

Molecular dynamics and Monte Carlo simulation methods for soft matter

Martin Vögele

17.04.2012

Contents

1	Soft Matter	2
1.1	What is soft matter?	2
1.2	Why is soft matter soft?	2
1.3	Types of soft matter	3
1.3.1	Colloids	3
1.3.2	Amphiphiles	3
1.3.3	Polymeres	4
2	Molecular Dynamics Simulation	5
2.1	Principles	5
2.2	Algorithms	5
2.3	Observables	6
2.4	Applications	6
3	Monte Carlo Simulation	7
3.1	Basics	7
3.2	Applications	7
3.3	MC Integration	7
3.3.1	Calculating π	7
3.3.2	Importance Sampling	7
3.4	Metropolis Algorithm	8
4	Boundary Conditions	9
4.1	Types of Boundary Conditions	9
4.2	Minimum Image Convention	9
4.3	Linked Cells and Neighbor Lists	9

1 Soft Matter

1.1 What is soft matter?

The classical sorts of matter usually considered in physics are all homogeneous already on scales of nanometers. If one looks at a crystalline solid or at some kind of liquid, there is no more structure visible on that scale. But there are also more complex materials, where there are still structures up to nanometers which determine their typical behavior. This is shown for a suspension in figure 1, where we can see the different length scales with different structures: First the atomic scale, then the scale of the single layers and at last the range where the interface is only a line and just the different areas of oil and water are visible.

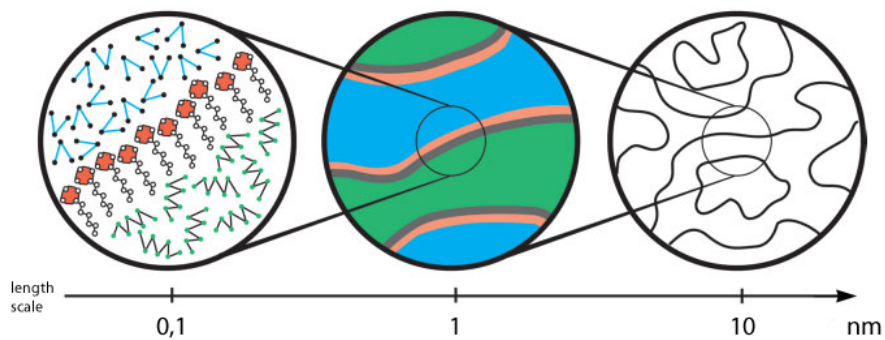


Figure 1: length scales of soft matter [4]

Based on this, we will use the following definition.

Soft matter:

Systems, whose properties are determined by structures in the mesoscopic scale (range between nano- and micrometers).

1.2 Why is soft matter soft?

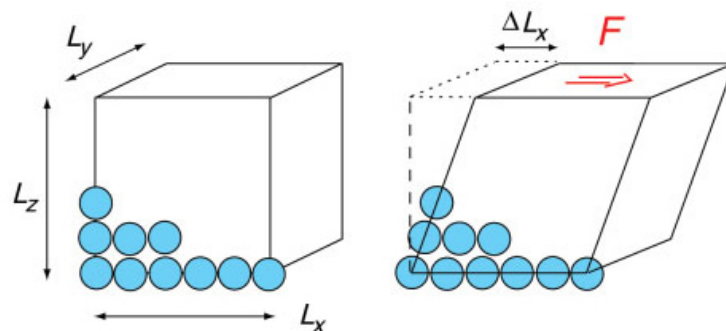


Figure 2: shearing [4]

The term soft matter describes one of its most important properties. We can understand this by estimating the shear modulus μ which is the resistance against a shearing force

(see figure 2). It is defined by

$$\frac{F}{L_x L_y} = \mu \frac{\Delta L_x}{L_z}.$$

As we can see, the dimension is $E \cdot L^{-3}$. For soft matter, the energy normally is in the thermal range, a typical value is $k_B T = (1/40) \text{ eV}$. The decisive part is the lattice constant. Let us assume a colloidal crystal. Its lattice size is 10000 times the one of a normal crystal, because that is the order of the determining structures. This means that soft matter has a shear modulus which is about 10^{12} times smaller than the one of a typical solid.

1.3 Types of soft matter

1.3.1 Colloids

Colloids are particles or drops with a size between 1 nm and 1 μm inside a liquid. The size differences are just in that range, where the surrounding liquid can be approximated as a continuum. This surrounding liquid causes thermal movement of the colloids by small collisions. The resulting movement of the particles is well-known as Brownian motion.

