

S-114.250
Laskennallisen tieteen erikoiskurssi

Antti Kuronen
Teknillinen korkeakoulu
Laskennallisen tekniikan laboratorio
PL 9400
02015 TKK

tammikuu 1999

Esittely iii**Materiaalia iii****Esitiedot ja edellytykset iv****Yleistä 1**

Fysiikan tietokonesimulaatioista 1

Deterministiset ja stokastiset simulaatiomenetelmät 2

Monte Carlo -menetelmän historiasta 5

Esimerkkejä sovellutuksista 6

Statistista fysiikkaa 10

Faasiavaruus 10

Ensemblet 12

Makroskooppisten suureiden laskeminen 17

Todennäköisyyslaskentaa 20

Yleistä 20

Tärkeimmät todennäköisyysjakaumat 23

Ajasta riippuvat ilmiöt 25

Metropoliksen Monte Carlo 35

Monte Carlo -integrointi 35

Metropolis-algoritmi 38

Metropolis-algoritmin parannuksia 49

Monte Carlo -simulaatiot eri ensembleissä 57

Yleistä 57

Mikrokanoninen ensemble 57

Vakiopaine-ensemble 58

Suurkanoninen ensemble 63

Simulaatio-ohjelma käytännössä 71

Reunaehdot 71

Potentiaalienergian laskeminen 74

Pitkän kantaman potentiaalit 78

Alkuehdot 83

Tulosten analysoinnista 85

Satunnaisluvuista 91

Tasaisesti jakautuneiden satunnaislukujen generointi 91

Erilaiset satunnaislukujakaumat 100

Atomien välisistä vuorovaikutuspotentiaaleista 105

Yleistä 105

Idealisoidut potentiaalit 106

Redusoidut yksiköt 109

Realistiset potentiaalit 110

Tasapainosimulaatioiden sovellutuksia 115

Tilanyhtälö ja faasitasapaino 115

Faasitasapainon simulointi 121

Ajasta riippuvien ilmiöiden simulointi 128

Yleistä 128

Säteilyn kulkeutumisen simulointi 129

Varaustenkuljettajien kulkeutuminen puolijohdemateriaaleissa 143

Kineettinen Monte Carlo 144

Yleistä 144

Pinnan kasvu ja pintadiffuusio 149

Van der Waalsin tilanyhtälö 167

Esittely

Kurssi **S-114.250 Laskennallisen tieteen erikoiskurssi** kuuluu laskennallisen tekniikan pää/sivuaineen vaihtoehtoihin opintoihin.

Kurssin laajuus on 4 opintoviikkoa ja suoritus koostuu harjoituksista (osuus loppuarvosanassa 40%) ja lopputyöstä (60%).

Kevätlukukaudella 1999 luennoitsijana toimii
tutkija Antti Kuronen
Laskennallisen tekniikan laboratorio
puh. 451 4846
e-mail Antti.Kuronen@hut.fi .

Lukujärjestys on seuraavanlainen:

luennot: ma klo 12-14 sali H402
to klo 9-10 sali F301
harjoitukset: ma klo 12-14 sali H402

Päämääränä on perehdyttää opiskelija stokastisiin ja jonkin verran myös deterministisiin simulointimenetelmiin ja niiden soveltamiseen fysikaalisten ilmiöiden tutkimisessa.

Materiaalia

Tärkein materiaali on kurssin luentomoniste, jotka on käsissäsi. Muuta materiaalia ja viitteitä löydät kurssin kotisivulta

<http://www.lce.hut.fi/teaching/S-114.250>

Kotisivulla on mm. kurssin ilmoitustaulu.

Kirjoista voisi mainita seuraavat:

1. M. P. Allen, D. J. Tildesley: *Computer Simulation of Liquids*.
Varsin kattava ja usein viitattu teos, joka käsittelee molekyyliysteemien

siumlontia Monte Carlo - ja molekyylidynamiikkamenetelmillä. (Viittauksissa nimellä *Allen-Tildesley*.)

2. D.W.Heermann: *Computer Simulation Methods in Theoretical Physics*. Kirja käsittelee sekä deterministisiä (molekyylidynamiikka) että stokastisia (Monte Carlo, Langevin) simulointimenetelmiä. (Viittauksissa nimellä *Heermann*.)

3. K.Binder, D.W.Heermann: *Monte Carlo Simulation in Statistical Physics: An Introduction*. Kirjassa käydään läpi Monte Carlo -menetelmän teoreettisia perusteita statistisen fysiikan sovellutusten kannalta. (Viittauksissa nimellä *Binder-Heermann*.)

4. D.Frenkel, B.Smit: *Understanding Molecular Simulation: From Algorithms to Applications*. Kirja rajoittuu -kuten nimikin kertoo- molekyyliysteemien simulointiin. Sekä molekyylidynamiikka että Monte Carlo -menetelmät käydään läpi. Kirjaan liittyy esimerkkiohjelmia, jotka voi hakea kirjan kotisivulta osoitteesta http://ct-cr6.chem.uva.nl/frenkel_smit . (Viittauksissa nimellä Frenkel-Smit.)

5. A.Kersch, W.J. Mokoroff: *Transport Simulation in Microelectronics*. Kirja käsittelee Monte Carlo -menetelmän käyttöä mikroelektroniikan materiaalien prosessoinnin ja komponenttien varauksenkuljetuksen mallintamisessa. (Viittauksissa nimellä Kersch-Mokoroff.)

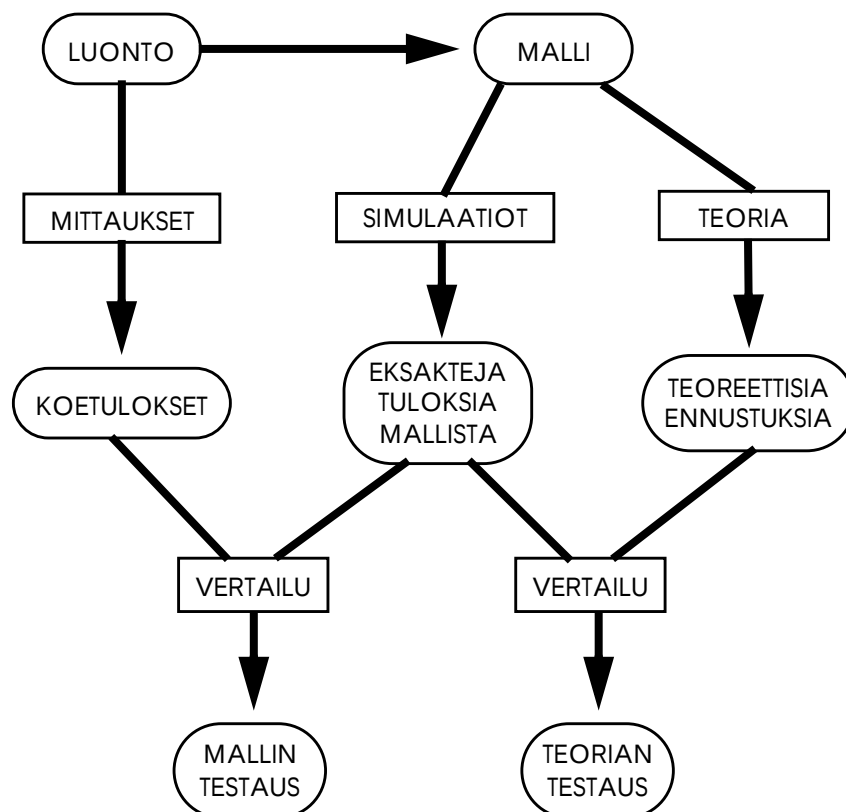
Esitiedot ja edellytykset

- Statistisen fysiikan perusteet.
- Jonkin ohjelmointikielen hallinta (C tai Fortran).
 - Kurssin laskuharjoituksissa ja varsinkin lopputyössä laaditaan ohjelmia, joten ohjelmointitaito on välttämätön.
- Tietokoneen käyttömahdollisuus.

1 Yleistä

1.1 Fysiikan tietokonesimulaatioista

Simulaatioiden (tai laskennallisten menetelmien) asemaa fysiikassa voi parhaiten havainnollistaa seuraavalla kuvalla.



Kuva 1.1: Tietokonesimulaatioiden asema fysiikassa.

Simulaatioilla voidaan testata sekä teorioita että malleja. Ne kaventavat kuilua teorian ja mittausten välillä. Toisaalta tämä menetelmä on algoritmien lähentymistapa: lopputulokseen ei ole 'oikotietä'.

Äärellisen laskentakapasiteetin vuoksi tutkittavan systeemin koko ja myös ilmiön mallintamisen aikaskaala ovat rajoitettuja. Tämän vuoksi ns. äärelliseen koon vaikutusten (finite size effects) osuutta simulaatiotuloksiin tulisi aina tutkia.

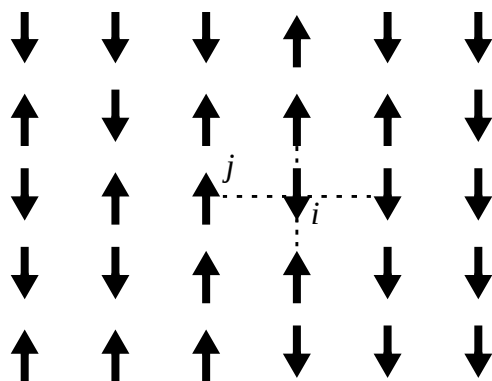
Lisäksi simulaatioiden statistisesta luonteesta johtuen tuloksissa on statistista epämääräisyyttä.

Simulaation suoritus voidaan jakaa eri vaiheisiin kuvan 1.2 mukaan.

Malli voi olla hyvinkin ilmeinen (esim. kiinteä aine=atomit, jotka vuorovaikuttavat potentiaalin välityksellä) tai sitten ei. Otetaan esimerkiksi ns. Isingin malli. Siinä systeemi koostuu 'spineistä' S_i , joilla voi olla arvot ± 1 . Systeemin energia on

$$E = -J \sum_{\langle i, j \rangle} S_i S_j, \quad (1.1)$$

missä summa käy yli lähinaapuriparien.

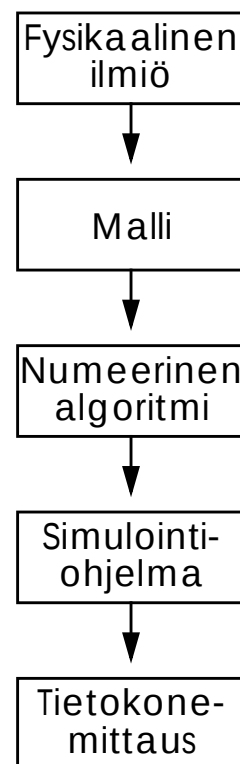


Kuva 1.3: Isingin malli

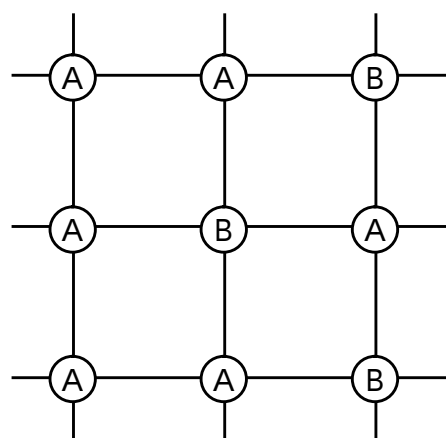
Isingin mallilla voidaan jossain määrin kuvata ferromagneettisia aineita, mihin malli alunperin kehitettiin. Mutta se soveltuu moneen muuhunkin ilmiöön. Esimerkiksi kaksikomponenttista metalliseosta voidaan kuvata tällä mallilla.

Numeerisen algoritmin avulla pystytään laskemaan mallin antamia tuloksia. Se voi olla esim. algoritmi, jolla kuljetetaan systeemiä ajassa eteenpäin ('liiketyöt') tai jolla käydään läpi systeemin faasiavaruutta.

Simulaatioissa ja kokeellisissa mittauksissa on samankaltaisia piirteitä (raakadatan käsittely, statistiset virheet tuloksissa,...), joten usein simulaatioita kutsutaan tietokonemittauksiksi (computer experiments).



Kuva 1.2: Simulaation suorituksen vaiheet.



Kuva 1.4: Kaksikomponenttisen metalliseoksen Isingin malli.,

1.2 Deterministiset ja stokastiset simulaatiomenetelmät

Simulaatiomenetelmät voidaan tietyllä tasolla jakaa deterministisiin ja stokastisiin. Esimerkiksi laskettaessa monen vapausasteen systeemin tasapaino-ominaisuuksia jako on selkeä.

Olkoon systeemin Hamiltonin funktio (eli kokonaisenergia) H , ja tilan määrää vektori $\mathbf{x}$

$$\mathbf{x} = (x_1, x_2, \dots, x_n) , \quad (1.2)$$

missä n on vapausasteiden lukumäärä. Monesti systeemi koostuu N :stä atomista tai molekyylistä, jolloin $n = 6N$ ja

$$\mathbf{x} = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N, \mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N) . \quad (1.3)$$

Usein haluamme laskea tietyn suureen A odotusarvon

$$\langle A \rangle = Z^{-1} \int_{\Omega} A(\mathbf{x}) f(H(\mathbf{x})) d\mathbf{x} , \quad (1.4)$$

$$Z = \int_{\Omega} f(H(\mathbf{x})) d\mathbf{x} \quad (1.5)$$

missä Ω on systeemin faasiavaruus ja $f(H(\mathbf{x}))$ on todennäköisyystiheysfunktio.

Tämä on ns. ensemblekeskiarvo, jota ei suoraan voi laskea simuloimalla, koska koko faasiavaruutta ei voida käydä läpi. On tyydyttävä enemmän tai vähemmän edustavaan otokseen.

Deterministinen tapa antaa systeemin oman dynamiikan (liikeyhtälöt) kuljettaa tilavektoria $\mathbf{x}$ halki faasiavaruuden. Ensemblekeskiarvo korvataan aikakeskiarvolla $\bar{A}_t$

$$\bar{A}_t = t^{-1} \int_0^t A(\mathbf{x}(\tau)) d\tau . \quad (1.6)$$

Ergodisuus takaa, että $\langle A \rangle = \bar{A}_{\infty}$. Käytännössä on tyydyttävä siihen, että $\langle A \rangle \approx \bar{A}_t$. Tätä menetelmää kutsutaan **molekyyliidynamiikaksi (MD)**.

