

7. tétel

Horváth Benedek

2019. június 9.

Kivonat

Molekuláris dinamika, Verlet- és sebesség-Verlet-algoritmus, termodinamikai mennyiségek meghatározása és relaxáció.

1. Bevezetés

A molekuladinamika hatékony és széles körben alkalmazott szimulációs eljárás kristályos, amorf, folyékony és gáz halmazállapotú anyagok, sőt, makromolekulák fizikai és kémiai tulajdonságainak meghatározására, illetve időfüggő mikroszkopikus folyamatok, mint például diffúzió modellezésére. A módszer hatékonysága abban rejlik, hogy az atomok belső – kvantummechanikával leírható – szerkezetét elhanyagolva azokat klasszikus és pontszerű objektumoknak tekintjük, és egyenként követjük a trajektóriájukat a klasszikus newtoni mozgásegyenletük numerikus integrálásával. A módszer tehát teljesen determinisztikus – ellentétben a sokrészecske-szimulációk egy másik típusával, a Monte Carlóval. Az atomok között párkölcsönhatásokat veszünk figyelembe, amit valamilyen potenciállal írunk le. Az egyszerű Lennard-Jones potenciáltól kezdve az igen bonyolult, molekulák rotációs és vibrációs módusait, illetve belső polarizációját is modellező erőtereken át a kovalens kötések létrejöttét és felbomlását is figyelembe vevő reaktív erőterekig igen sokféle potenciált használnak a szimulációkban. Közös tulajdonságuk, hogy elméleti – gyakran kvantummechanikai, pl. sűrűségfukcionál-elmélet, Hartree-Fock módszer – háttérszámolásokkal vannak validálva, többparaméteres függvényillesztés révén. (Részletekre itt most nem térek ki, az érdeklődők számára szemléltető példaként ajánlom a szén, hidrogén és oxigén atomokat tartalmazó rendszerek összetett Reactive Force Field típusú potenciáljáról szóló publikációt [6].) A párkölcsönhatások számolása és a részecskék egyenkénti mozgatása komolyan limitálja a szimulálható részecskék számát, a mai számítástechnika mellett jellemzően $10^3 - 10^8$ közé. A részecskék dinamikáját követve elegendően hosszú számolás után a rendszer kollektív termodinamikai tulajdonságai is számolhatók, a statisztikus fizika törvényeit alkalmazva.

2. A dinamika szimulációja

A molekuladinamikai szimulációban az egy részecskére ható erő az összes többi részecskével való párkölcsönhatásból származó erő eredője:

$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = \sum_{j \neq i=1}^N \mathbf{f}_{ij}, \quad (1)$$

ahol N az összes részecske száma a szimulációban. Másképp fogalmazva, a teljes rendszert leíró potenciál a párpotenciálok összege:

$$U(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N) = \sum_{i < j}^N u(r_{ij}) = \sum_{i=1}^{N-1} \sum_{j=i+1}^N u(r_{ij}), \quad (2)$$

ahol $r_{ij} = r_{ji} = |\mathbf{r}_i - \mathbf{r}_j|$ két részecske távolsága, $u(r_{ij})$ a köztük lévő távolságfüggő párpotenciál, az összegzésnél pedig minden párkölcsönhatást csak egyszer számolunk. A rendszert tehát egy konzervatív potenciál írja le, így a teljes energia (a kinetikus és potenciális energia összege) elviekben állandó [7]. Az egy részecskére ható erő a teljes potenciálból egyszerűen kifejezve:

$$\mathbf{F}_i(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N) = -\nabla_{\mathbf{r}_i} U(\mathbf{r}_1, \mathbf{r}_2 \dots \mathbf{r}_N). \quad (3)$$

