=(\frac{dr_0}{dt})\Delta t=v_0\Delta t \tag{8}$$

$$r_2=\frac{1}{2}(\frac{d^2r_0}{dt^2})\Delta t^2=\frac{1}{2}a_0\Delta t^2 \tag{9}$$

$$r_3=\frac{1}{6}(\frac{d^3r_0}{dt^3})\Delta t^3=\frac{1}{6}b_0\Delta t^3 \tag{10}$$

The vector above satisfies the following relationship

$$\begin{pmatrix} r_0^p(t+\Delta t)\\ r_1^p(t+\Delta t)\\ r_2^p(t+\Delta t)\\ r_3^p(t+\Delta t)\end{pmatrix}=\begin{pmatrix}1&1&1&1\\0&1&2&3\\0&0&1&3\\0&0&0&1\end{pmatrix}\begin{pmatrix} r_0(t)\\ r_1(t)\\ r_2(t)\\ r_3(t)\end{pmatrix} \tag{11}$$

For Gear algorithm, the correction factor is predicted as:

$$\begin{pmatrix} r_0^c(t+\Delta t) \\ r_1^c(t+\Delta t) \\ r_2^c(t+\Delta t) \\ r_3^c(t+\Delta t) \end{pmatrix} = \begin{pmatrix} r_0^p(t+\Delta t) \\ r_1^p(t+\Delta t) \\ r_2^p(t+\Delta t) \\ r_3^p(t+\Delta t) \end{pmatrix} + \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{pmatrix} \Delta r \tag{12}$$

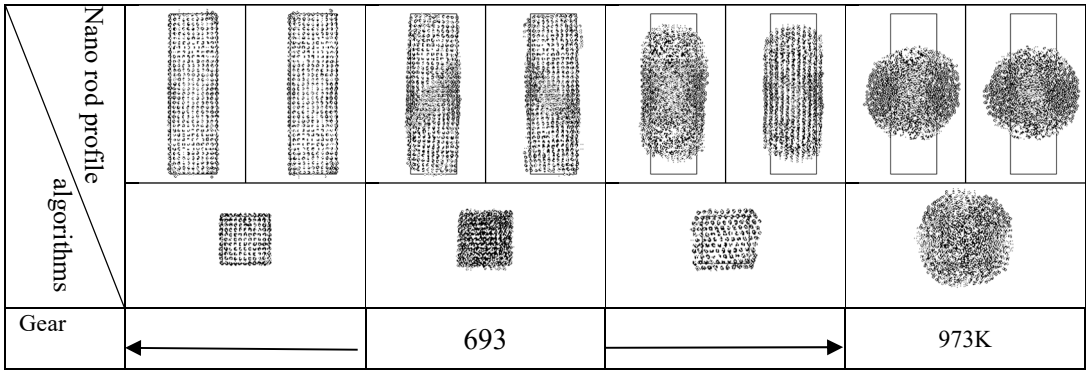
For second-order equations of motion, $\ddot{r} = f(r, \dot{r})$, $\Delta r = r_2^c - r_2^p$. For the example in this paper 4th-order correction factor coefficients are used, and the storage space for calculation variables is also 12N. The calculation method can be divided into three steps: first, according to the Taylor expansion, the new position r^p, velocity v^p, and acceleration a^p are predicted, and then calculate the force and acceleration $a(t+\Delta t)$ according to the new position. This acceleration is then compared with the acceleration a^p which is predicted by the Taylor's expansion, and finally the difference between the two is used to correct the position and velocity terms during the correction process.

In principle, an algorithm should be able to predict all particle trajectories at the same time whether for short period or for a long period of time. In theory, the Newton's equation of motion is time-invertible, but due to the approximation of the numerical calculation method, the system error and truncation error exist. It is not time reversible, that is, if the momentum of all particles is reversed at a certain moment, the system cannot be inverted in the phase space. The numerical calculation method should reduce the calculation error as much as possible. In addition, particle distribution, the stability of the system during calculation, the trend of the kinetic energy potential energy curve, the simplicity of the calculation method, the amount of memory used and the cycle time, etc., All of these have become factors influencing the choice of numerical methods in the simulation process. In this paper different calculation models will be used to compare different numerical calculation methods and provide reference for the correct choice of calculation methods in different situations.

2. Influence of Integral Algorithm on Temperature Stability of Solid Model

Nanoscale copper rod model is used to research the model's changing with temperature under different integration methods, aiming to discuss the applicability of different numerical integration methods to temperature. The copper rod atoms are arranged according to the FCC ideal lattice, and the nanorod size is selected as 5a×5a×17a (1.805 nm×1.805 nm×6.137 nm), so the number of atoms is 2118. The surface of the model is defined as a free surface boundary, and the temperature is controlled by the NTV ensemble velocity scale, time step is set as 1.0E-15 seconds. The initial velocity of the particles in the model is subject to the Maxwell-Boltzmann distribution of the temperature, and the interatomic force is EAM[10] potential which is suitable for the multi-body potential mosaic function, and for simulating the mechanical behavior of metals. With relaxing 100,000 steps, the comparison results are shown in Table 1.