The particles themselves can interact by hard sphere repulsion, electrostatic forces, Van der Waals forces and by entropic effects. Most of these interactions can easily be modified: the electrostatic force by inserting salt whose ions shield the electric potential, the Van der Waals forces by changing the refraction index of the medium and the hard sphere and entropic effects by changing the form of the particles.

Important for applications is the nonlinear dynamic behavior when a force is exerted. this is for example used in dispersion paint, which becomes hard as soon as it isn't pressed against the wall any more.

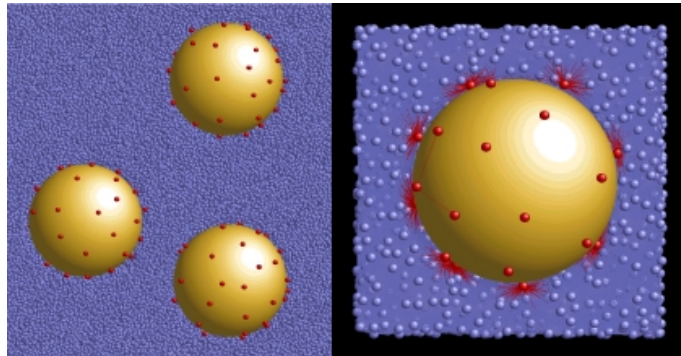


Figure 3: colloids in a medium [1]

1.3.2 Amphiphiles

Amphiphiles are molecules that are made up of a hydrophile head and lipophile tail. That is why they preferably accumulate at interfaces when put into a system containing water and oil. Substances that are made up of these components are called microemulsions and one can roughly distinguish between three sorts:

- more water \Rightarrow drops of oil in water
- more oil \Rightarrow drops of water in oil
- equal amounts \Rightarrow bicontinuous phase: water-oil-network

In case there is only water and amphiphiles, they form either spherical aggregates (mycels) or double layers (membranes).

The widest-known application are tensides used for cleaning. A simulation for such a tenside is shown in figure 4.

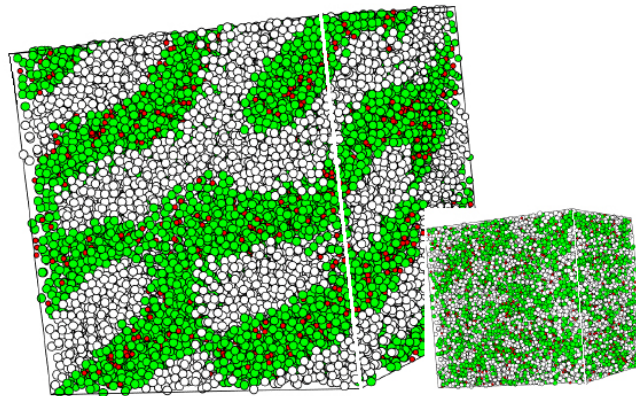


Figure 4: MD simulation of a system with 3350 tenside and 3200 water molecules. bottom right: random start configuration. [6]

1.3.3 Polymeres

Polymeres are chains of monomeres, that are connected by covalent bonds. These bonds can be rotated or flipped, which results in many degrees of freedom. This is why for long polymeres the conformation entropy is the dominat term in the free energy. So their behavior can be described very well statistical methods, for example as a random walk.

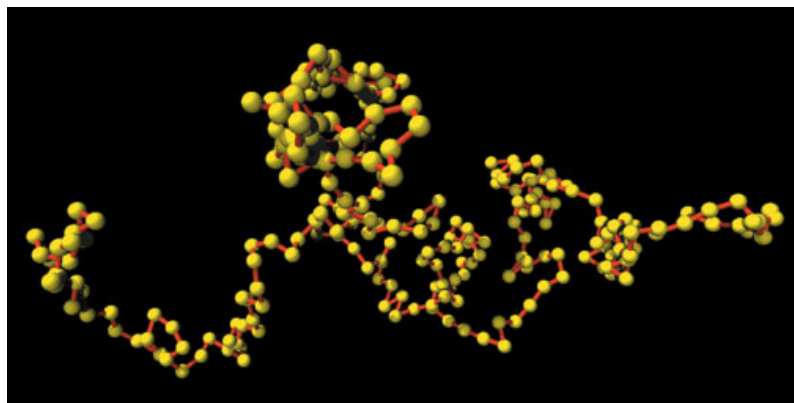


Figure 5: typical conformation of a fluctuating polymer chain, simulated with MC method [4]