Ugyanílyen logikával az egy elemi párkölcsönhatásból származó erő a párpotenciálból, komponensenként számolva:

$$(f_{ij})_x = -\frac{\partial u(r_{ij})}{\partial x_i} = -\frac{\partial u(r_{ij})}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial x_i}, \quad (4)$$

ahol

$$r_{ij} = ((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2)^{\frac{1}{2}}, \quad (5)$$

azaz

$$\frac{\partial r_{ij}}{\partial x_i} = \frac{x_i - x_j}{r_{ij}}. \quad (6)$$

A j részecske által az i részecskére ható erő tehát:

$$\mathbf{f}_{ij} = -\frac{du(r_{ij})}{dr_{ij}} \begin{pmatrix} \frac{x_i - x_j}{r_{ij}} \\ \frac{y_i - y_j}{r_{ij}} \\ \frac{z_i - z_j}{r_{ij}} \end{pmatrix} = -\frac{du(r_{ij})}{dr_{ij}} \hat{\mathbf{e}}_{ij}, \quad (7)$$

ahol $\hat{\mathbf{e}}_{ij}$ a j részecske pozíciójából az i felé mutató egységvektor. Látszik, hogy a hatás-ellenhatás törvényének megfelelően a párkölcsönhatásban résztvevő két részecskére ható erő csak előjelben különbözik, amit a szimulációban érdemes kihasználni. Mivel a részecskékre erő-ellenőrző párok hatnak – nincs külső erő –, a részecskék összimпульzusa elviekben zérus, így a rendszer tömegközéppontja nem mozoghat a szimuláció során. Szimulációs kód fejlesztése során ennek ellenőrzésével meggyőződhetünk az algoritmus és a numerikus integrálás helyességéről.

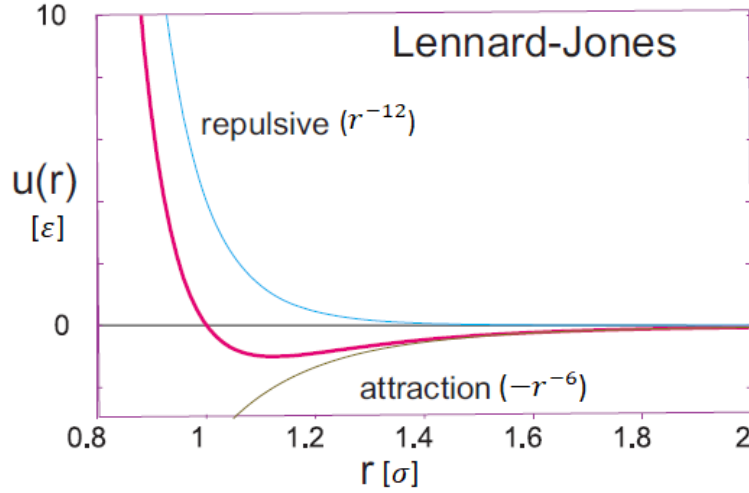
A párpotenciálra egy konkrét, egyszerű példa a vonzó-taszító Lennard-Jones potenciál, ami másodrendű kölcsönhatások leírására jól alkalmazható (a gyakorlatban például nemesgázokra, ahol nincsenek kovalens kötések és bonyolult, sok szabadsági fokú molekulák):

$$u(r) = 4\varepsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right], \quad (8)$$

ahol ε a kölcsönhatás erősségét, σ a hatótávolságát határozza meg, az r^{-6} tag a nagy távolságokon fellépő van der Waals vonzó kölcsönhatást, az r^{-12} tag pedig az elektronok Coulomb-taszításából és a Pauli-féle kizárási elvből kiindulva gyakorlatilag kemény gömbként veszi figyelembe az atomokat (lásd: 1. ábra). Bonyolultabb rendszerek leírására a kölcsönható részecskék típusától függően más és más, különféle tagok összegéből adódó párpotenciálokat használnak.

2.1. A potenciál levágása

A számítások egyszerűsítése érdekében a párpotenciálhoz minden esetben definiálnak egy levágási sugarat:



1. ábra. A Lennard-Jones potenciál távolságfüggése σ és ε egységekben, a taszító és vonzó tag összegéből képezve. Ábra: [7].

$$u(r) = \begin{cases} 4(r^{-12} - r^{-6}), & \text{ha } r < r_{\text{cut}}, \\ 0, & \text{ha } r > r_{\text{cut}}, \end{cases} \quad (9)$$