Stokastisessa menetelmässä systeemin tiloja $\mathbf{x}^{(m)}$ generoidaan satunnaisesti Markovin prosessilla. Useimmiten ollaan kiinnostuneita vain $\mathbf{x}$:n konfiguraatio-osasta eli koordinaateista $\mathbf{r}_i$. Liikemäärä voidaan integroida erikseen. Suureen A odotusarvo on nyt

$$\bar{A} = M^{-1} \sum_{k=1}^M A(\mathbf{x}^{(m)}) . \quad (1.7)$$

Tämä menetelmä on **Monte Carlo -simulaatio (MC)**. Ongelmana on kehittää tehokas algoritmi, jolla käydään läpi faasiavaruuden niitä osia, jotka ovat merkittäviä suureen A laskeamisen kannalta. Yksi algoritmi on kanonisen ensemblen yhteydessä käytettävä Metropolis-algoritmi:

1. generoi uusi tila: $\mathbf{x}^{(m)} \rightarrow \mathbf{x}^{(m+1)}$
2. laske energiaero $\Delta E = H(\mathbf{x}^{(m+1)}) - H(\mathbf{x}^{(m)})$
3. jos $\Delta E < 0$, hyväksy uusi tila todennäköisyydellä 1 , muuten hyväksy se todennäköisyydellä $\exp(-\Delta E/kT)$.

Voidaan osoittaa (ja myöhemmin kurssilla osoitetaan), että algoritmi generoi tiloja, jotka noudattavat kanonista jakaumaa $\exp(-H(\mathbf{x})/kT)$.

MD-menetelmän etuna on, että sillä voidaan tutkia ajasta riippuvia ilmiöitä; Metropolis-MC:llä taas ei. Sen avulla toisaalta voidaan tutkia systeemeitä, joilla ei varsinaista sisäistä dynamiikkaa ole ollenkaan. Esimerkkinä olkoon Ising-malli:

$$H_{\text{Ising}} = -J \sum_{\langle i, j \rangle} S_i S_j - B \sum_i S_i ; S_i = \pm 1 \quad (1.8)$$

Lisäksi MC-menetelmän avulla voidaan systeemiin tuoda esimerkiksi kemiallisia vapausasteita: algoritmi voi muuttaa atomin tai molekyylin lajia.

Huomaa, että ylläolevassa esimerkissä ei itse systeemissä (tai sen mallissa) ollut mitään stokastista, satunnaista. Integroitaessa faasiavaruuden yli MC-menetelmässä vain käytettiin satunnaisotantaa.

Itse systeemissä voi olla satunnaislementtejä.

Esimerkiksi gammasäteilyn etenemisessä väliaineessa vuorovaikutusten välimatka on satunnainen suure, joka noudattaa tiettyä todennäköisyysjakaumaa.

Satunnaisuus voi johtua myös 'puutteellisesta' mallista: jotkut vapausasteet otetaan huomioon stokastisina elementteinä. Esimerkkinä Brownin liike, jossa väliaineen vaikutus otetaan huomioon Langevinin liikeyhtälöllä

$$m \frac{dv}{dt} = -\gamma v + R(t) , \quad (1.9)$$

missä γ on kitkavakio ja $R(t)$ ajasta riippuva satunnaisvoima, jonka statistisista ominaisuuksista tiedämme jotain. Langevinin liikeyhtälöitä voidaan käyttää kuvaamaan systeemiä ympäröivää lämpökylpyä (kanoninen ensemble).

Termiä Monte Carlo käytetään varsin erilaisisten stokastisten simulaatiomenetelmien yhteydessä. Karkeasti ottaen voisi sanoa, että Monte Carlo -menetelmiä ovat simulaatiot, joissa käytetään paljon satunnaislukuja.

1.3 Monte Carlo -menetelmän historiasta

Italialainen matemaatikko Lazzerini arvioi piin likiarvoa heittämällä 3407 kertaa neulan tasavälisen suorien päälle (Buffonin neula). Likiarvoksi tuli 3.1415929 eli 7 numeron tarkkuudella oikea (sattumalta?).

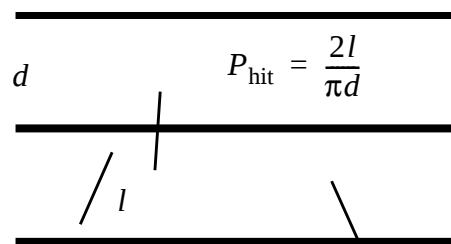
W. S. Gossett ('Student') arvioi -jakaumansa korrelaatiokertoimia otantakokeella.

Lordi Kelvinin assistentti generoi 5000 satunnaista rataa tutkiessaan hiukkasen ja kaarevan välisiä törmäyksiä.

Varsinainen Metropolis-MC kehitettiin Los Alamosissa 50-luvulla. Ensimmäinen julkaisu lie-
nee

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, Equation of State Calculations by Fast Computing Machines, *J. Chem. Phys.*, **21** (1953) 1087.

Julkaisussa tutkittiin kaksikulotteisten kovien kiekkojen tilanyhtälöä.



Kuva 1.5: Buffonin neula

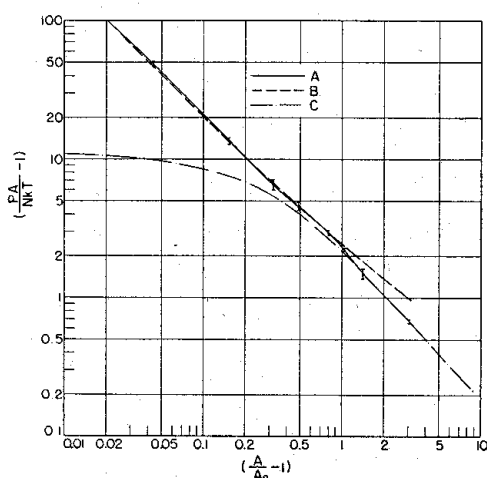


FIG. 4. A plot of $(PA/NkT) - 1$ versus $(A/A_0) - 1$. Curve A (solid line) gives the results of this paper. Curves B and C (dashed and dot-dashed lines) give the results of the free volume theory and of the first four virial coefficients, respectively.

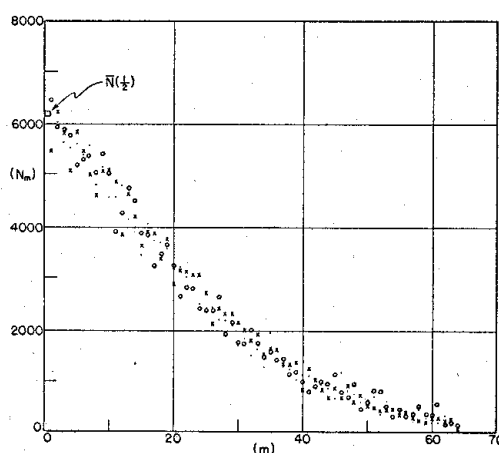


FIG. 5. The radial distribution function N_m for $\nu=5$, $(A/A_0) = 1.31966$, $K=1.5$. The average of the extrapolated values of N_1 in $N_1 = 6301$. The resultant value of $(PA/NkT) - 1$ is $64N_1/N^2(K^2 - 1)$ or 6.43. Values after 16 cycles, $\bullet$; after 32, $\times$; and after 48, $\circ$.

Kuva 1.6: Ensimmäinen MC-julkaisu.

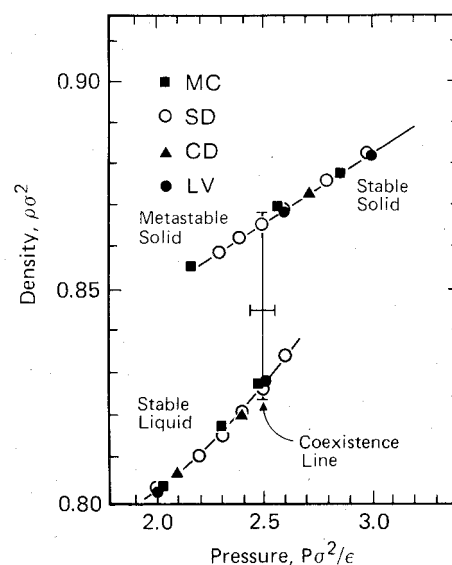
1.4 Esimerkkejä sovellutuksista

Tasapainoilmiöiden simulaatioista otetaan esimerkiksi Lennard-Jones-systeemin (LJ-systeemin) tilanyhtälö. LJ-systeemi koostuu atomeista, jotka vuorovaikuttavat LJ-potentiaalin välityksellä

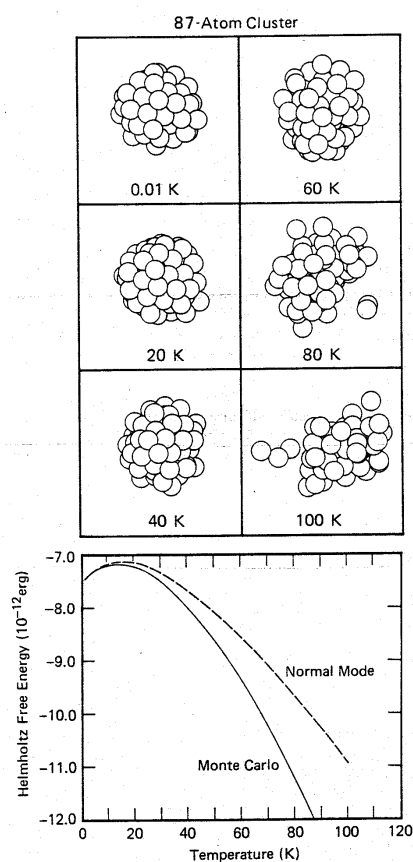
$$V_{LJ}(r_{ij}) = 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right] \quad (1.10)$$

Oheisessa kuvassa on esitetty eri simulaatiomenetelmillä lasketut tilanyhtälöt $\rho(P)$ [F. F. Abraham *Adv. Phys.* **35** (1986) 1.]. Paineen ja tiheyden yksikköinä on käytetty LJ-systeemin redusoituja yksiköitä. Kuvasta näkyvät selvästi eri olomuodot sekä kiteen ylikuumentuminen ja nesteen alijäähtyminen.

Faasitransitiosta olkoon esimerkkinä kaasun kondensoituminen ja sen klusterimalli. [F. F. Abraham, *J. Vac. Sci. Technol. B* **2** (1984) 534.]



Kuva 1.7: Lennard-Jones -systeemin tilanyhtälö. MC=Monte Carlo, SD,CD,LV=dynaamisia menetelmiä.



Kuva 1.8: Argonkaasun kondensoituminen.

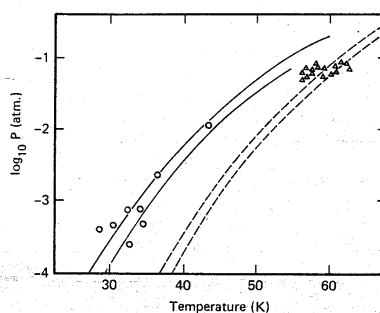


FIG. 4. Comparison of different calculations with experimental data for the onset of argon vapor condensation. Broken curves: classical theory; full curves: Monte Carlo work. In each case, the lower and upper curves are for a nucleation rate of 1 and $10^6 \text{ cm}^{-3} \text{ s}^{-1}$, respectively. Open triangles and full circle are for pure argon, and open circles are for argon in a helium carrier gas [from Garcia and Torroja (Ref. 16)].

Esimerkkinä kuljetusilmiöstä on gamma- ja elektronisäteilyn eteneminen väliaineessa. Analyyttisten ratkaisutapojen ongelmana on mm. sekundäärihiukkasten synty: elektroni-gammakaskadi.

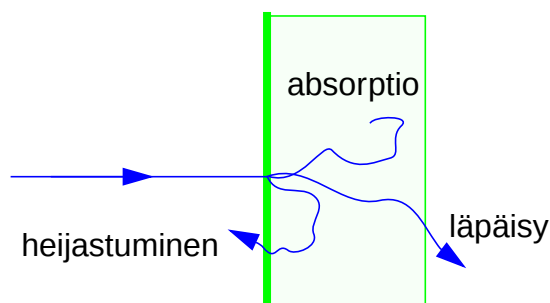
Ilmiön MC-simuloinnin periaate on seuraava:

Seurataan hiukasta väliaineessa vuorovaikutuksesta toiseen.

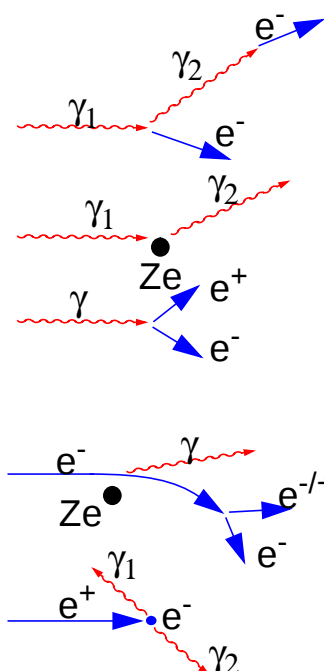
Vuorovaikutusten välimatkan todennäköisyysjakauma on $P(s) = e^{-s/\lambda}/\lambda$, missä hiukkasen keskimääräinen vapaa matka $\lambda \propto 1/\sigma$ ja σ on vuorovaikutuksen vaikutusala eli todennäköisyys.

Simuloinnissa generoidaan välimatkoja s ja jokaisen vuorovaikutuksen kohdalla arvotaan vuorovaikutusmekanismi (ks. oheinen kuva). Hiukkasen uusi suunta ja energia saadaan ko. vuorovaikutuksen differentiaalivaikutusalasta. Myös mahdollisen sekundäärihiukkasen tiedot otetaan muistiin.

Kuvassa 1.11 on esitetty muutamia MC-menetelmällä simuloituja elektronien ja niiden synnyttämien fotonien ratoja¹.

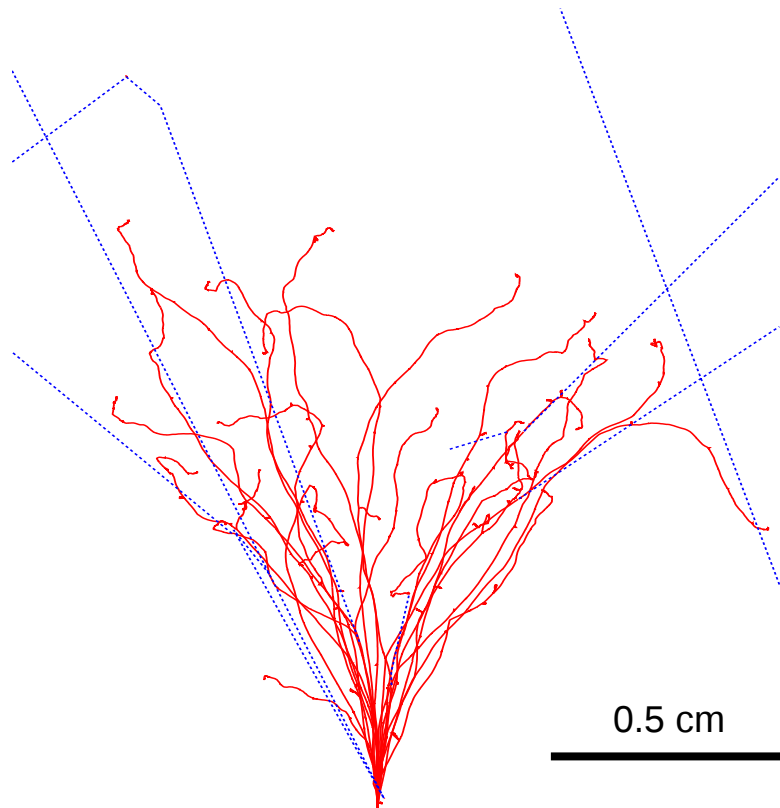


Kuva 1.9: Sätelilyn eteneminen väliaineessa.