2 Molecular Dynamics Simulation

2.1 Principles

In molecular dynamics simulations, one assumes a material (solid, liquid or gas) to contain a certain number N of classical particles. These particles interact by (mostly semiempirical) potentials $v_{ij} = v(\vec{r}_{ij})$. By differentiating these potentials, one obtains the forces

$$\vec{f}_i = \sum_{i \neq j} \vec{f}_{ij} = - \sum_{i \neq j} \frac{\partial v(\vec{r}_{ij})}{\partial \vec{r}_i}$$

which can be used to solve Newton's equations for each particle.

$$m_i \ddot{\vec{r}}_i = \vec{f}_i$$

So the trajectories for all particles can be obtained and used for calculating macroscopic properties.

2.2 Algorithms

As all many body problems, MD equations can't be solved analytically. That's why numerical algorithms have to be used. A good algorithm must be exact but nevertheless fast, so one has to find compromises. It also needs to be time-reversible and certain properties must be conserved (energy, momentum,...).

The most most common algorithms for MD are:

- Runge-Kutta 4th order
- Verlet algorithm
- Velocity Verlet algorithm
- Predictor-corrector method

Here the Verlet algorithm shall be shown as an example. It is derived by using two Taylor series of \vec{r}_i (backward and forward):

$$\begin{aligned}\vec{r}_i(t + \Delta t) &= \vec{r}_i(t) + \Delta t \cdot \dot{\vec{r}}_i + \frac{(\Delta t)^2}{2} \ddot{\vec{r}}_i + \frac{(\Delta t)^3}{6} \frac{d^3 \vec{r}_i}{dt^3} + \mathcal{O}(\Delta t^4) \\ \vec{r}_i(t - \Delta t) &= \vec{r}_i(t) - \Delta t \cdot \dot{\vec{r}}_i + \frac{(\Delta t)^2}{2} \ddot{\vec{r}}_i - \frac{(\Delta t)^3}{6} \frac{d^3 \vec{r}_i}{dt^3} + \mathcal{O}(\Delta t^4)\end{aligned}$$

Add both equations and use $\ddot{\vec{r}}_i = \frac{\vec{f}_i}{m_i}$:

$$\vec{r}_i(t + \Delta t) = 2\vec{r}_i(t) - \vec{r}_i(t - \Delta t) + \frac{\vec{f}_i}{m_i} \Delta t^2 + \mathcal{O}(\Delta t^4)$$

In this algorithm, the velocity is not needed, but the local coordinates of two steps before. The velocity can though be calculated as:

$$v(t) = \frac{\vec{r}_i(t + \Delta t) - \vec{r}_i(t - \Delta t)}{2\Delta t}$$

Because of using forward and backward series the Verlet algorithm is time-reversible and it can also be shown that the energy is conserved.

2.3 Observables

The MD simulation at first only gives the trajectories of the particles, but they can hardly be compared with experiments and anyway we are mostly interested in the macroscopic behavior. So macroscopic values, called observables, must be determined. They are usually evaluated by statistical averages over all particles. That's where we need the ergodic hypothesis, which says that in a chaotic system, all accessible microstates are equally probable over a long time, so that the time average is equal to the statistical average: $\bar{A}_t = \langle A \rangle$.

Important observables are the mean potential energy per particle, the mean kinetic energy per particle respectively the temperature or pressure.

2.4 Applications

- Noble gases (classical example: liquid Argon)
- *Solid state physics*:
all sorts of solids, especially alloys
- *Astrophysics*:
planetary motion, behavior of galaxies
- *Fluid dynamics*:
Liquids that are too complex for assuming continuum
- *Biochemistry*:
Large molecules, proteins

3 Monte Carlo Simulation

3.1 Basics

Monte Carlo simulations use probability theory to solve mathematical problems. The most important principles for the stochastic models are the law of large numbers and the central limit theorem which make sure that the probabilistic algorithm converges to a reproducible result for a sufficient number of random experiments. The results of such random experiments must always be analysed statistically. Depending on how elaborate the problem is, one can use naive methods with uniformly distributed random numbers or adapted distributions, which is called importance sampling.