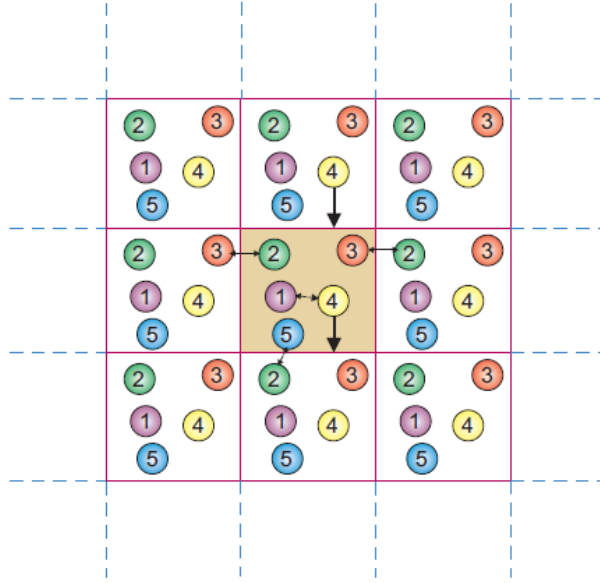
szemléltető példaként továbbra is a Lennard-Jones potenciálnál maradva. Az egy részecskére ható erőhöz tehát csak a levágási sugáron belül lévő részecskék adnak járulékot. A potenciál így nem folytonos, az abból számolt erő (a potenciál deriváltja) pedig szinguláris a levágási sugár értékénél ($r = r_{\text{cut}}$). Az elviekben konzervatív potenciál így már nem az, vagyis az energiamegmaradás nem teljesül egzaktul. A gyakorlatban viszont a levágási sugarat kellőképpen nagyra választjuk ahhoz, hogy a környezetében az erő értéke elhanyagolható legyen az egyéb közelítési és numerikus hibákhoz képest, így az energiamegmaradás ne sérüljön lényegesen [7].

2.2. Felületi jelenségek

Molekuladinamikai szimulációk segítségével gyakran makroszkopikus rendszerek viselkedésére igyekszünk következtetni 10^{23} részecskeszámmal. Mivel ennek a részecskeszámnak csak a töredékét szimuláljuk egy véges térfogatban, lényeges a felületi jelenségek megfelelő kezelése¹. Leggyakrabban periodikus határfeltételt alkalmazunk, az alábbiakban kizárólag ezt mutatom be. Ettől eltérő eset például a kapilláris csövekben modellezett dinamika, ahol kemény falat kell feltételezni, illetve biopolimerek szimulációjakor szükség lehet stohasztikus határfeltételre vagy határfeltétel nélküli (effektíve végtelen) szimulációs térfogatra [1].

A periodikus határfeltétel esetében gyakorlatilag azt feltételezzük, hogy a szimulációban definiált térfogat/síkidom végtelenszer ismétlődik minden irányban (szemléltetésként lásd: 2. ábra). Ha egy integrálási lépés során egy részecske elhagyná a szimulációs térfogatot, az ellenkező oldalon "beléptetjük" az úgynevezett képét. Precízen arra az esetre, ha a szimulációs

¹Minél kevesebb részecskét szimulálunk, annál nagyobb problémát jelenthetnek a mesterséges felületi jelenségek. Vegyünk például 1000 gázcsekkét egy $10 \times 10 \times 10$ egység méretű kockában, azaz egy részecske pont egy térfogategységet töltson be átlagosan. Így $10^3 - 8^3 = 488$ részecske a felület szomszédságában helyezkedik el, ez 49%-a az összes részecskének. 10^6 számú részecskénél ugyanilyen megfontolás mentén már csak a részecskék 6%-a található a felület szomszédságában.