Kuva 1.10: Gamma- ja elektronisäteilyn vuorovaikutusmekanismit.

1. Elektronin energia lienee muutama sata keV.

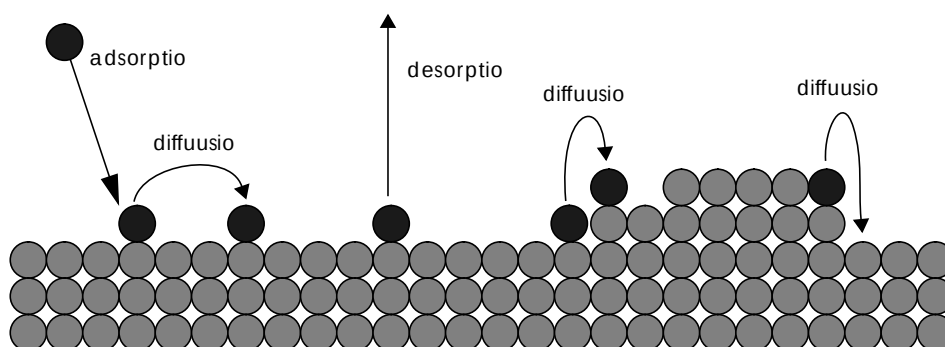


Kuva 1.11: Energeettisten (muutama sata keV) elektronien eteneminen vedessä Monte Carlo -simulaatiolla laskettuna.

Erilaisia kuljetus- ja kasvuilmiöitä voidaan mallintaa kineettisellä Monte Carlolla (KMC). Tärkeitä sovellutuksia ovat mm. diffuusio kiinteässä aineessa, kiteen kasvu, säteilyvaurioiden kulkeutuminen.

KMC perustuu siihen, että kuvaamme systeemiä sellaisella tasolla, että voimme erottaa tietyn joukon diskreettejä tapahtumia $\mathbf{E} = \{e_1, e_2, \dots, e_N\}$, joissa systeemi siirtyy tilasta toiseen. Lisäksi aikaskaala on sellainen, että mitkään tapahtumat eivät tapahdu samanaikaisesti.

Otetaan esimerkiksi kiteen kasvu kaasufaasista. Tapahtumia ovat: adsorptio, desorptio ja atomin hyppy pinnalla erilaisissa geometrioissa.



Kuva 1.12: Kiteen kasvatuksessa havaittavat tapahtumat.

Algoritmi tämän toteuttamiseksi voisi olla seuraavanlainen:

- i. Generoi tasaisesti jakautunut satunnaisluku $\xi \in [0, Q(C_k))$.
- ii. Valitse tätä vastaava tapahtuma: valitse ensimmäinen indeksi s , jolle pätee

$$\sum_{a=1}^s R_a(C_k) \geq \xi .$$
- iii. Etene uuteen konfiguraatioon C_{k+1} toteuttamalla tapahtuma s .
- iv. Päivitä niitä todennäköisyyksiä R_a , jotka ovat muuttuneet tapahtuman s seurauksena. Päivitä Q ja muut tarvittavat tietorakenteet.

Tässä C_k on systeemin tila simulointiaskeleella k , $Q(C_k)$ on kokonaistodennäköisyys aikayksikköä kohti, että systeemi siirtyy johonkin toiseen tilaan ja $R_a(C_k)$ tapahtuman a todennäköisyys aikayksikköä kohti systeemin tilassa C_k .

Lisäksi Monte Carlo -menetelmiä voidaan käyttää kvanttimekaanisten systeemien tutkimiseen (kvantti-Monte Carlo, quantum Monte Carlo QMC).

2 Statistista fysiikkaa

2.1 Faasiavaruus

Kertaamme seuraavassa lyhyesti kurssilla tarvittavia statistisen fysiikan käsitteitä. Pitäydymme klassisessa statistisessa fysiikassa.

Statistisessa fysiikassa on viimekädessä kyse siitä, miten aineen mikroskooppisesta kuvauksesta saadaan makroskooppisia (termodynaamisia, mitattavia suureita):

$\{\mathbf{q}, \mathbf{p}\}$ (tai $\{\mathbf{r}, \mathbf{p}\}$) $\rightarrow$ makroskooppiset suureet ,

missä $\mathbf{q} = \{\mathbf{q}_i\}$ ovat systeemin koordinaatit ja $\mathbf{p} = \{\mathbf{p}_i\}$ liikemäärät.

Systeemiä kuvaa Hamiltonin funktio (eli kokonaisenergia) $H(\mathbf{q}, \mathbf{p})$. Atomisysteemeille Hamiltonin funktio on useimmiten muotoa

$$H(\mathbf{q}, \mathbf{p}) = H(\mathbf{r}, \mathbf{p}) = \sum_{i=1}^N \frac{p_i^2}{2m_i} + V(\mathbf{q}) \quad (2.1)$$

Systeemin dynamiikkaa (aikakehitystä) kuvaavat Hamiltonin liikeyhtälöt¹

$$\dot{q}_k = \frac{\partial}{\partial p_k} H(\mathbf{q}, \mathbf{p}) \quad (2.2)$$

$$\dot{p}_k = -\frac{\partial}{\partial q_k} H(\mathbf{q}, \mathbf{p}) \quad (2.3)$$

Yksinkertaisissa tapauksissa Hamiltonin yhtälöistä saadaan Newtonin liikeyhtälöt

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i \quad , \quad \frac{d(m_i \mathbf{v}_i)}{dt} = \sum_{i \neq j} \mathbf{F}(r_{ij}) \quad (2.4)$$

Jos meillä on N hiukkasta, koko systeemin tilaa voidaan kuvata pisteellä Γ $6N$ -ulotteisessa faasiavaruudessa. Pisteiden kulun määräävät liikeyhtälöt.

Olemme luonnollisesti kiinnostuneita jonkin suureen $A = A(\Gamma)$ arvosta. Kokeellisesti pääsemme käsiksi keskiarvoon:

$$A_{\text{obs}} = \langle A \rangle_t = \langle A(\Gamma(t)) \rangle_t = \lim_{t_{\text{obs}} \rightarrow \infty} \frac{1}{t_{\text{obs}}} \int_0^{t_{\text{obs}}} A(\Gamma(t)) dt \quad (2.5)$$

1. Kuten olemme jo todenneet, aina ei mallissa ole liikeyhtälöitä.

Simulaatioissa tulisi pystyä laskemaan A_{obs} mikroskooppisesta mallista lähtien.

Statistisessa fysiikassa käytetään Gibbsin ensemblejä kuvaamaan systeemin todennäköisyyttä olla tietyssä faasiavaruuden pisteessä. Tällä tavalla pystytään helpommin laskemaan erilaisia asioita verrattuna siihen, että integroisimme liikeyhtälöitä. Olkoon meillä suuri määrä (ensemble) samanlaisia systeemejä (mutta erilaisissa mikroskooppissa tiloissa).

Todennäköisyystiheys $\rho_{\text{ens}}(\Gamma)d\Gamma$ on verrannollinen faasiavaruuden alkiossa $d\Gamma$ olevien systeemien lukumäärään. Tiheys ρ_{ens} on erilainen erilaisille ulkoisille olosuhteille. Koska piste Γ on yhtä kuin systeemi, $\rho_{\text{ens}}(\Gamma)$ riippuu ajasta.

Tiheydelle ρ_{ens} pätevät säilymislait

i. ρ käyttäytyy kuten kokoonpuristumaton neste.

ii. Systeemejä ei häviä eikä synny: $\frac{d\rho_{\text{ens}}}{dt} = 0$.

iii. Tasapainossa $\frac{\partial\rho_{\text{ens}}}{\partial t} = 0$.

Näistä voidaan johtaa tiheydelle liikeyhtälö

$$\frac{\partial}{\partial t}\rho_{\text{ens}}(\Gamma, t) = -iL\rho_{\text{ens}}(\Gamma, t) , \quad (2.6)$$

missä Liouvillen operaattori L on

$$iL = \sum_i \dot{\mathbf{r}}_i \cdot \nabla_{\mathbf{r}_i} + \sum_i \dot{\mathbf{p}}_i \cdot \nabla_{\mathbf{p}_i} \quad (2.7)$$

Statistisessa fysiikassa aikakeskiarvo siis korvataan ensemblekeskiarvolla

$$A_{\text{obs}} = \langle A \rangle_{\text{ens}} = \sum_{\Gamma} A(\Gamma)\rho_{\text{ens}}(\Gamma) \quad (2.8)$$

Eräs tärkeä ja kiistanalainenkin asia on ergodisuus. Jos systeemi on ergodinen, tietystä alkutilasta lähdettäessä käydään läpi (liikeyhtälöiden mukaan) kaikki faasiavaruuden pisteet, joissa ρ_{ens} on nolasta poikkeava.

Todennäköisyystiheydestä käytetään myös normittamatonta muotoa

$$\rho_{\text{ens}}(\Gamma) = \frac{w_{\text{ens}}(\Gamma)}{Q_{\text{ens}}}; \quad Q_{\text{ens}} = \sum_{\Gamma} w_{\text{ens}}(\Gamma) \quad (2.9)$$

$$\langle A \rangle_{\text{ens}} = \frac{\sum_{\Gamma} w_{\text{ens}}(\Gamma) A(\Gamma)}{\sum_{\Gamma} w_{\text{ens}}(\Gamma)} \quad (2.10)$$

Tuo normitustekijä eli partitiofunktio Q_{ens} riippuu systeemin makroskooppisista ominaisuuksista. Yhteys termodynamiikkaan saadaan määrittelemällä

$$\Psi_{\text{ens}} = -\ln Q_{\text{ens}} = \text{termodynaaminen potentiaali.} \quad (2.11)$$

Simuloinnin¹ tarkoitus on yleensä käydä läpi faasiavaruutta mahdollisimman tehokkaasti keskiarvojen A_{obs} laskemista varten

Kuten aikaisemmin mainittiin tähän tehtävään on kaksi lähestymistapaa

- i. molekyyliidynamiikka: $\langle A \rangle_t$
- ii. Monte Carlo: $\langle A \rangle_{\text{ens}}$ (importance sampling)

Tärkeimmät ensemblit eli ulkoiset olosuhteen statistisessa fysiikassa ovat

- i. mikrokanoninen: hiukkaslukumäärä, tilavuus ja sisäinen energia vakioita (NVE)
- ii. kanoninen (NVT): kuten edellä, mutta energian sijasta lämpötila vakio
- iii. isoterminen-isobaarinen (NPT): kuten edellä, mutta tilavuuden sijasta paine vakio
- iv. suurkanoninen (μ VT): kuten kanoninen ensemble, mutta hiukkaslukumäärän sijasta kemiallinen potentiaali vakio

(Huom: Seuraavassa symbolilla V merkitään systeemin tilavuutta ja symbolilla U systeemin hiukkasten välistä vuorovaikutuspotentiaalia.)

2.2 Ensemblit

Miten saadaan nuo tiheysfunktiot ρ_{ens} erilaisille ulkoisille olosuhteille?²

Gibbsin entropia määritellään

$$S = -k_B \int \rho_{\text{ens}}(\Gamma) \ln(C^N \rho_{\text{ens}}(\Gamma)) d\Gamma, \quad (2.12)$$

missä k_B on Boltzmannin vakio.

Mikrokanonisessa ensemblessä systeemin hiukkaslukumäärä, tilavuus ja kokonaisenergia ovat

1. Tarkemmin: tasapainosimulaation
 2. Katso esimerkiksi L. E. Reichl, *A Modern Course in Statistical Physics*.

vakioita (NVE-ensemble) . Tasapainotilassa entropia on maksimissaan ja tiheysfunktion on oltava normitettu:

$$\int_{H(\Gamma) = E} \rho_{NVE}(\Gamma) d\Gamma = 1 . \quad (2.13)$$

Käytetään Lagrangen kerrointa α_0 etsittäessä tiheysfunktiota:

$$\delta \left[\int_{H(\Gamma) = E} (\alpha_0 \rho_{NVE}(\Gamma) - k_B \rho_{NVE}(\Gamma) \ln(C^N \rho_{NVE}(\Gamma))) d\Gamma \right] = 0 . \quad (2.14)$$

Tästä edelleen

$$\int_{H(\Gamma) = E} (\alpha_0 - k_B \ln[C^N \rho_{NVE}(\Gamma)] - k_B) \delta \rho_{NVE}(\Gamma) d\Gamma = 0 , \quad (2.15)$$

ja koska variaatio $\delta \rho_{NVE}(\Gamma)$ on mielivaltainen, on integrandin hävittävä:

$$\alpha_0 - k_B \ln[C^N \rho_{NVE}(\Gamma)] - k_B = 0 . \quad (2.16)$$

Saamme siis

$$\rho_{NVE}(\Gamma) = \begin{cases} K, H(\Gamma) = E \\ 0, \text{muulloin} \end{cases} . \quad (2.17)$$

Normitusehdon huomioonottaen saadaan lopulta

$$\rho_{NVE}(\Gamma) = \begin{cases} \frac{1}{\Omega(E, V, N)}, H(\Gamma) = E \\ 0, \text{muulloin} \end{cases} \quad (2.18)$$

Kanonisen ensemblen tapauksessa energian sijasta lämpötila on vakio. Mukaan tulee siten lisäehto, että sisäisen energian keskiarvo pysyy vakiona:

$$\langle E \rangle = \int H(\Gamma) \rho_{NVT}(\Gamma) d\Gamma \quad (2.19)$$

Variaatioyhtälö tulee nyt muotoon (uusi Lagrangen kerroin α_E):

$$\delta \left[\int (\alpha_0 \rho_{NVE}(\Gamma) + \alpha_E H(\Gamma) \rho_{NVT}(\Gamma) - k_B \rho_{NVE}(\Gamma) \ln[C^N \rho_{NVE}(\Gamma)]) d\Gamma \right] = 0 \quad (2.20)$$

josta edelleen

$$\alpha_0 + \alpha_E H(\Gamma) - k_B \ln[C^N \rho_{NVT}(\Gamma)] - k_B = 0 \quad (2.21)$$

ja

$$\rho_{NVT}(\Gamma) = \frac{1}{C^N} \exp \left[\frac{\alpha_0}{k_B} - 1 + \frac{\alpha_E}{k_B} H(\Gamma) \right] . \quad (2.22)$$

Normalisaatiosta seuraa

$$Q_{NVT} \equiv \exp \left[1 - \frac{\alpha_0}{k_B} \right] = \frac{1}{C^N} \int \exp \left[\frac{\alpha_E}{k_B} H(\Gamma) \right] d\Gamma . \quad (2.23)$$

Seuraavaksi määritämme kertoimen α_E . Kertomalla yhtälö (2.21) ρ_{NVT} :llä ja integroimalla saadaan

$$\begin{aligned} (\alpha_0 - k_B) \int \rho_{NVT}(\Gamma) d\Gamma + \alpha_E \int H(\Gamma) \rho_{NVT}(\Gamma) d\Gamma \\ - k_B \int \rho_{NVT}(\Gamma) \ln[C^N \rho_{NVT}(\Gamma)] d\Gamma = 0 \end{aligned} \quad (2.24)$$

jonka voimme kirjoittaa muotoon

$$-k_B \ln Q_{NVT} + \alpha_E \langle E \rangle + S = 0 . \quad (2.25)$$

Helmholtzin vapaa energia määritellään

$$A - U + ST = 0 , \quad (2.26)$$

joten voimme tehdä seuraavat identifiikaatiot

$$\alpha_E = -\frac{1}{T} , \quad A = -k_B T \ln Q_{NVT} , \quad (2.27)$$

missä partitiofunktio on nyt

$$Q_{NVT} = \frac{1}{C^N} \int e^{-\beta H(\Gamma)} d\Gamma \quad (\beta = (k_B T)^{-1}) \quad (2.28)$$

ja tiheysfunktio

$$\rho_{NVT}(\Gamma) = \frac{1}{C^N Q_{NVT}} e^{-\beta H(\Gamma)} . \quad (2.29)$$

Samaan tapaan saadaan muiden ensembleiden tiheydet. Metropoliksen MC-algoritmi kehitettiin alunperin kanonista ensembleä varten, mutta on helposti yleistettävissä muihinkin ensemb-

leihin.