3.2 Applications

- Calculating integrals, especially in higher dimensions
- evolution of galaxies
- production processes
- stock markets
- weather and climate
- calculating observables in quantum field theory
- virtual reality and artificial intelligence

3.3 MC Integration

3.3.1 Calculating π

One of the most simple (and the best-known) example for a Monte Carlo integration is the stochastic determination of π . The algorithm is quite easy: Create a square with a quarter circle in it. Then choose N_{total} random points in the square and count those who are inside the circle. The ratio is just $\frac{\pi}{4}$.

$$\frac{\pi}{4} = \frac{N_{\text{inside}}}{N_{\text{total}}}$$

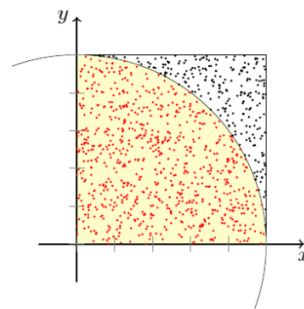


Figure 6: statistical calculation of π [8]

3.3.2 Importance Sampling

Suppose we want to calculate averages of the type

$$\langle A \rangle = \frac{\int d\vec{r} \exp[-\beta U(\vec{r})] A(\vec{r})}{\int d\vec{r} \exp[-\beta U(\vec{r})]}$$

with $p(\vec{r})$ being the probability density to find the system in a state around \vec{r} :

$$p(\vec{r}) = \frac{\exp[-\beta U(\vec{r})]}{Z} \quad \text{with} \quad Z = \int d\vec{r} \exp[-\beta U(\vec{r})]$$

Our problem is that the direct calculation of Z is too time-consuming. The solution is using importance sampling: We generate random states according to a probability distribution $w(\vec{r})$. So for the average holds:

$$\langle A \rangle \approx \frac{\sum_{i=1}^N \exp[-\beta U(\vec{r}_i)] A(\vec{r}_i) / w(\vec{r}_i)}{\sum_{i=1}^N \exp[-\beta U(\vec{r}_i)] / w(\vec{r}_i)}$$

If we choose $w(\vec{r}) = \exp[-\beta U(\vec{r})]$ the average $\langle A \rangle$ simplifies:

$$\Rightarrow \langle A \rangle \approx \frac{1}{N} \sum_{i=1}^N A(\vec{r}_i)$$

3.4 Metropolis Algorithm

The Metropolis algorithm creates a series of random variables, where the probability for a new state only depends on the probability for the previous one (Markov chain). Starting from the initial value x_0 , generate a potential successive value x_1 and calculate the transition probability $p(x_0, x_1)$. Then generate a random number $r \in [0, 1]$ and accept the new value if $p(x_0, x_1) > r$.

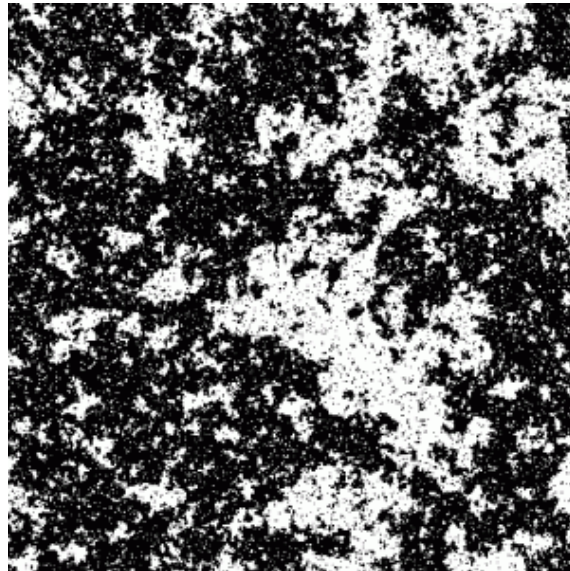


Figure 7: 2D Ising model 300×300, spin up/down as white/black. [7]

One very famous example is the Ising model for magnetization, where spins are placed on a lattice with external magnetic field H_z . It is assumed that they can only be $+1$ or -1 . The energy of the system is determined by the Hamiltonian $\hat{H} = -\frac{1}{2} \sum_{ij} J_{ij} s_i s_j + H_z \sum_i s_i$. The spins flip with a certain probability $p = \exp[-\beta(E_{\text{new}} - E_{\text{old}})]$ for $E_{\text{new}} > E_{\text{old}}$ and $p = 1$ for $E_{\text{new}} < E_{\text{old}}$. In this model, critical points can be determined and domains of different magnetization can be observed as in figure 7.