2. ábra. A periodikus határfeltétel szemléltetése [7]. Középen a valódi szimulációs térfogat, ami minden irányban ismétlődik. A 4-es számú részecskékhez tartozó függőleges nyilak a határon kilépő részecske ellenkező oldalon történő beléptetését jelzik (lásd: 10. összefüggés), a kettős végű nyilak pedig azt mutatják, hogy a párkölcsönhatások számolásakor mikor vesszük az eredeti részecskéket, illetve valamelyiknek a képét ($r_{cut} < L_i/2$ esetben).

térfogat a $[0, L_x] \times [0, L_y] \times [0, L_z]$ téglatest, és a részecske x koordinátája lép ki a lehetséges intervallumból:

$$x \Rightarrow \begin{cases} x + L_x, & \text{ha } x < 0, \\ x - L_x, & \text{ha } x > 0. \end{cases} \quad (10)$$

Valójában a periodikus határfeltétel ennél többet jelent: a valódi atomok mellett ugyanis azok képei is részt vehetnek párkölcsönhatásokban. Ez effektíve olyan, mintha végtelen számú részecskét szimulálnánk [7], miközben technikailag csak N darab mozgásegyenletet integrálunk. Hogy mennyi képet kell számon tartani egy atomnak a kölcsönhatások számolásakor, az a párpotenciál levágási sugarától függ – ez teszi a gyakorlatban végezzé a párkölcsönhatások számát és a periodikusan kiterjesztett térfogatot. Ha a levágási sugár kisebb a szimulációs doboz egy adott oldalhosszána felénél ($r_{cut} < L_i/2$), elegendő egy képet nyilvántartani abban az irányban. Ilyenkor egy atom egy másik atomnak vagy a képével, vagy a valódi atommal hat kölcsön, attól függően, melyik van hozzá közelebb. Szemléltetésként lásd a 2. ábrán a kettős végű nyilakat.

3. Integrálási algoritmusok

A newtoni mozgásegyenletek integrálása közönséges differenciálegyenlet-rendszer lévén egzaktul megoldható numerikusan, sokféle integrálási sémával. Molekuladinamikai szimulációk során legelterjedtebben a Verlet- és sebesség-Verlet-algoritmusokat használjuk, az energiamegmaradás biztosítéka és a kiemelkedő stabilitás miatt.

3.1. Verlet-algoritmus

Egy részecske pozíciójának diszkrét léptetése τ időlépéssel, Taylor-sorfejtést alkalmazva:

$$\mathbf{r}_i(t + \tau) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\tau + \frac{1}{2}\mathbf{a}_i(t)\tau^2 + \frac{1}{6}\ddot{\mathbf{r}}_i(t)\tau^3 + \mathcal{O}(\tau^4). \quad (11)$$

Ugyanígy, negatív időlépést $(-\tau)$ véve:

$$\mathbf{r}_i(t - \tau) = \mathbf{r}_i(t) - \mathbf{v}_i(t)\tau + \frac{1}{2}\mathbf{a}_i(t)\tau^2 - \frac{1}{6}\ddot{\mathbf{r}}_i(t)\tau^3 + \mathcal{O}(\tau^4). \quad (12)$$

A fenti egyenleteket összeadva ((11) + (12)):

$$\mathbf{r}_i(t + \tau) + \mathbf{r}_i(t - \tau) = 2\mathbf{r}_i(t) + \mathbf{a}_i(t)\tau^2 + \mathcal{O}(\tau^4). \quad (13)$$

Ezt átrendezve, a gyorsulást átírva a következő alternatív összefüggést kapjuk a pozíció léptetésére:

$$\mathbf{r}_i(t + \tau) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \tau) + \frac{\mathbf{F}_i(\mathbf{r}_1(t), \mathbf{r}_2(t) \dots \mathbf{r}_N(t))}{m_i}\tau^2 + \mathcal{O}(\tau^4). \quad (14)$$

Ez az egyenlet (14) adja a Verlet-algoritmus alapját [1, 7]. Használatával a sebesség kiszámolása nélkül, negyedrendű integrálási hibával adhatjuk meg a részecskék pályáját. Az erők számolása közvetlenül a részecskék pozícióiból történik, minden időlépésben. Az algoritmus integrálási sémáját a 3. ábra szemlélteti. Mivel ebben a sémában az erőnek nincs explicit időfüggése – nem integrálással számoljuk, hanem a részecskék pozícióiból, a konzervatív potenciál kiértékelésével –, a teljes energiamegmaradás teljesül, nem befolyásolja az integrálási hiba. Noha a sebesség kiszámolása nem szükséges a pálya követéséhez, molekuladinamikai szimulációkban általában szükség van a sebesség ismeretére (a teljes mozgási energia, illetve ebből a hőmérséklet számolásához, lásd később, a 4. fejezetben). Ez a középpontidifferencia-módszerrel számolható:

$$\mathbf{v}_i(t) = \frac{d\mathbf{r}_i(t)}{dt} = \frac{\mathbf{r}_i(t + \tau) - \mathbf{r}_i(t - \tau)}{2\tau} + \mathcal{O}(\tau^2). \quad (15)$$