Listataan lopuksi vielä eri ensembleiden tiheysfunktiot ja termodynaamiset potentiaalit¹.

- **Mikrokanoninen** ensemble: NVE vakioita (eristetty)

$$\rho_{NVE}(\Gamma) \propto \delta(H(\Gamma) - E) \quad (2.30)$$

$$Q_{NVE} = \sum_{\Gamma} \delta(H(\Gamma) - E) = \frac{1}{C^N} \int d\mathbf{r} d\mathbf{p} \delta(H(\mathbf{r}, \mathbf{p}) - E) \quad (2.31)$$

Termodynaaminen potentiaali on nyt **entropia**:

$$\frac{S}{k_B} = \ln Q_{NVE} \quad (2.32)$$

- **Kanoninen** ensemble : NVT vakioita (suljettu)

$$\rho_{NVT}(\Gamma) \propto \exp(-H(\Gamma)/k_B T) \quad (2.33)$$

$$Q_{NVT} = \sum_{\Gamma} \exp(-H(\Gamma)/k_B T) = \frac{1}{C^N} \int d\mathbf{r} d\mathbf{p} \exp(-H(\mathbf{r}, \mathbf{p})/k_B T) \quad (2.34)$$

Termodynaaminen potentiaali on **Helmholtzin vapaa energia**:

$$\frac{A}{k_B T} = -\ln Q_{NVT} \quad (2.35)$$

- **Isotermien-isobaarinen** ensemble: NPT vakioita

$$\rho_{NPT}(\Gamma) \propto \exp((-H(\Gamma) + PV)/k_B T) \quad (2.36)$$

$$\begin{aligned} Q_{NPT} &= \sum_{\Gamma} \exp((-H(\Gamma) + PV)/k_B T) \\ &= \frac{1}{C^N} \frac{1}{V_0} \int d\mathbf{r} d\mathbf{p} \exp((-H(\mathbf{r}, \mathbf{p}) + PV)/k_B T) \end{aligned} \quad (2.37)$$

Termodynaaminen potentiaali on **Gibbsin vapaa energia**:

$$\frac{G}{k_B T} = -\ln Q_{NPT} \quad (2.38)$$

1. Vakio $C^N = h^{3N}$ erilaisille hiukkasille ja $N!h^{3N}$ identtisille.

- **Suurkanoninen** ensemble: μVT vakioita

$$\rho_{\mu VT}(\Gamma) \propto \exp((-H(\Gamma) + \mu N)/k_B T) \quad (2.39)$$

$$\begin{aligned} Q_{\mu VT} &= \sum_{\Gamma, N} \exp((-H(\Gamma) + \mu N)/k_B T) \\ &= \sum_N \frac{1}{C^N} \exp(-\mu N/k_B T) \int d\mathbf{r} d\mathbf{p} \exp(-H(\mathbf{r}, \mathbf{p})/k_B T) \end{aligned} \quad (2.40)$$

Termodynaaminen potentiaali on **suuri potentiaali**:

$$\frac{-\Omega}{k_B T} = -\ln Q_{\mu VT} \quad (2.41)$$

Jos systeemin Hamiltonin funktion voidaan jakaa koordinaateista ja liikemääristä riippuviin osiin

$$H(\mathbf{r}, \mathbf{p}) = K(\mathbf{p}) + U(\mathbf{r}) , \quad (2.42)$$

voimme integroida pois liikemääräosuuden:

$$Q_{NVT} = \frac{1}{C^N} \int d\mathbf{p} \exp(-K(\mathbf{p})/k_B T) \int d\mathbf{r} \exp(-U(\mathbf{r})/k_B T) \quad (2.43)$$

$$Q_{NVT} = Q_{NVT}^{\text{id}} Q_{NVT}^{\text{ex}} , \quad (2.44)$$

Tässä ideaalikaasuosuus on

$$Q_{NVT}^{\text{id}} = \frac{V^N}{N! \Lambda^{3N}} ; \Lambda = \left(\frac{h^2}{2\pi m k_B T} \right)^{1/2} \quad (2.45)$$

ja hiukkasten vuorovaikutuksesta aiheutuva osuus on

$$Q_{NVT}^{\text{ex}} = V^{-N} \int d\mathbf{r} \exp(-U(\mathbf{r})/k_B T) . \quad (2.46)$$

Metropolis-MC:llä lasketaan vain koordinaateista riippuva osa (konfiguraatio-osa) Q_{NVT}^{ex} .

2.3 Makroskooppisten suureiden laskeminen

Simulaatiossa on tarkoituksena siis laskea makroskooppisia (termodynaamisia) suureita systeemin mikroskooppisten ominaisuuksien avulla. Seuraavassa on lyhyesti esitetty tärkeimpien suureiden laskeminen lähinnä atomeista koostuville systeemeille.

Systeemin sisäinen energia on sen kokonaisenergian keskiarvo

$$E = \langle H \rangle = \langle K \rangle + \langle U \rangle = \left\langle \sum_i \frac{|\mathbf{p}_i|^2}{2m_i} \right\rangle + \langle U(\mathbf{q}) \rangle . \quad (2.47)$$

Lämpötila taas on kineettisen energian keskiarvo

$$T = \frac{2\langle K \rangle}{3Nk_B} = \frac{1}{3Nk_B} \left\langle \sum_{i=1}^N \frac{|\mathbf{p}_i|^2}{2m_i} \right\rangle . \quad (2.48)$$

Systeemin paine P taas saadaan ns. viriaalin avulla

$$PV = Nk_B T + \langle W \rangle , \quad (2.49)$$

missä viriaali W on

$$W = \frac{1}{3} \sum_{i=1}^N \mathbf{r}_i \cdot \mathbf{f}_i , \quad (2.50)$$

ja $\mathbf{f}_i$ on hiukkaseen i kohdistuva voima.

Ylläolevat yhtälöt voidaan johtaa ns. yleistetystä ekvipartitioteoreemasta¹

$$\left\langle p_k \frac{\partial H}{\partial p_k} \right\rangle = k_B T , \quad \left\langle q_k \frac{\partial H}{\partial q_k} \right\rangle = k_B T . \quad (2.51)$$

Kuten myöhemmin tulemme huomaamaan, ei tasapainoilmiöiden Monte Carlo -simulaatioissa ole ollenkaan mukana hiukkasten liikemääriä, joten lämpötilaa ei voi laskea. Toisaalta, jos tutkimme ensembleä, jossa lämpötila on vakio, on se ennalta annettu parametri; siis simulaation syöttötieto.

Erilaiset vastefunktiot kertovat, miten systeemi reagoi tietyn tilamuuttujan muutokseen. Tärkein näistä lienee vakio-tilavuuslämpökapasiteetti C_V . Sehän määritellään sisäisen energian lämpötiladerivaattana

1. Ks. Allen-Tildesley kappale 2.4

$$C_V(T) = \left(\frac{\partial E}{\partial T} \right)_V . \quad (2.52)$$

Simulaatioilla C_V :n laskemisen voisi toteuttaa tekemällä useita ajoja eri lämpötiloilla ja integroimalla yhtälö (2.52). Toisaalta kanonisessa ensemblessä fluktuaatioilla ja vastefunktioilla on yhteys¹, josta voimme lämpökapasiteetin laskea

$$\langle \delta H^2 \rangle = k_B T^2 C_V , \quad (2.53)$$

missä

$$\langle \delta H^2 \rangle = \langle H^2 \rangle - \langle H \rangle^2 . \quad (2.54)$$

Tämän fluktuaatio-vastefunktio -yhteyden avulla voidaan monia muitakin vastefunktioita laskea; esimerkiksi lämpölaajenemiskerroin, isoterminen puristuvuus jne.

Termodynaamisten potentiaalien (eli vapaiden energioiden) laskemista tarvitaan monessa yhteydessä. Esimerkiksi kiinteän aineen sulamispisteen saa selville, jos pystyy laskemaan kiteisen ja nestemäisen rakenteen Helmholtzin vapaat energiat. Potentiaalien laskeminen ei ole kuitenkaan kovin helppoa. Esimerkiksi Helmholtzin vapaan energian potentiaalienergiasta riippuva osa² voidaan lausua muodossa

$$\exp\left(\frac{A^{\text{ex}}}{k_B T}\right) = \frac{1}{Q_{NVT}^{\text{ex}}} = \langle \exp(U/k_B T) \rangle . \quad (2.55)$$

Toisaalta kanonisessa ensemblessä todennäköisimmät tilat ovat sellaisia, joille $\exp(-U/k_B T)$ on suuri, joten suora keskiarvon laskeminen on tehotonta.

Käyttökelpoinen tapa laskea energiaeroja on integrointi reversiibeliä reittiä pitkin. Esimerkiksi

$$\left(\frac{A}{T}\right)_2 - \left(\frac{A}{T}\right)_1 = - \int_{T_1}^{T_2} \left(\frac{E}{T^2}\right) dT \quad (2.56)$$

tai

$$\left(\frac{A}{T}\right)_2 - \left(\frac{A}{T}\right)_1 = - \int_{V_1}^{V_2} \left(\frac{P}{T}\right) dV . \quad (2.57)$$

Toinen tapa on lähteä idealisoidusta mallista, jonka vapaa energia pystytään laskemaan eksaktisti. Olkoon systeemin potentiaalienergia riippuvainen parametrilla λ : $U = U(\mathbf{r}, \lambda)$. Tällöin saamme seuraavan yhteyden

1. AT , kappale 2.5

2. $A = A^{\text{id}} + A^{\text{ex}}$, ks. kaavat (2.35) ja (2.45).

$$\begin{aligned}
\frac{\partial A}{\partial \lambda} &= -k_B T \frac{\partial}{\partial \lambda} [\ln \int d\mathbf{r} \exp(-U(\mathbf{r}, \lambda)/k_B T)] \quad . \\
&= \frac{\int d\mathbf{r} \frac{\partial U}{\partial \lambda} \exp(-U/k_B T)}{\int d\mathbf{r} \exp(-U/k_B T)} = \left\langle \frac{\partial U}{\partial \lambda} \right\rangle
\end{aligned}
\tag{2.58}$$

Vapaan energian absoluuttiarvo on laskettavissa, jos λ :n avulla voimme kuvata systeemiä, jonka A on laskettavissa (ideaalikaasu, harmoninen kide):

$$A(\lambda) - A(\lambda_0) = \int_{\lambda_0}^{\lambda} \left\langle \frac{\partial U}{\partial \lambda} \right\rangle d\lambda \tag{2.59}$$

Otetaan esimerkiksi harmoninen kide (Einsteinin malli)

$$U(\mathbf{r}, \lambda) = U_0(\mathbf{r}) + \lambda \sum_{i=1}^N (\mathbf{r}_i - \mathbf{r}_{i0})^2 . \tag{2.60}$$

Kun $\lambda = 0$, systeemi on alkuperäinen, ja kun λ kasvaa lähestyy systeemi harmonista kidettä. Vapaa energia saadaan integraalina

$$A(\lambda = 0) = A(\lambda) - \int_0^{\lambda} \left\langle \frac{\partial U}{\partial \lambda} \right\rangle d\lambda' . \tag{2.61}$$

Helmholtzin vapaa energia suurella λ :n arvolla voidaan laskea tarkasti

$$A(\lambda = 0) = \frac{3N\hbar\omega}{2} - 3Nk_B T \ln(1 - e^{-\hbar\omega/k_B T}) + O(1/\lambda) \tag{2.62}$$

3 Todennäköisyyslaskentaa

3.1 Yleistä

Seuraavassa käydään läpi kurssiin liittyvää todennäköisyyslaskentaa.

Todennäköisyyden avulla voimme kuvata enemmän tai vähemmän kvantitatiivisesti jonkin tapahtuman tai 'kokeen' odotettavissa olevaa tulosta.

Jos tapahtuman A todennäköisyys on $P(A)$, voimme odottaa, että N :n identtisen kokeen tuloksena saamme $NP(A)$ kappaletta tapahtumia A . Rajalla $N \rightarrow \infty$, tapahtumien A osuus lähestyy arvoa $P(A)$.

Kokeen *otosavaruus* (*sample space*) S on kokeen mahdollisten tulosten joukko. Siis jokainen kokeen tulos vastaa yhtä tai useampaa joukon alkia (otosavaruuden pistettä). *Koe* tai *tapahtuma* on S :n osajoukko.

Todennäköisyys, että saadaan joko tulos A tai B on

$$P(A \cup B) = P(A) + P(B) - P(A \cap B), \quad (3.1)$$

missä $P(A \cap B)$ on todennäköisyys, että saadaan sekä A että B .