4 Boundary Conditions

4.1 Types of Boundary Conditions

The number of particles in a simulation is limited by calculation time and memory capacity. For sufficiently small systems, the boundary conditions can be obtained directly from the real problem. But most real systems are much larger. And because usually there is a different behavior at the surface and inside the system, artificial boundary conditions must be defined.

One option is the hard walls model, where every particle hitting the wall is reflected. These walls are like a vessel for the simulated material. But surface and volume can not be scaled linearly, which results in many unwanted effects.

The more common choice are therefore periodic boundary conditions: A particle that leaves the simulation box, enters at the opposite side with same velocity. This is equivalent to an infinitely large, periodic volume, which often is much nearer to the properties of a macroscopic system.

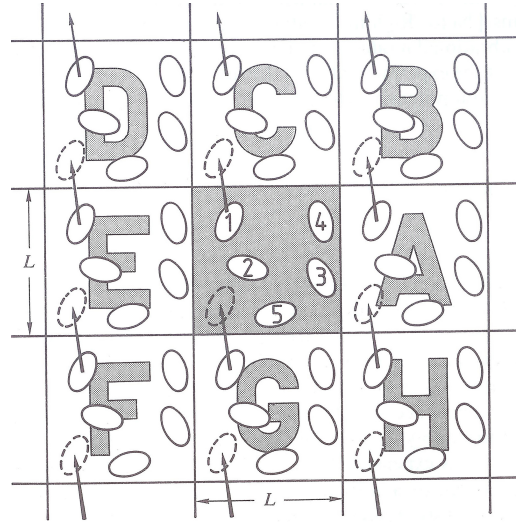


Figure 8: periodic boundary conditions

4.2 Minimum Image Convention

Periodic boundary conditions result in an infinite number of particles. This would lead to an infinite number of interactions which would need infinitely much time to be calculated. This problem is solved by the minimum image convention. For each particle, one can consider only the region inside a so called cut-off radius r_{cut} , where the interaction potential is truncated. This is a very good approximation for short range interactions.

4.3 Linked Cells and Neighbor Lists

There are two ways of speeding up a calculation with periodic boundary conditions.

The first one is the linked cell method. The simulation area is partitioned into cells and one must consider only particles in neighbor cells. Of course, the side length of the cells must be bigger than r_{cut} or else the potential is cut off too near.

The second possibility is using neighbor lists, that contains for each particle all neighbors inside a list radius $r_l > r_{\text{cut}}$ and only those are used for the calculation. These lists must be updated from time to time, so this method is only useful for large numbers of particles.

References

- [1] *SoftSimu - Biophysics and Soft Matter*. <http://www.softsimu.net>
- [2] ALLEN, M. P. ; TILDESLEY, D. J.: *Computer Simulation of Liquids*. Oxford University Press, USA, 1989. – ISBN 0198556454
- [3] FRENKEL, Daan ; SMIT, Berend: *Understanding Molecular Simulation, Second Edition: From Algorithms to Applications*. Academic Press, 2002. – ISBN 0–12–267351–4
- [4] GOMPPER, Gerhard ; DHONT, Jan K. G. ; RICHTER, Dieter: Komplexe Materialien auf mesoskopischer Skala: Was ist Weiche Materie? In: *Physik in unserer Zeit* 34 (2003), Nr. 1, S. 12–18. – ISSN 1521–3943
- [5] GRIEBEL, Michael ; KNAPEK, Stephan ; ZUMBUSCH, Gerhard ; CAGLAR, Attila: *Numerische Simulation in der Moleküldynamik: Numerik, Algorithmen, Parallelisierung, Anwendungen*. Springer, 2004. – ISBN 3–540–41856–3
- [6] HENTSCHE, R. *Theoretische Chemische Physik*. <http://constanze.materials.uni-wuppertal.de/Beispiele.html>
- [7] JUUL, Jeppe. *The Ising Model*. <http://jeppejuul.com/proposed-projects/bachelors-project/the-ising-model/>
- [8] WIKIPEDIA: *Kreiszahl* — *Wikipedia, Die freie Enzyklopädie*. <http://de.wikipedia.org/w/index.php?title=Kreiszahl&oldid=101469010>. 2012. – [Online; Stand 8. April 2012]