Megjegyzendő, hogy a Verlet-szimuláció kezdetén szükséges egy inicializáló integrálási lépés, mivel az algoritmus az új pozíciót két korábbi időlépésben felvett pozícióból számolja. Az inicializáció egyszerű Euler-lépéssel:

$$\mathbf{r}_i(-\tau) = \mathbf{r}_i(0) - \mathbf{v}_i(0)\tau + \frac{1}{2}\frac{\mathbf{F}_i(\mathbf{r}_1(0), \mathbf{r}_2(0) \dots \mathbf{r}_N(0))}{m_i}\tau^2 + \mathcal{O}(\tau^3). \quad (16)$$

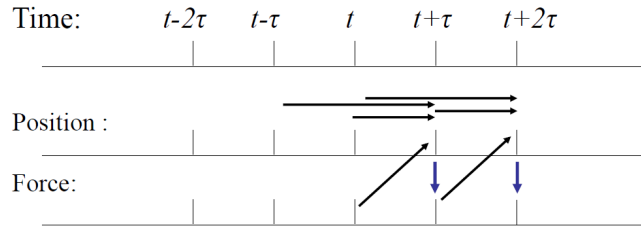
3.2. Sebesség-Verlet-algoritmus

A Verlet-algoritmus másik verziója a sebesség-Verlet-algoritmus, ami explicit módon számolja a sebességet és használja fel az integráláshoz minden lépésben:

$$\mathbf{r}_i(t + \tau) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\tau + \frac{1}{2}\mathbf{a}_i(t)\tau^2 + \mathcal{O}(\tau^3), \quad (17)$$

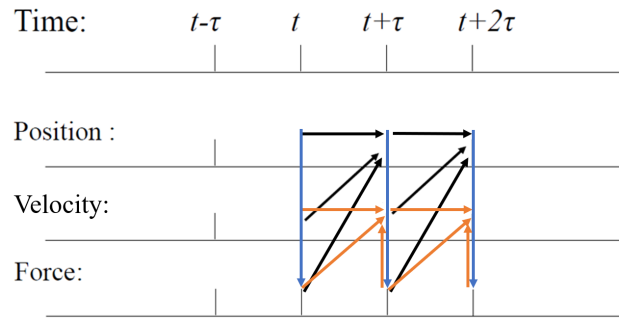
$$\begin{aligned} \mathbf{v}_i(t + \tau) &= \mathbf{v}_i(t) + \overline{\mathbf{a}_i(t)}\tau + \mathcal{O}(\tau^2) = \\ &= \mathbf{v}_i(t) + \left[\frac{\mathbf{a}_i(t + \tau) + \mathbf{a}_i(t)}{2} \right] \tau + \mathcal{O}(\tau^2). \end{aligned} \quad (18)$$

Time line for a Verlet simulation



3. ábra. A pozíció léptetése és az erő számolása a Verlet-algoritmus során [1].

Time line for a velocity-Verlet simulation



4. ábra. A pozíció, a sebesség és az erő számolásának menete a sebesség-Verlet-algoritmus során.