Table 1. The relationship between control temperature and the profile of crystal copper nanorods model using MDS method with different algorithms。



Beeman	←	673	→	983K
Velocity-Velert	←	333	→	1003K

Note: The upper side is the left view and the right view respectively, and the lower side is the top view

With velocity-Verlet algorithm, when the control temperature is lower than 333K, the copper rod maintains the original geometric shape, and when the temperature is controlled at 333K, a 45-degree-dark diagonal region appears along the length of the copper rod. With further increase in temperature, the copper rod shrinks to a spherical shape at 1003K. The copper rod of the Beeman algorithm keeps the original shape of the holding temperature up to 673K. When the control temperature is higher than 673K, the copper rod shrinks. The original shape of the Gear algorithm keeps the temperature up to 693K, but the subsequent contraction speed is larger than that of the Beeman algorithm, it has shrunk to a nearly spherical shape at 973K, and the result of the Beeman algorithm is still in the process of continuous contraction. By comparing the temperature control of nanometer copper rods, it can be seen that when using the EAM potential function for molecular dynamics simulation of single crystal copper nanorods, the original shape of the Gear algorithm maintains a higher temperature of 693 K. When the Velocity-Verlet algorithm is used, its temperature stability is lower than that of the other two algorithms, and its original shape maintaining temperature is only 333K.

3. The Applicability of Different Algorithms for Uniform Fluid Transport Parameters

Liquid argon is used as the fluid model medium, and the influence of the different algorithms method on the results of the fluid transport coefficient is discussed using a homogeneous fluid model under periodic boundary conditions. There is no obvious interface between the model and the surrounding molecules. The molecular arrangement still adopts the FCC crystal structure. The fluid density is 1.413 g/cm^3, with the NVT ensemble, and the control temperature is 91.1K. The intermolecular interaction potential is the commonly used LJ potential [11]. The model size is set to 10×10×10 FCC and the number of atoms is 4000. The transport-coefficient calculation method uses the Green-Kubo formula under the time autocorrelation theory [12].

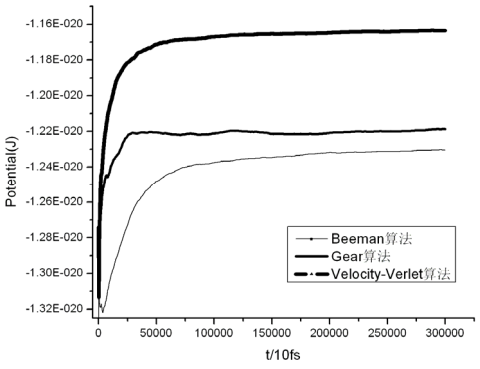


Figure 1. Average potential-time curve of uniform fluid 3-d model with various methods

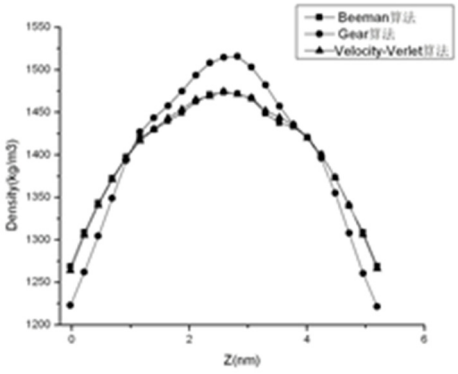


Figure 2. Density distribution curve of uniform fluid 3-d model with various methods

The potential energy, density distribution of the fluid model was simulated with Velocity-Verlet algorithm, Beeman algorithm, and Gear prediction-correction integration method, and the results under different algorithms were obtained as shown in Fig.1 and Fig.2. Through comparative analysis of the calculation results of the fluid model parameters, the effects of different calculation methods on the fluid transport parameter results are discussed.

From the average potential energy calculation results and curve trend among the three calculation methods in Figure 1, we can see that it is more moderate using Beeman algorithm to obtain the average potential energy per unit of molecular energy; the average molecular potential energy obtained by the Verlet-Velocity algorithm is -1.16E-20 joule; while the Gear algorithm gives a molecular mean potential energy -1.32E-20 Joules. The value of potential energy can reflect the size of the intermolecular forces and the tightness of molecular arrangement from another aspect, from the average potential energy of Velocity-Verlet algorithm, we can see that the molecular arrangement is slightly closer than the other two algorithms.