Jos tapahtumat $A_1, A_2, \dots, A_m$ ovat toisensa poissulkevia ja lisäksi A_i :t jakavat S :n osiin:

$$A_1 \cup A_2 \cup \dots \cup A_m = S, \quad (3.2)$$

niin pätee

$$P(A_1) + P(A_2) + \dots + P(A_m) = 1$$

Tapahtumat A ja B ovat riippumattomia, jos

$$P(A \cap B) = P(A)P(B). \quad (3.3)$$

Ehdollinen todennäköisyys on todennäköisyys, että tapahtuma A toteutuu ehdolla, että myös tapahtuma B toteutuu. Se määritellään

$$P(B|A) = \frac{P(A \cap B)}{P(A)}. \quad (3.4)$$

Koska $P(A \cap B) = P(B \cap A)$ pätee myös

$$P(A)P(A|B) = P(B)P(B|A). \quad (3.5)$$

Jos A ja B ovat riippumattomia

$$P(B|A) = P(A) . \quad (3.6)$$

Suure, jonka arvon määrää edelläesitellyn kokeen tulos on satunnaismuuttuja tai stokastinen muuttuja. Otosavaruuden S satunnaismuuttuja X on funktio, joka kuvaa S :n alkioita reaalilukujoukolle.

Jokaisessa kokeessa muuttuja X voi saada jonkin arvon joukosta $\{x_i\}$. Pari esimerkkiä X :stä:

- a) kruunujen lukumäärä kolmen kolikon heiton jälkeen
- b) noppien silmälukujen maksimi, neljän nopan heiton jälkeen.

Olkoon X stokastinen muuttuja avaruudessa S . Olkoot sallitut arvot $X(S) = \{x_1, x_2, \dots\}$. Voimme tehdä $X(S)$:stä otosavaruuden antamalla jokaiselle x_i :lle todennäköisyyden. Nämä todennäköisyydet $f(x_i)$ määrittelevät S :n todennäköisyysjakauman ja niille pätee

$$f(x_i) \geq 0 \quad (3.7)$$

$$\sum_i f(x_i) = 1 . \quad (3.8)$$

Usein meillä on tietoa vain jakauman f momenteista:

$$\langle X^n \rangle = \sum_i x_i^n f(x_i) . \quad (3.9)$$

Ensimmäinen momentti $\langle X \rangle$ on keskiarvo ja jakauman standardipoikkema on

$$\sigma_X \equiv (\langle X^2 \rangle - \langle X \rangle^2)^{1/2} . \quad (3.10)$$

Stokastinen muuttuja X voi tietysti saada myös jatkuvia arvoja. Esimerkiksi reaalilukuakselin väli $a \leq X \leq b$ voi vastata yhtä tapahtumaa. Todennäköisyysjakauma on sellainen paloittain jatkuva funktio, että tapahtuman $a \leq X \leq b$ todennäköisyys on

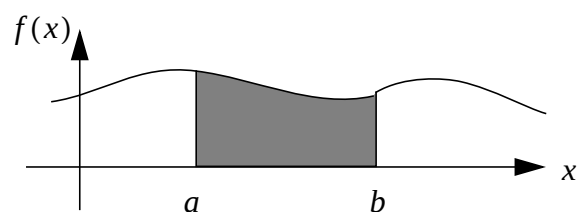
$$P(a \leq X \leq b) = \int_a^b f_X(x) dx . \quad (3.11)$$

Lisäksi jakaumafunktio toteuttaa ehdot

$$f_X(x) \geq 0 \quad (3.12)$$

ja

$$\int f_X(x) dx = 1 . \quad (3.13)$$



Kuva 3.1: Todennäköisyysjakauma

Vastaavasti momentit määritellään

$$\langle X^n \rangle = \int x^n f_X(x) dx . \quad (3.14)$$

Jos tunnemme f_X :n kaikki momentit, tunnemme jakauman täysin. Tämä voidaan osoittaa ns. karakteristisen funktion $\phi_X(k)$ avulla:

$$\phi_X(k) = \langle e^{ikX} \rangle = \int e^{ikx} f_X(x) dx = \sum_{n=0}^{\infty} \frac{(ik)^n \langle X^n \rangle}{n!} . \quad (3.15)$$

Todennäköisyyksiä on karakteristisen funktion Fourier-muunnos:

$$f_X(x) = \frac{1}{2\pi} \int e^{-ikx} \phi_X(k) dk . \quad (3.16)$$

Vastaavasti jakauman momentit saadaan karakteristisen funktion derivaattoina:

$$\langle X^n \rangle = \frac{1}{i^n} \left[\frac{d^n}{dk^n} \phi_X(k) \right]_{k=0} . \quad (3.17)$$

Stokastisia muuttujia voi olla useampiakin. Olkoon meillä muuttujat $X(S) = \{x_1, x_2, \dots\}$ ja $Y(S) = \{y_1, y_2, \dots\}$. Näiden tulojoukko $X(S) \times Y(S) = \{(x_1, y_1), (x_1, y_2), \dots, (x_i, y_j), \dots\}$ muodostaa nyt otosavaruuden, kun määrittelemme parin $\{x_i, y_j\}$ todennäköisyydeksi

$$P(X = x_i, Y = y_j) = f(x_i, y_j) . \quad (3.18)$$

Muuttujien kovarianssi määritellään

$$\begin{aligned} \text{cov}(X, Y) &= \int (x - \langle X \rangle)(y - \langle Y \rangle) f(x, y) dx dy . \\ &= \int xy f(x, y) dx dy - \langle X \rangle \langle Y \rangle \\ &= \langle XY \rangle - \langle X \rangle \langle Y \rangle \end{aligned} \quad (3.19)$$

ja korrelaatio

$$\text{cor}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} . \quad (3.20)$$

Korrelaatiolla on seuraavat ominaisuudet

- (i) $\text{cor}(X, Y) = \text{cor}(Y, X)$,
- (ii) $-1 \leq \text{cor}(X, Y) \leq 1$,

$$\begin{aligned}
\text{(iii)} \quad \text{cor}(X, X) &= 1, \text{cor}(X, -X) = -1, \\
\text{(iv)} \quad \text{cor}(aX + b, cY + d) &= \text{cor}(X, Y), \text{ jos } a, c \neq 0.
\end{aligned} \tag{3.21}$$

Jos muuttujat X ja Y ovat riippumattomia pätevät seuraavat relaatiot

$$\begin{aligned}
\text{(i')} \quad f(x, y) &= f_X(x) f_Y(y), \\
\text{(ii')} \quad \langle XY \rangle &= \langle X \rangle \langle Y \rangle, \\
\text{(iii')} \quad \langle (X + Y)^2 \rangle - \langle X + Y \rangle^2 &= \langle X^2 \rangle - \langle X \rangle^2 + \langle Y^2 \rangle - \langle Y \rangle^2, \\
\text{(iv')} \quad \text{cov}(X, Y) &= 0.
\end{aligned} \tag{3.22}$$

3.2 Tärkeimmät todennäköisyysjakaumat

Usein meillä on kyseessä tilanne, jossa on suuri määrä N kokeita, joilla jokaisella on kaksi mahdollista tulosta ($+1$ ja -1). Esimerkiksi hiukkanen joko siroaa tietyllä matkalla tai sitten ei.

Olkoot todennäköisyydet tuloksille p ja q . Selvästikin $p + q = 1$. Todennäköisyys, että N :n kokeen tuloksena on n_1 kertaa $+1$ ja n_2 kertaa -1 on

$$P_N(n_1) = \frac{N!}{n_1! n_2!} p^{n_1} q^{n_2}. \tag{3.23}$$

Tämä on ns. binomijakauma. Sen keskiarvo ja standardipoikkeama ovat

$$\langle n_1 \rangle = pN, \quad \sigma_N^2 = Npq. \tag{3.24}$$

Binomijakaumasta saadaan rajalla $N \rightarrow \infty$ ja $pN \rightarrow \infty$ (siis p ei ole kovin pieni) Gaussin jakauma

$$P_N(n_1) = \frac{1}{\sigma_N \sqrt{2\pi}} \exp \left[-\frac{1}{2} \frac{(n_1 - \langle n_1 \rangle)^2}{\sigma_N^2} \right], \tag{3.25}$$

missä

$$\sigma_N = \sqrt{Npq}. \tag{3.26}$$

Gaussin jakauman määräävät kaksi ensimmäistä momenttia $\langle n_1 \rangle$ ja σ_N .

Taasen rajalla $N \rightarrow \infty$ ja $p \rightarrow 0$ siten, että $Np = a \ll N$ (missä a on äärellinen vakio) binomijakaumaa voidaan approksimoida Poissonin jakaumalla

$$P_N(n_1) = \frac{a^{n_1} e^{-a}}{n_1!} . \quad (3.27)$$

Poissonin jakauman määrää ensimmäinen momentti

$$\langle n_1 \rangle = a . \quad (3.28)$$

Esimerkkinä binomijakauman käytöstä olkoon yksiulotteinen satunnaiskävelijä. Todennäköisyys, että kävelijä ottaa askeleen vasemmalle on $p = 1/2$ ja oikealle $q = 1/2$. Askelten lukumäärä on N ja vasemmalle suuntautuneiden askelten lukumäärä on n_1 ja oikealle n_2 . Poikkeama origosta on siis $m = n_1 - n_2$. Koska $N = n_1 + n_2$ on $m = 2n_1 - N$. Suurilla N :n arvoilla poikkeaman tn.jakauma saadaan sijoittamalla $n_1 = (m + N)/2$ Gaussin jakaumaan (3.25):

$$P_N(m) = \left(\frac{2}{\pi N} \right)^{1/2} e^{-m^2/2N} . \quad (3.29)$$

Gaussin jakauman tärkeys perustuu ns. *keskeisraja-arvoteoreemaan (central limit theorem)*.

Haluamme tietää, satunnaismuuttujan Y jakaumasta. Y :n arvot vastaavat N :ää muuttujan X mittausta:

$$y_N = \frac{x_1 + x_2 + \dots + x_N}{N} . \quad (3.30)$$

Tutkitaan todennäköisyysjakaumaa $f_Y(y_N - \langle X \rangle)$. Sen karakteristinen funktio voidaan kirjoittaa muodossa

$$\begin{aligned} \Phi(k) &= \int e^{ik(y_N - \langle X \rangle)} f_Y(y_N - \langle X \rangle) dy_N \\ &= \int e^{i(k/N)((x_1 - \langle X \rangle) + \dots + (x_N - \langle X \rangle))} f_X(x_1) \dots f_X(x_N) dx_1 \dots dx_N \\ &= \left[\phi\left(\frac{k}{N}\right) \right]^N \end{aligned} \quad (3.31)$$

Jos $\sigma^2 = \langle X^2 \rangle - \langle X \rangle^2$, niin

$$\phi\left(\frac{k}{N}\right) = \int e^{i(k/N)(x_1 - \langle X \rangle)} f_X(x_1) dx_1 = 1 - \frac{1}{2} \frac{k^2}{N^2} \sigma^2 + \dots \quad (3.32)$$

Tämä funktio pienenee nopeasti k :n kasvaessa, johtuen oskillaatiotermitä $e^{i(k/N)x_1}$, ja $[\phi(k/N)]^N$ pienenee tätäkin nopeammin. Olettaen, että jakauman $f_X(x_1)$ momentit ovat äärellisiä, saadaan

$$\Phi(k) = \left[1 - \frac{1}{2} \frac{k^2}{N^2} \sigma^2 + O\left(\frac{k^3}{N^3}\right) \right]^N \rightarrow e^{-k^2 \sigma^2 / 2N} , \text{ kun } N \rightarrow \infty . \quad (3.33)$$

Tästä saamme lopulta todennäköisyystiheydeksi

$$\begin{aligned}
 f_Y(y_N - \langle X \rangle) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-ik(y_N - \langle X \rangle)} \Phi(k) dk \\
 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-ik(y_N - \langle X \rangle)} e^{-k^2 \sigma^2 / 2N} dk = \sqrt{\frac{N}{2\pi\sigma^2}} \exp\left[-\frac{N(y_N - \langle X \rangle)^2}{2\sigma^2}\right]
 \end{aligned} \quad (3.34)$$

Tämä on tärkeä tulos, sillä sen mukaan jakaumasta $f_X(x)$ riippumatta tulos on gaussinen. Ehtona on $f_X(x)$:n momenttien olemassaolo.

Toinen tärkeä teoreema on *suurten lukujen laki* (*law of large numbers*)¹. Sen mukaan tapahtumien A (todennäköisyys p) osuus N :stä riippumattomasta kokeesta lähestyy arvoa p , kun $N \rightarrow \infty$.

3.3 Ajasta riippuvat ilmiöt

Tarkastellaan nyt, miten todennäköisyysjakaumat kehittyvät ajan mukana eli tarkastellaan stokastista prosessia. ‘Ajan’ ei välttämättä tarvitse vastata reaalimaailman aikaa.

Olkoon meillä systeemi, jota kuvataan yhdellä stokastisella muuttujalla Y . Määritellään seuraavat todennäköisyydet

$$P_1(y_1, t_1) = \text{tn. että } Y \text{:llä on arvo } y_1 \text{ ajanhetkellä } t_1.$$

$$P_2(y_1, t_1; y_2, t_2) = \text{tn. että } Y \text{:llä on arvo } y_1 \text{ ajanhetkellä } t_1 \text{ ja } y_2 \text{ ajanhetkellä } t_2.$$

Vastaavasti voidaan määritellä $P_n(y_1, t_1; y_2, t_2, \dots, y_n, t_n)$.

Luonnollisesti näillekin jakaumille pätevät ehdot $P_n \geq 0$ ja $\int P_n(y_1, t_1; y_2, t_2, \dots, y_n, t_n) dy_1 dy_2 \dots dy_n = 1$.

Voimme määritellä ajasta riippuvia momentteja

$$\langle y_1(t_1) \dots y_n(t_n) \rangle = \int y_1 \dots y_n P_n(y_1, t_1, \dots, y_n, t_n) dy_1 \dots dy_n, \quad (3.35)$$

jotka kertovat muuttujan Y korrelaatioista eri ajanhetkillä.

Prosessi on *stationäärinen*, jos pätee

$$P_n(y_1, t_1, \dots, y_n, t_n) = P_n(y_1, t_1 + \tau, \dots, y_n, t_n + \tau) \quad (3.36)$$

1. L.E. Reichl, *A Modern Course in Statistical Physics*, kappale 5H.

kaikille $n:n$ ja $\tau:n$ arvoille. Tällöin

$$P_1(y_1, t_1) = P_1(y_1) , \quad (3.37)$$

ja $\langle y_1(t_1)y_2(t_2) \rangle$ riippuu vain aikaerosta $|t_1 - t_2|$.