Megjegyzendő, hogy az $\mathbf{a}_i(t)$ jelölésmód a gyorsulásra csak rövidítés: azt továbbra is az atomok pozíciójából, azaz implicit időfüggés alapján számoljuk, ahogy a (14) és (16) számú összefüggésekben. A koordináták, a sebességek és az erő számolásának a fenti (17 és 18) összefüggések által meghatározott menetét a 4. ábra szemlélteti. A számolás pontosságát növeli, hogy a sebesség léptetésekor a gyorsulást az időlépés elején és végén lévő konfigurációból számolt érték átlagának vesszük, illetve, hogy az így számolt sebességet használjuk a koordináta következő léptetésekor. Ez ellensúlyozza azt a hátrányt, hogy a sebesség-Verlet-algoritmus hibája a koordináta léptetésére alacsonyabb rendű a Verlet-algortimushoz képest (harmadrendű vs negyedrendű), és végeredményben a két algoritmus hasonló pontosságot eredményez [7]. Ahogy az látható, a számolás során a koordinátákat mindig előbb kell léptetnünk, mint a sebességet. A koordináták léptetésekor fontos szerepe van a periodikus határfeltételnek, hiszen módosítja az atomokra ható erőt, így a sebesség közvetlen utána következő léptetését. A határokon történő ellenőrzést tehát a pozíció léptetése után, a sebesség léptetése előtt kell elvégezni.

4. Termodinamikai mennyiségek meghatározása

Egy rendszer részecskéinek trajektóriáit – mikroszkopikus viselkedését – követő molekuladinamikai szimuláció eredményeit a statisztikus fizika törvényei alapján hozhatjuk kapcsolatba a makroszkopikus termodinamikai paraméterekkel. Ehhez elegendően nagy számú részecskére van szükség. A sokrészecske-rendszer teljes translációs kinetikus energiája egy adott konfigu-

rációban (időlépésben) a szimulációból a következő egyszerű összegzés szerint kapható:

$$KE = \sum_{i=1}^N \frac{1}{2} m_i v_i^2. \quad (19)$$

Az ekvipartíció tétele szerint a hőmérséklet és a translációs kinetikus energia kapcsolata:

$$\langle KE \rangle = N \frac{d}{2} k_B T, \quad (20)$$

ahol d a térbeli dimenzió (a gyakorlatban 2 vagy 3), a $\langle KE \rangle$ jelölés pedig a mozgási energia időátlagát jelenti. Az időátlag azért lényeges, mert a kinetikus és potenciális energia egy bizonyos érték körül fluktuál az egyensúlyi szimulációban, azoknak csak az összege állandó, így egy időpillanatra vonatkozó adat nem releváns az egyensúlyi hőmérsékletre nézve. A hőmérséklet a fenti egyenlet (20) alapján:

$$T = \frac{2 \langle KE \rangle}{k_B d N}. \quad (21)$$

A nyomás a Viriál-tétel felhasználásával számolható [7, 2]:

$$pV = N k_B T + \frac{w}{d}, \quad (22)$$

ahol

$$w = \left\langle \sum_{i < j}^N \mathbf{r}_{ij} \cdot \mathbf{f}_{ij} \right\rangle \quad (23)$$

a viriál, az összes párkölcsönhatásból származó erő nyomatéka időátlagban. Az összegzés során minden részecskepárt csak egyszer számolunk, és a 2.2 fejezetben elmondottak szerint a részecskék képei is adhatnak járulékot az eredeti részecskék helyett/mellett. Az időátlag ismét a fluktuációk kiküszöbölése érdekében szükséges. A nyomás tehát, a (21) és (22) számú összefüggéseket felhasználva, a szimulációból közvetlenül számolható mennyiségekkel kifejezve:

$$p = \frac{1}{Vd} (2 \langle KE \rangle + w). \quad (24)$$

5. Relaxáció

Molekuladinamikai szimulációval egyaránt modellezhető egyensúlyi és nemegyensúlyi folyamatok dinamikája, és a célnak megfelelően különféle – esetenként bonyolult és egyedi – kezdőfeltételeket lehet implementálni. A legegyszerűbb típusú szimuláció, amikor valamilyen random vagy mesterségesen rendezett kezdeti állapotból egyensúlyi állapotba relaxáltatjuk a rendszert. Ilyenkor alapesetben mikrokanonikus sokaságnak tekinthető a rendszer, hiszen a részecskék száma, a szimulációs térfogat és a teljes energia állandó [7]; a szimuláció során folyamatos átmenet van a kinetikus és potenciális energia között, és a kettő aránya végül egy állandó érték körül fluktuál bizonyos hibahatáron belül.