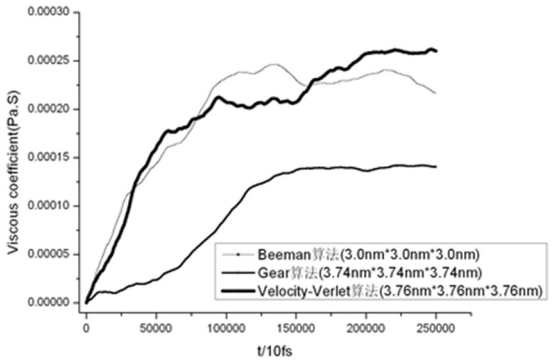


Figure 3. Viscous coefficient-time curve of uniform fluid 3-d model with various method

As a major application of the molecular dynamics method in the computation of fluid properties, the viscosity of the equilibrium nanofluids was calculated using the Green-Kubo formula with the three methods. As shown in Fig. 3, when the viscosity experimental value is 2.21E-4Pa.s [13], the relative error between the calculated results and the experimental values obtained by the Velocity-Verlet algorithm, the Beeman algorithm, and the Gear algorithm is 2.0%, 17.7%, 36.3% respectively. After comparing the viscosity curves obtained by different calculation methods, it can be known that the Velocity-Verlet algorithm can obtain ideal viscosity calculation results, but with a relatively large calculation range, the Beeman algorithm can obtain relatively accurate viscosity under a relatively small calculation range, when the calculation scale is reduced, the time required for each cycle calculation will be greatly shortened, so the Beeman algorithm has higher efficiency in fluid viscosity calculation.

4. Conclusions

1) With the simulation of Nanoscale copper rod at different temperatures, it was found that the temperature stability of Beeman algorithm and Gear algorithm is better than that of Velocity-Verlet algorithm.

2) When using the equilibrium molecular dynamics method to study the physical parameters of the fluid model, the potential energy curve and viscosity curve of the Gear algorithm require a relative long process into a stable state, and have a large fluctuation range, so the multi-period calculation is needed. In order to get stable results, although the Velocity-Verlet algorithm has fast convergence, and the density distribution and pressure distribution are relatively uniform, the computational scale needed for viscosity calculation is relatively large, which virtually lengthens the calculation cycle. The density and pressure distribution of the Beeman algorithm are uniform, and the viscosity simulation can be completed within a relatively small calculation range, which can shorten the calculation period and ensure the accuracy of calculation. Therefore, when the main purpose of the simulation calculation is to calculate the fluid transport coefficient, the Beeman algorithm can be considered as the integral calculation method.

References

[1] Chang W J, Fang T H. Influence of Temperature on Tensile and Fatigue Behavior of Nanoscale Copper Using Molecular Dynamics Simulation. Journal of Physics and Chemistry of Solids, 2004, 64(8): 1279-1283.
[2] Zhou XU, Xiu-xi WANG, Hai-jun LIANG, Heng-an WU. Numerical simulation of mechanical behavior of nanocrystalline and polycrystalline copper thin films. Acta Physica Sinica, 2004, 53(11): 3637-3643.
[3] Nosé S. A Molecular Dynamics Method for Simulations in the Canonical Ensemble. Molecular Physics, 1984, 52(2): 255-268.
[4] Weiwei Sun. Computer simulation experiments of classical fluids: Monte Carlo and Molecular Dynamics. Mechanics and Practice [M], 1983, 5(6):59-62.
[5] Xinian Cai. Molecular dynamics and physical mechanics. Edited by Institute of Mechanics, Chinese Academy of Sciences. Mechanics in the next 15 years [M]. Beijing: Science Press, 1986.100~110.
[6] L Verlet. Computer "Experiments" On Classical Fluids. I. Thermo dynamical Properties of Lennard-Jones Molecules[J]. Physical Review. 1967, 159(1): 98-106.
[7] W C Swope, H C Anderson, P H Berens, et al. A Computer Simulation Method for the Calculation of Equilibrium Constants for the Formation of Physical Clusters of Molecules: Application to Small Water Clusters[J]. Journal of Chemical Physics. 1982, 76: 637-649.
[8] D Beeman. Some Multistep Methods for Use in Molecular Dynamics Calculations[J]. Journal of Computational Physics. 1976, 20: 130-139.
[9] C W Gear. Numerical Initial Value Problems in Ordinary Differential Equation [M]. Englewood Cliffs NJ: Prentice-Hall, 1971.4.
[10] M. Doyama, Y. Kogure. Embedded Atom Potentials in FCC and Bcc Metals[J]. Computational Materials Science. 1999,14(1):80~83.
[11] Ying Hu et al. Physical Chemistry Reference [M]. Beijing: Higher Education Press, 2003.
[12] M.P. Allen, D.J. Tlldesley. Computer simulation of liquids, Oxford, Clarendon Press, 1987
[13] D.G.Naugle, J.H.Lunsford, J.R.Singer. Volume Viscosity in Liquid Argon at High Pressures[J]:The Journal of Chemical Physics.1966,45(12):4669-4676.