Määritellään myös ehdollinen todennäköisyys

$$P_{1|1}(y_1, t_1 | y_2, t_2) = \text{tn.}, \text{ että } Y:\text{llä on arvo } y_2 \\ \text{ ajanhetkellä } t_2, \text{ olettaen että} \\ \text{ sillä oli arvo } y_1 \text{ hetkellä } t_1 .$$

$P_{1|1}$ saadaan määritelmästä

$$P_1(y_1, t_1)P_{1|1}(y_1, t_1 | y_2, t_2) = P_2(y_1, t_1; y_2, t_2) . \quad (3.38)$$

Tästä voimme johtaa yhteyden todennäköisyystiheyksille eri aikoina

$$P_1(y_2, t_2) = \int P_1(y_1, t_1)P_{1|1}(y_1, t_1 | y_2, t_2)dy_1 . \quad (3.39)$$

Ehdollinen todennäköisyys voidaan tietysti yleistää:

$$P_{k|l}(y_1, t_1; \dots; y_k, t_k | y_{k+1}, t_{k+1}, \dots, y_n, t_n) \quad . \quad (3.40) \\ = \frac{P_{k+l}(y_1, t_1; \dots; y_{k+l}, t_{k+l})}{P_k(y_1, t_1; \dots; y_k, t_k)}$$

Jos stokastisen prosessi muistaa vain lähimenneisyytensä, silloin todennäköisyys $P_{n-1|1}(y_1, t_1; \dots; y_{n-1}, t_{n-1} | y_n, t_n)$ (missä $t_1 < t_2 < \dots < t_n$) redusoituu muotoon

$$P_{n-1|1}(y_1, t_1; \dots; y_{n-1}, t_{n-1} | y_n, t_n) = P_{1|1}(y_{n-1}, t_{n-1} | y_n, t_n) . \quad (3.41)$$

Eli systeemin tilan ajanhetkellä t_n määrää vain aikaisempi tila hetkellä t_{n-1} . Todennäköisyys $P_{1|1}(y_1, t_1; y_2, t_2)$ on *siirtymätodennäköisyys*.

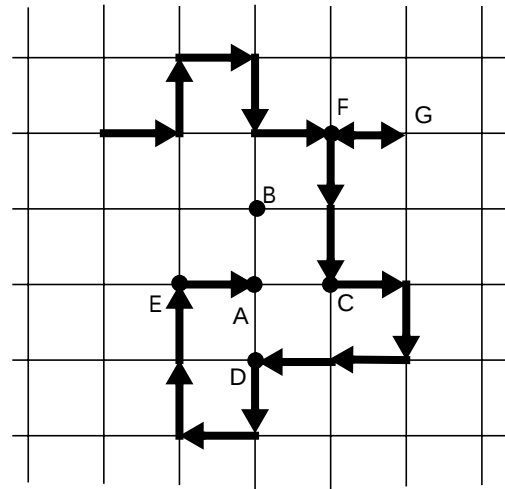
Tällaista prosessia kutsutaan *Markovin prosessiksi*. Sen voisi ajatella olevan liikeyhtälöiden stokastinen vastine.

Kaikkia prosesseja ei kuitenkaan voi kuvata Markovin prosessina. Otetaan kaksi esimerkkiä.

Normaali (hila)satunnaiskävely (*random walk*) on Markovin prosessi: Seuraava paikka pisteen A jälkeen määräytyy vain A:sta. Mikä tahansa paikoista B, C, D tai E on sallittu.

Itseään välttävä satunnaiskävely (*self avoiding random walk*) taas ei ole. Kävelijä ei saa mennä paikkaan, jossa se on jo ollut. Kuvan tapauksessa on sallittu vain paikka B. Lisäksi siirtymä F—G—F on kielletty. Tällaisella mallilla on tutkittu esimerkiksi pitkien taipuisien molekyylien ominaisuuksia.

Markovin prosessin määräävät siis kaksi funktiota $P_1(y_1, t_1)$ ja $P_{1|1}(y_1, t_1|y_2, t_2)$. Näistä voimme muodostaa koko hierarkian tiheysfunktioita, esim.



Kuva 3.2: Satunnaiskävelijä

$$\begin{aligned} P_3(y_1, t_1; y_2, t_2; y_3, t_3) &= P_2(y_1, t_1; y_2, t_2) P_{2|1}(y_1, t_1; y_2, t_2 | y_3, t_3) \\ &= P_1(y_1, t_1) P_{1|1}(y_1, t_1 | y_2, t_2) P_{1|1}(y_2, t_2 | y_3, t_3) \end{aligned} \quad (3.42)$$

Otetaan toinen yksinkertainen esimerkki. Todennäköisyys, että tietokone on kunnossa riippuu edellisen päivän tilanteesta. Jos kone oli kunnossa eilen on se tänäänkin 60% todennäköisyydellä. Jos se oli eilen alhaalla on se tänäänkin 70% todennäköisyydellä.

On siis kaksi mahdollista tilaa (tila-avaruus)

- 1 = kone käytössä
- 2 = kone alhaalla .

Siirtymätodennäköisyydet voidaan esittää matriisina

$$\mathbf{Q} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} = \begin{pmatrix} 0.6 & 0.4 \\ 0.3 & 0.7 \end{pmatrix}, \quad (3.43)$$

eli Q_{ij} kertoo siirtymätodennäköisyyden (aikayksikköä kohti) tilasta i tilaan j . Matriisille $\mathbf{Q}$ pätee tietenkin

$$\sum_j Q_{ij} = 1. \quad (3.44)$$

Tästä syystä sitä kutsutaan **stokastiseksi matriisiksi**.

Oletetaan, että alussa kone voi olla molemmissa tiloissa samalla todennäköisyydellä eli toden-

näköisyyksiä vektorina on

$$\mathbf{P}^{(1)} = \begin{pmatrix} 0.5 & 0.5 \end{pmatrix}. \quad (3.45)$$

Seuraavan päivän tilanne saadaan siirtymätodennäköisyysmatriisin avulla (ks. kaava (3.39)):

$$\mathbf{P}^{(2)} = \mathbf{P}^{(1)}\mathbf{Q} = \begin{pmatrix} 0.45 & 0.55 \end{pmatrix} \quad (3.46)$$

ja seuraavan

$$\mathbf{P}^{(3)} = \mathbf{P}^{(2)}\mathbf{Q} = \mathbf{P}^{(1)}\mathbf{Q}^2 = \begin{pmatrix} 0.435 & 0.565 \end{pmatrix}. \quad (3.47)$$

Pidemmällä aikavälillä todennäköisyys saadaan raja-arvosta

$$\mathbf{P} = \lim_{\tau \rightarrow \infty} \mathbf{P}^{(1)}\mathbf{Q}^\tau. \quad (3.48)$$

Tämän tasapainojakauman $\mathbf{P}$ tulee toteuttaa ominaisarvoyhtälö

$$\mathbf{P}\mathbf{Q} = \mathbf{P} \quad (3.49)$$

eli

$$\sum_m P_m Q_{mn} = P_n. \quad (3.50)$$

Saavutamme siis tasapainojakauman riippumatta, mikä oli jakauma alussa.

Rajoittavampi ehto kuin (3.50) on tasapainoehto eli detaljibalanssi (*detailed balance*)

$$P_i Q_{ij} = P_j Q_{ji}. \quad (3.51)$$

Se, että tästä seuraa (3.50), nähdään helposti:

$$\sum_i P_i Q_{ij} = \sum_i P_j Q_{ji} = P_j \sum_i Q_{ji} = P_j. \quad (3.52)$$

Mitä hyötyä tästä kaikesta sitten on MC:n kannalta?

MC:ssä käymme läpi faasiavaruutta satunnaiskävelyllä ja jos meillä on oikein valitut siirtymätodennäköisyydet Q_{ij} lähestymme tasapainojakaumaa riippumatta siitä mikä oli systeemin alkutila.

Siirtymätodennäköisyyksien tulee siis toteuttaa ehdot

$$(i) \quad Q_{ij} \geq 0 \quad (3.53)$$

$$(ii) \quad \sum_j Q_{ij} = 1 \quad (3.54)$$

$$(iii) \quad P_i Q_{ij} = P_j Q_{ji} . \quad (3.55)$$

Nämä ehdot antavat melkolailla vapauksia valita siirtymätodennäköisyydet. Yksi esimerkki on vanha tuttu Metropolis

$$Q_{ij} = \begin{cases} \frac{1}{A} e^{-\Delta E(\Gamma_i, \Gamma_j)/k_B T}, \text{ jos } \Delta E > 0 \\ \frac{1}{A}, \text{ jos } \Delta E \leq 0 \end{cases} . \quad (3.56)$$

Tutkitaan todennäköisyystiheyden $P_1(y, t)$ aikariippuvuutta.

Kirjoitetaan (3.39) muotoon

$$P_1(y_2, t_1 + \tau) = \int P_1(y_1, t_1) P_{1|1}(y_1, t_1 | y_2, t_1 + \tau) dy_1 . \quad (3.57)$$

$P_1(y, t)$:n aikaderivaatta on

$$\frac{dP_1(y, t)}{dt} = \lim_{\tau \rightarrow 0} \frac{P_1(y, t + \tau) - P_1(y, t)}{\tau} , \quad (3.58)$$

joten saadaksemme yhtälön (3.57) aikaderivaatan, on meidän laskettava suure

$$\lim_{\tau \rightarrow 0} \frac{P_{1|1}(y_1, t | y_2, t + \tau) - P_{1|1}(y_1, t | y_2, t)}{\tau} . \quad (3.59)$$

Kehitetään ehdollinen todennäköisyys $P_{1|1}(y_1, t_1 | y_2, t_1 + \tau)$ sarjaksi τ :n suhteen siten, että normalisaatio

$$\int P_{1|1}(y_1, t_1; y_2, t_2) dy_2 = 1 \quad (3.60)$$

säilyy jokaisella τ :n potenssilla. Lisäksi pätee

$$P_{1|1}(y_1, t_1; y_2, t_1) = \delta(y_1 - y_2) . \quad (3.61)$$

Saamme lopulta

$$\begin{aligned}
P_{1|1}(y_1, t_1 | y_2, t_1 + \tau) \\
= \delta(y_1 - y_2) - \tau \int W_{t_1}(y_1, y) \delta(y_1 - y_2) dy + \tau W_{t_1}(y_1, y_2)
\end{aligned} \quad (3.62)$$

Suure $W_{t_1}(y_1, y_2)$ on ehdollinen todennäköisyystiheys (siirtymätodennäköisyys) aikayksikköä kohti siirtymälle $y_1 \rightarrow y_2$ aikavälillä $[t_1, t_1 + \tau]$.

Suure $\tau \int W_{t_1}(y_1, y) dy$ on siis kokonaistodennäköisyys siirtymälle tilasta y_1 mihin tahansa muuhun tilaan aikavälillä $[t_1, t_1 + \tau]$.

Eli $[1 - \tau \int W_{t_1}(y_1, y) dy]$ on todennäköisyys, että siirtymiä ei tapahdu ajassa τ ,

$[1 - \tau \int W_{t_1}(y_1, y) dy] \delta(y_1 - y_2)$ on todennäköisyystiheys, sille ettei ajassa τ tapahdu siirtymiä ja $\tau W_{t_1}(y_1, y_2)$ on todennäköisyystiheys siirtymälle $y_1 \rightarrow y_2$ aikavälillä $[t_1, t_1 + \tau]$.

Yhdistämällä (3.57), (3.58) ja (3.62) saadaan lopulta

$$\frac{\partial P_1(y_2, t)}{\partial t} = \int \{W(y_1, y_2) P_1(y_1, t) - W(y_2, y_1) P_1(y_2, t)\} dy_1 \quad (3.63)$$

Tämä on ns. *masteryhtälö*. Se kertoo todennäköisyystiheyden $P_1(y_2, t)$ muutoksen, joka aiheutuu siirtymistä tilaan y_2 kaikista muista tiloista (1. termi) ja siirtymistä tilasta y_2 kaikkiin muihin tiloihin. Tällä tietyllä tavalla itsestäänselvällä yhtälöllä on paljon sovellutuksia. MC-menetelmän voi myös hahmottaa masteryhtälön ratkaisemiseksi.

Jos systeemin tilat ovat diskreettejä masteryhtälössä integraali korvaantuu summalla

$$\frac{\partial P_1(y_2, t)}{\partial t} = \sum_i \{W(y_i, y_2) P_1(y_i, t) - W(y_2, y_i) P_1(y_2, t)\} \quad (3.64)$$

Sillä rajalla, että muutokset y :ssä tapahtuvat pieninä hyppäyksinä, saadaan masteryhtälölle eräs approksimaatio.

Tällöin $W(y', y)$ pienenee nopeasti $|y - y'|$:n kasvaessa. Voimme kirjoittaa siirtymätodennäköisyyden muotoon $W(y', y) \equiv W(y'; y - y') \equiv W(y'; \xi)$, missä $\xi = y - y'$. Masteryhtälö tulee nyt muotoon

$$\frac{\partial P_1(y, t)}{\partial t} = \int W(y - \xi; \xi) P_1(y - \xi; t) d\xi - P_1(y, t) \int W(y; -\xi) d\xi \quad (3.65)$$

Kehitetään tulo $W(y - \xi; \xi) P_1(y - \xi; t)$ sarjaksi ξ :n suhteen, jolloin saadaan

$$\begin{aligned} \frac{\partial P_1(y, t)}{\partial t} = & \int W(y; \xi) P_1(y, t) d\xi - \int \xi \frac{\partial}{\partial y} W(y; \xi) P_1(y, t) d\xi \quad . \\ & + \frac{1}{2} \int \xi^2 \frac{\partial^2}{\partial y^2} W(y; \xi) P_1(y, t) d\xi + \dots - P_1(y, t) \int W(y; -\xi) d\xi \end{aligned} \quad (3.66)$$

Oikean puolen ensimmäinen ja viimeinen termi kumoavat toisensa, joten saamme

$$\frac{\partial P_1(y, t)}{\partial t} = - \frac{\partial}{\partial y} [\alpha_1(y) P_1(y, t)] + \frac{1}{2} \frac{\partial^2}{\partial y^2} [\alpha_2(y) P_1(y, t)] , \quad (3.67)$$

missä $\alpha_n(y)$ on n :n kertaluvun momentti

$$\alpha_n(y) = \int \xi^n W(y; \xi) d\xi . \quad (3.68)$$

Yhtälö (3.67) on ns. Fokker-Planck-yhtälö.