Hogy csak egy egyszerű példát vegyünk: gáz halmazállapotú rendszer szimulációjakor például lehet egy bizonyos hőmérsékletnek megfelelő Maxwell-Boltzmann eloszlásból random kezdősebességet generálni a gáزرészecskéknél, majd a newtoni dinamika szerint mozgatva őket

megvárni az egyensúlyi állapot kialakulását az adott hőmérsékleten². A hőmérséklet rögzítésére egyszerű megoldás a sebességskálázás: a hőmérsékletdrift elkerülése érdekében az atomok mozgási energiáját a szimuláció elején bizonyos időlépésenként úgy növeljük vagy csökkentjük egységesen egy szorzófaktorral, hogy a pillanatnyi hőmérséklet megfeleljen a kívánt egyensúlyi hőmérsékletnek. Azaz:

$$\mathbf{v}'_i(t) = s \cdot \mathbf{v}_i(t), \quad (25)$$

ahol

$$s = \sqrt{\frac{T_{eq}}{T(t)}} \quad (26)$$

a skálafaktor, T_{eq} a kívánt egyensúlyi, $T(t)$ a pillanatnyi hőmérséklet. A skálafaktor hőmérsékletek arányától való gyökös függéséhez lásd a (19) és (21) számú összefüggéseket. Az egyensúlyi állapotba történő relaxáció után már nincs szükség a sebességskálázásra, a kinetikus és potenciális energia, és így a hőmérséklet is az egyensúlyi értékük körül fluktuál. Az állandó fluktuáció miatt termodinamikai mennyiségek számolásához minél hosszabban kell futtatni a számolást, hogy az így képzett időátlagok (lásd: (21), (23)) hibája minél kisebb legyen ($\text{Err} \propto \frac{1}{\sqrt{N}}$ szerint).

A gyakorlatban a fent ismertetett sebességskálázásnál bonyolultabb módszerekkel biztosítjuk a rendszer megfelelő egyensúlyi állapotba való relaxációját: a célnak megfelelően úgynevezett termosztátokat/barosztátokat lehet alkalmazni az atomok egy csoportjára vagy a rendszer egészére, amelyek megadják, hogy a hőmérséklet/nyomás milyen értékhez konvergáljon, illetve, hogy az integrálás során ez milyen numerikus módszerrel legyen biztosítva (pl. Langevin termosztát, 300 K hőmérséklet [3]). Ennek részletei túlmutatnak a tétel mélységein, az érdeklődők számára a bevett és széles körben alkalmazható nyílt forrású LAMMPS szoftver dokumentációját ajánlom kiindulópontnak: [4, 5].

Hivatkozások

- [1] David J. Willock: Molecular Dynamics. Lecture 1: Basic Ideas, Newtonian motion, trajectories, sampling. ISIS Neutron training course, Cardiff University, U.K. 2014. március 6. Webcím: <https://www.isis.stfc.ac.uk/Pages/md-lecture-1.pdf>.
- [2] LAMMPS manual, compute pressure command: https://lammps.sandia.gov/doc/compute_pressure.html.
- [3] LAMMPS manual, fix langevin command: https://lammps.sandia.gov/doc/fix_langevin.html.
- [4] LAMMPS manual, overview on thermostatting: https://lammps.sandia.gov/doc/Howto_thermostat.html.
- [5] LAMMPS manual, overview on barostatting: https://lammps.sandia.gov/doc/Howto_barostat.html.

²Amikor egy bizonyos hőmérsékletnek megfelelő Maxwell-Boltzmann-eloszlásból generálunk sebességet a részecskéknél, az korántsem egyensúlyi helyzet, hiszen a mozgás ilyenkor teljesen korrelálatlan.

- [6] Kimberly Chenoweth, Adri C. T. van Duin, and William A. Goddard III. ReaxFF Reactive Force Field for Molecular Dynamics Simulations of Hydrocarbon Oxidation. *The Journal of Physical Chemistry A*, 112:1040–1053, 2008.
- [7] Cristian C. Bordeianu Rubin H. Landau, Manuel José Páez. *A Survey of Computational Physics*. Introductory Computational Science. Princeton University Press, 2012.