Otetaan taas esimerkiksi yksiulotteinen satunnaiskävelijä. Yhtälö (3.39) voidaan kirjoittaa muotoon

$$P_1(n_2 l, s\tau) = \sum_{n_1} P_1(n_1 l, (s-1)\tau) P_{1|1}(n_1 l, (s-1)\tau | n_2 l, s\tau) , \quad (3.69)$$

missä $n = 0, \pm 1, \pm 2, \dots$ ilmoittaa kävelijän paikan. Suure $P_{1|1}(n_1 l, (s-1)\tau | n_2 l, s\tau)$ on ehdollinen todennäköisyys sille, että yksi askel vie n_1 :stä n_2 :een. Jos on yhtä todennäköistä mennä sekä oikealle että vasemmalle pätee

$$P_{1|1}(n_1 l, (s-1)\tau | n_2 l, s\tau) = \frac{1}{2} \delta_{n_2, n_1 + 1} + \frac{1}{2} \delta_{n_2, n_1 - 1} \quad (3.70)$$

ja yhtälö (3.69) tulee muotoon

$$P_1(nl, s\tau) = \frac{1}{2} P_1((n-1)l, (s-1)\tau) + \frac{1}{2} P_1((n+1)l, (s-1)\tau) . \quad (3.71)$$

Differentiaaliyhtälö tästä saadaan, kun kirjoitetaan se muotoon

$$\begin{aligned} & \frac{P_1(nl, s\tau) - P_1(nl, (s-1)\tau)}{\tau} \\ & = \frac{l^2}{2\tau} \left[\frac{P_1((n+1)l, (s-1)\tau) + P_1((n-1)l, (s-1)\tau) - 2P_1(nl, (s-1)\tau)}{l^2} \right] \end{aligned} \quad (3.72)$$

Merkitään $x = nl$ ja $t = s\tau$ ja mennään rajalle $l \rightarrow 0$, $\tau \rightarrow 0$ siten, että

$$D \equiv l^2/2\tau \quad (3.73)$$

pysyy äärellisenä. Näin saadaan diffuusioyhtälö

$$\frac{\partial}{\partial t}P_1(x, t) = D\frac{\partial^2}{\partial x^2}P_1(x, t) . \quad (3.74)$$

Tämä yhtälö on yksi esimerkki Fokker-Planck-yhtälöstä.

Ratkaistaan diffuusioyhtälö alkuehdolla $P_1(x, 0) = \delta(0)$. Fourier-muunnetaan P_1 :

$$\tilde{P}_1(k, t) = \int_{-\infty}^{\infty} P_1(x, t)e^{ikx}dx . \quad (3.75)$$

Yhtälö tulee nyt muotoon

$$\frac{\partial}{\partial t}\tilde{P}_1(k, t) = -Dk^2\tilde{P}_1(k, t) \quad (3.76)$$

ja ratkaisu on muotoa

$$\tilde{P}_1(k, t) = Ae^{-Dk^2t} . \quad (3.77)$$

Alkuehto tulee k -avaruudessa muotoon $\tilde{P}_1(k, 0) = 1$ eli $A = 1$. Ja käänteisellä Fourier-muunnoksella saadaan lopullinen tulos

$$P_1(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-ikx}e^{-Dk^2t}dk = \frac{1}{\sqrt{4\pi Dt}}e^{-x^2/4Dt} . \quad (3.78)$$

Tämä antaa siis todennäköisyyden, että kävelijä on paikassa x ajanhetkellä t olettaen, että se aloitti paikasta $x = 0$ ajanhetkellä $t = 0$.

Tarkastellaan vielä $x(t)$:n momenttien liikeyhtälöitä. Ensimmäinen momentin ($\langle x(t) \rangle = \int_{-\infty}^{\infty} x(t)P_1(x, t)dx$) liikeyhtälö saadaan esimerkiksi kertomalla diffuusioyhtälö (3.74) x :llä ja integroimalla x :n yli:

$$\frac{d\langle x(t) \rangle}{dt} = 0 \quad (3.79)$$

eli keskimäärin kävelijä ei kulje mihinkään.

Toiselle momentille ($\langle x^2(t) \rangle = \int_{-\infty}^{\infty} x^2(t)P_1(x, t)dx$) saadaan vastaavasti

$$\frac{d\langle x^2(t) \rangle}{dt} = 2D , \quad (3.80)$$

josta edelleen

$$\langle x^2(t) \rangle = 2Dt . \quad (3.81)$$

Tällainen käyttäytyminen on diffuusioprosessille tyypillistä.

Metropolis-MC:ssä ratkaisemme masteryhtälöä generoimalla oikealla tavalla jakautuneita tiloja systeemin faasiavaruudessa. Kuten yhtälöt (3.53), (3.54) ja (3.55) osoittivat, on meillä melko paljon liikkumavaraa valitessamme algoritmia, jonka mukaan liikumme faasiavaruudessa.

Jos kuitenkin haluamme tutkia systeemin aikakehitystä (eli aika masteryhtälössä on todellinen aika) on siirtymien eri tilojen välillä vastattava todellisia systeemin muutoksia. Puhumme ns. *kineettisestä Monte Carlosta*.

Olkoon meillä systeemin konfiguraatio C , jossa on N eri mahdollista tapahtumaa tai siirtymää. Esimerkiksi kiteen kasvatuksessa ne voivat olla adatomien hyppäys pinnalla uuteen paikkaan, uuden atomin adsorptio jne. Olkoot tapahtuman a nopeudet R_a . Sekä N että $\{R_a\}$ riippuvat systeemin konfiguraatiosta. Eri tapahtumien kokonaisnopeus on

$$Q = Q(C) = \sum_{a=1}^N R_a . \quad (3.82)$$

Tilojen väliset siirtymätodennäköisyydet voidaan muodollisesti lausua

$$W(C \rightarrow C') = \sum_{a=1}^N R_a V^a(C \rightarrow C') , \quad (3.83)$$

missä $V^a(C \rightarrow C')$ on stokastinen matriisi, joka kertoo siirtymän $C \rightarrow C'$ todennäköisyydestä. Jokaisen siirtymämekanismin a tulisi tapahtua todennäköisyydellä $R_a/Q(C)$.

Eli suoraviivainen algoritmi olisi seuraava:

1. Laske maksiminopeus $R_{\max}$ ja suhteelliset todennäköisyydet eri mekanismeille

$$P_a = R_a/R_{\max} \text{ ja tee näistä lista alkukonfiguraatiossa.}$$

2. Joka simulaatioaskeleella tee seuraavat asiat:

- a) Poimi yksi konfiguraatiossa C mahdollinen tapahtuma. Olkoon sen todennäköisyys P_e

- b) Generoi satunnaisluku väliltä $[0, 1]$.

- c) Vertaa satunnaislukua ja todennäköisyyttä P_e . Jos $r \leq P_e$, päivitä tila C uudeksi ja päivitä mahdollisten tapahtumien lista. Muuten pidä tila samana.

4 Metropoliksen Monte Carlo

4.1 Monte Carlo -integrointi

Monissa tapauksissa Monte Carlo -simulaatiossa on kyse integraalin laskemisesta. Yksi tapa MC-integroinnissa on ottaa painottamaton otos.

Esimerkiksi piin likiarvo saadaan arpomalla N_{tot} satunnaislukuparia välillä $[0, 1)$ ja laskemalla harmaalle alueelle osuneiden pisteiden lukumäärä N_{hit} :

$$\pi = \frac{4 \times \text{käyrän AC alla oleva ala}}{\text{neliön OABC ala}} = \frac{4N_{\text{hit}}}{N_{\text{tot}}} \quad (4.1)$$

Tuloksen tarkkuus luonnollisesti riippuu N_{tot} :n arvosta ja se käyttäytyy kuten $O(N_{\text{tot}}^{-1/2})$.

Seuraavassa kuvassa esimerkkinä MC-integroinnin tulos ja sen virhe N_{hit} :n funktiona.

Yksiulotteisessa tapauksessa MC-integrointi häviää selvästi muille (deterministisille) menetelmille; esimerkiksi Simpsonin menetelmälle (10^4 funktion arvon laskulla $\pi \approx 3.141593$).

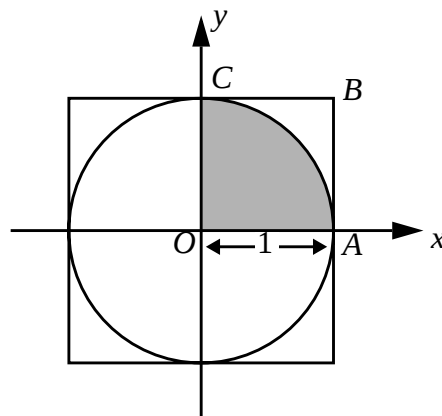
Mutta otetaan esimerkki statistisesta fysiikasta. On laskettava konfiguraatiointegraali $Z_{NVT} = \int e^{-\beta U} d\mathbf{r}$, missä $\beta = 1/k_B T$, ja U on systeemin potentiaalienergia. Olkoon meillä 100 atomia kuutiassa $x, y, z \in [-L/2, L/2]$. Karkea Simpsonin menetelmä (10 funktion evaluointia/koordinaatti) vaatisi 10^{300} funktion evaluointia.

Konfiguraatiointegraali voidaan tietysti laskea MC-integroinnilla käyttäen painottamatonta otosta:

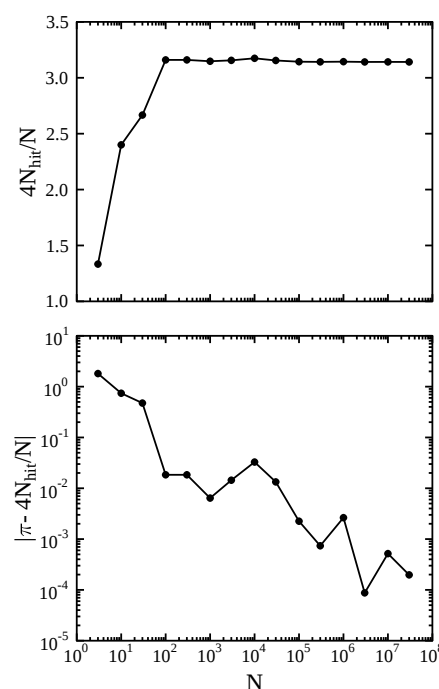
- i. Arvo satunnaisesti kaikkien 100 atomin paikat (konfiguraatio i).
- ii. Laske potentiaalienergia $U(i)$, ja Boltzmanin tekijä.

Integraali on nyt

$$Z_{NVT} = \frac{V^N}{n} \sum_{i=1}^n e^{-\beta U(i)} . \quad (4.2)$$

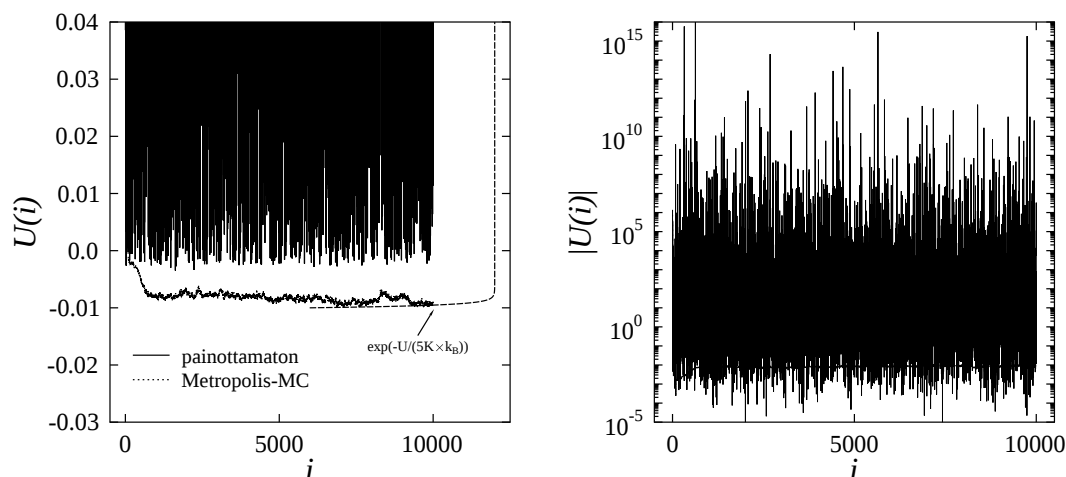


Kuva 4.1: Piin likiarvon laskemisen MC-integroinnilla.



Kuva 4.2: Piin likiarvon MC-laskun tulokset. Ylemmässä kuvassa on esitetty itse likiarvo ja alemmassa sen absoluuttinen virhe.

Tämä on tehoton menetelmä, koska suurin osa otoksista on sellaisia, joille Boltzmannin tekijä $e^{-\beta U}$ on hyvin pieni. Oheisen kuvan esittämässä tapauksessa on laskettu 13 LJ-potentiaalilla¹ vuorovaikuttavan atomin potentiaalienergia. Painottamattoman otoksen tapauksessa atomit on sijoitettu satunnaisesti $10 \times 10 \times 10 \text{ \AA}$ suuruiseen kuutioon. Metropolis-MC:n tapauksessa taas alkutila arvottiin satunnaisesti, mutta seuraavat tilat generoitiin Metropolis-algoritmilla, jossa $T = 5 \text{ K}$.



Kuva 4.3: LJ-systeemin potentiaalienergia painottamattomalla ja Metropolis-otannalla.

Nähdään selkeästi, että satunnaisotanta on täysin hyödytön yhtälön (4.2) tapaisten suureiden laskemisessa.

Samat vaikeudet koskevat ensemblekeskiarvojen

$$\langle A \rangle = \frac{\int d\mathbf{r} A e^{-\beta U}}{\int d\mathbf{r} e^{-\beta U}} \approx \frac{\sum_{i=1}^n A(i) e^{-\beta U(i)}}{\sum_{i=1}^n e^{-\beta U(i)}} \quad (4.3)$$

laskemista. Parempi tulos saadaan painotetulla otoksella.

Olkoon meillä laskettavana integraali

$$F = \int_{x_1}^{x_2} dx f(x) . \quad (4.4)$$

Kirjoitetaan se muotoon

1. Lennard-Jones: $U(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]$.

$$F = \int_{x_1}^{x_2} dx \left[\frac{f(x)}{\rho(x)} \right] \rho(x) , \quad (4.5)$$

missä $\rho(x)$ on mielivaltainen todennäköisyystiheys eli $[\rho(x) \geq 0 \text{ ja } \int_{x_1}^{x_2} dx \rho(x) = 1]$.

Otetaan N :n suuruinen otos siten, että valitaan satunnaislukuja ζ , joka noudattavat jakaumaa $\rho(x)$ välillä $[x_1, x_2]$.

Tällöin

$$F = \left\langle \frac{f(\zeta)}{\rho(\zeta)} \right\rangle , \quad (4.6)$$

missä keskiarvo otetaan otoksen yli.

Yksinkertaisin valinta olisi

$$\rho(x) = \frac{1}{x_2 - x_1} , \quad x_1 \leq x \leq x_2 , \quad (4.7)$$

jolloin arvio integraalille on

$$F \approx \frac{x_2 - x_1}{N} \sum_{i=1}^N f(\zeta_i) . \quad (4.8)$$

Edellämainittuun esimerkkiin (π :n arvon laskeminen) sovellettuna $f(x) = (1 - x^2)^{-1/2}$ ja $0 \leq x \leq 1$.

Mikä olisi optimaalinen tiheysfunktio $\rho(x)$? Integraalin (4.6) virheeksi voidaan arvioida seuraava

$$(\Delta F)^2 = \frac{\langle f^2/\rho^2 \rangle - \langle f/\rho \rangle^2}{N} . \quad (4.9)$$

Tämä voidaan lausua integraalina

$$(\Delta F)^2 \propto \int_{x_1}^{x_2} \frac{f^2}{\rho^2} \rho dx - \left[\int_{x_1}^{x_2} \frac{f}{\rho} \rho dx \right]^2 = \int_{x_1}^{x_2} \frac{f^2}{\rho^2} \rho dx - \left[\int_{x_1}^{x_2} f dx \right]^2 . \quad (4.10)$$

Nyt etsimme variaatiolaskennan avulla funktion ρ , joka minimoi suureen (4.10) lisäehdolla, että funktio on normitettu ykköseen:

$$\frac{\delta}{\delta \rho} \left\{ \int_{x_1}^{x_2} \frac{f^2}{\rho^2} \rho dx - \left[\int_{x_1}^{x_2} f dx \right]^2 + \lambda \int_{x_1}^{x_2} \rho dx \right\} = 0 . \quad (4.11)$$

Jättämällä keskimäinen termi pois (se ei riipu ρ :sta) ja siirtämällä variaatio integraalin sisään saadaan

$$-\frac{f^2}{\rho^2} + \lambda = 0 \Rightarrow \rho = \frac{|f|}{\int_{x_1}^{x_2} |f| dx} , \quad (4.12)$$

missä on otettu huomioon ρ :n normitus. Tulos on muuten hyvä, mutta tiheysfunktion normitustekijässä on funktion itseisarvon integraali!

Tästä ongelmasta voimme päästä eroon useammallakin tavalla. Välttämättä ei tarvitse pyrkiä täsmälleen (4.12):n mukaiseen optimiin, vaan riittää, että tiheysfunktio on lähellä optimia. Tällöin sen normituksen laskeminen voi olla mahdollista.

Tietyn jakauman mukaisen otoksen voi myös ottaa tuntematta jakauman normitusta. Tästä puhuttiin kappaleessa 3.3.

4.2 Metropolis-algoritmi

Statistisen fysiikan ensemblekeskiarvo (4.3) voidaan kirjoittaa muotoon

$$\langle A \rangle = \int d\Gamma \rho_{\text{ens}}(\Gamma) A(\Gamma) , \quad (4.13)$$

missä ρ_{ens} on kyseessäolevan ensemblen tiheysfunktio. Painotetulla otoksella (painofunktio ρ) integraalin MC-arvio tulee muotoon

$$\langle A \rangle = \left\langle \frac{A \rho_{\text{ens}}}{\rho} \right\rangle . \quad (4.14)$$

Lähes aina MC-simuloinneissa valitaan otoksen painofunktioksi tuo ρ_{ens} . Se ei välttämättä ole täysin optimaalinen tietyn suureen kannalta, mutta useinhan simuloinneissa halutaan tietoa useasta systeemin ominaisuudesta. Tällöin ensemblekeskiarvo tulee yksinkertaiseen muotoon

$$\langle A \rangle = \langle A \rangle_{\text{run}} = \frac{1}{N} \sum_{i=1}^N A(i) . \quad (4.15)$$

Otoksen generointi tehdään siis Markovin prosessin avulla. Prosessin määrittelevät siirtymätodennäköisyydet¹ π_{mn} tilojen välillä ja todennäköisyysjakauma ρ_n . Eli käytännössä otetaan systeemille jokin alkutila ja arvotaan seuraava tila siirtymätodennäköisyyksien π_{mn} mukaan. Käymällä läpi tiloja tarpeeksi kauan, lähestymme prosessin tasapainojakaumaa.

Jotta lähestyisimme tasapainojakaumaa, tulee siirtymätodennäköisyyksien toteuttaa ehdot

$$(i) \quad \pi_{mn} \geq 0 \quad (4.16)$$

$$(ii) \quad \sum_n \pi_{mn} = 1 \quad (4.17)$$

$$(iii) \quad \rho_m \pi_{mn} = \rho_n \pi_{nm} . \quad (4.18)$$

(Tässä ρ on siis tasapainojakauma.)

Lisäksi prosessin tulisi olla *ergodinen* eli mistä tahansa tilasta tulee olla mahdollista siirtyä mihin tahansa toiseen tilaan. Voidaan osoittaa, että tämä ehto toteutuu, jos

- i. Prosessi on aperiodinen (eli kaikki tilat ovat aperiodisia).
- ii. Prosessin tilat ovat toistuvia (*persistent*) ja toistumisaika (*recurrence time*) on äärellinen.

Periodiselle tilalle i pätee: $\pi_{ii}^{(n)} > 0$ vain, jos kokonaisluku $n = t$ on t :n monikerta ($t > 1$), ja t on suurin tällainen kokonaisluku. Tässä $\pi_{ii}^{(n)}$ on todennäköisyys, että siirtymä $i \rightarrow i$ tapahtuu n :llä askeleella. Jos tila ei ole periodinen, se on aperiodinen.

Tilan i toistumisaika määritellään seuraavasti. Olkoon $f_{ij}^{(n)}$ todennäköisyys, että prosessi, joka alkaa tilasta i käy ensimmäisen kerran tilassa j täsmälleen n :llä askeleella. Edelleen,

$$f_{ij}^{(0)} = 0 , \quad (4.19)$$

$$f_{ij} = \sum_{n=1}^{\infty} f_{ij}^{(n)} , \quad (4.20)$$

$$\mu_i = \sum_{n=1}^{\infty} n f_{ii}^{(n)} . \quad (4.21)$$

Tässä f_{ij} on todennäköisyys, että tilasta i aloitettaessa käydään tilassa j .

Kanonisen ensemblen tapauksessa siis tasapainojakauma on

$$\rho_{NVT} = \rho_m = \frac{1}{Z_{NVT}} e^{-U(m)/k_B T} , \quad (4.22)$$

missä normitustekijä on

1. Merkintätapa on nyt hieman toinen kuin kappaleessa 3.3: $Q_{mn} \rightarrow \pi_{mn}$, $P_m \rightarrow \rho_m$.

$$Z_{NVT} = \int d\Gamma e^{-U(\Gamma)/k_B T}. \quad (4.23)$$

Ehdot (4.16)-(4.18) ovat varsin väljät, joten voimme melko vapaasti valita siirtymätodennäköisyydet π_{mn} . Koska yhtälössä (4.18) esiintyy vain todennäköisyystiheyksien suhde, yksi ehdotus siirtymätodennäköisyyksiksi on

$$\pi_{mn} = \begin{cases} \alpha_{mn}; & \rho_n \geq \rho_m \quad m \neq n \\ \alpha_{mn} \left(\frac{\rho_n}{\rho_m} \right); & \rho_n < \rho_m, \quad m \neq n \\ 1 - \sum_{m \neq n} \pi_{mn}, & m = n \end{cases}. \quad (4.24)$$

Tässä α on symmetrinen stokastinen matriisi $\alpha_{mn} = \alpha_{nm}$. Se sisältää tavallaan yritesiirtymätodennäköisyydet. Symmetrisyyteen perustuen voidaan osoittaa, että yhtälön (4.24) siirtymätodennäköisyydet toteuttavat ehdot (4.16)-(4.18).

Yhtälö (4.24) ei tietenkään ole ainoa vaihtoehto. Barkerin menetelmässä¹ matriisi π saa muodon

$$\pi_{mn} = \begin{cases} \alpha_{mn} \left(\frac{\rho_n}{\rho_n + \rho_m} \right), & m \neq n \\ 1 - \sum_{m \neq n} \pi_{mn}, & m = n \end{cases}. \quad (4.25)$$

Jos MC-ajo koostuu n :stä simulaatioaskeleesta (otoksesta) saadaan arvio jonkin suureen kanonisen ensemblen keskiarvolle $\langle A \rangle_{NVT}$:

$$\langle A \rangle_{NVT} = \langle A \rangle_{\text{run}} + O(n^{-1/2}). \quad (4.26)$$

Koska on monia eri mahdollisuuksia konstruoida siirtymätodennäköisyydet π_{mn} , parasta olisi valita sellainen, joka minimoi keskiarvon $\langle A \rangle_{\text{run}}$ varianssin. Eräs mitta on ns. statistinen tehottomuus s

$$s = \lim_{n \rightarrow \infty} \left[\frac{n \sigma^2(\langle A \rangle_{\text{run}})}{\sigma^2(A)} \right], \quad (4.27)$$

joka mittaa sitä, miten hitaasti ajo suppenee kohti raja-arvoa. Merkitään

$$\pi_1 \leq \pi_2, \quad (4.28)$$

jos jokainen π_1 :n ei-diagonaalinen elementti on pienempi kuin π_2 :n. Tällöin voidaan osoit-

1. P. A. Flinn, G. M. McManus, *Phys. Rev.* **124** (1961) 54. ja A. A. Barker, *Aust. J. Phys.* **18** (1965) 119.

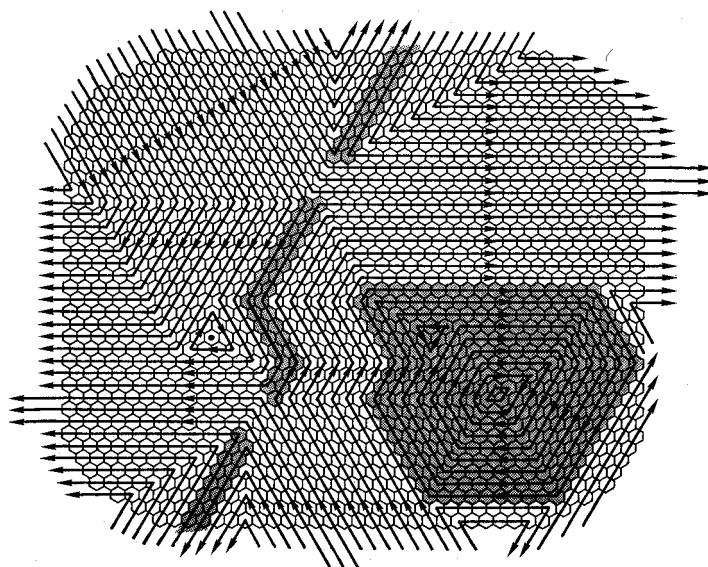
taa¹, että

$$s(\langle A \rangle, \pi_1) \geq s(\langle A \rangle, \pi_2) . \quad (4.29)$$

Jos ei-diagonaalelementit ovat suuria, on todennäköisyys pysyä samassa tilassa pieni, ja otos faasiavaruuden on siten parempi. Yhtälöiden (4.24) ja (4.25) perusteella ($\rho_n, \rho_m > 0$) Metropolis-algoritmillä saavutetaan pienempi statistinen tehottomuus.

Toinen tärkeä asia on se, että algoritmi käy järkevässä ajassa (simulaatioaskelissa) läpi edustavan otoksen faasiavaruudesta. Eli kyse on ergodisuudesta.

Joissain tilanteissa faasiavaruudessa on erillisiä alueita, joiden väliset siirtymät ovat hyvin epätodennäköisiä: faasiavaruudessa on pullonkauloja.



Kuva 4.4: Ergodisuus.

Esimerkkejä tällaisista tapauksista ovat mm. kahden faasin koeksistenssi, faasitransitiot yleensä ja nesteet hyvin suurissa tiheyksissä.

Eräs tärkeä siirtymätodennäköisyyksien π_{mn} piirre on, että siinä esiintyy vain tilojen todennäköisyyksien suhde ρ_n/ρ_m . Tästä syystä emme tarvitse normitettuja tiheyksiä

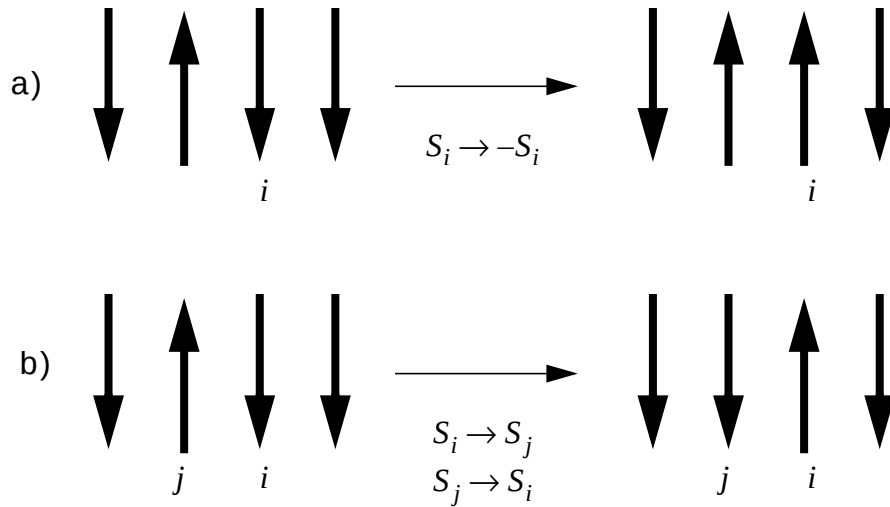
$$\rho_m = \frac{e^{-U(m)/k_B T}}{\int d\Gamma e^{-U(\Gamma)/k_B T}} . \quad (4.30)$$

Tuon normitustekijän laskeminenhan on yhtä vaikeaa kuin ensemblekeskiarvojen. Toisaalta emme suoraan pysty laskemaan esimerkiksi vapaata energiaa, joka on oleellisesti normitustekijän logaritmi, tai entropiaa.

Miten käytännössä toteutetaan nuo yritesiirrot? Se tietysti riippuu täysin tutkittavasta mallista. Aina kuitenkin pyritään uusi tila valitsemaan siten, että se ei poikkea paljon edellisestä. Tällöin

1. P. H. Peskun, *Biometrika* **60** (1973) 607.

on suurempi todennäköisyys hyväksyä se. Esimerkiksi Isingin mallissa siirto voitaisiin toteuttaa oheisen kuvan mukaisilla tavoilla.



Kuva 4.5: Yritesiirrot Isingin mallissa. a) Yhden spinin kääntäminen. b) Naapurispinien vaihtaminen. Tässä tapauksessa magnetoituma säilyy vakiona.

Tarkastellaan yksityiskohtaisemmin systeemiä, jossa on N kappaletta atomia tai molekyyliä, jotka vuorovaikuttavat (pari)potentiaalin $U(r)$ välityksellä. Systeemin tilan m määräävät atomien paikat $\{\mathbf{r}_i^m\}$ ($i = 1 \dots N$).

Liikemäärät unohdamme, koska pystymme niistä ensemblekeskiarvoihin aiheutuvan osan laskemaan eksaktisti (tietysti olettaen jotain suureesta A). Eli systeemin potentiaalienergia

$$U(m) = \sum_{i=1}^{N-1} \sum_{j=i+1}^N U(r_{ij}^m) \quad ; \quad r_{ij}^m = |\mathbf{r}_i^m - \mathbf{r}_j^m| \quad (4.31)$$

määrää siirtymätodennäköisyydet π_{mn} .

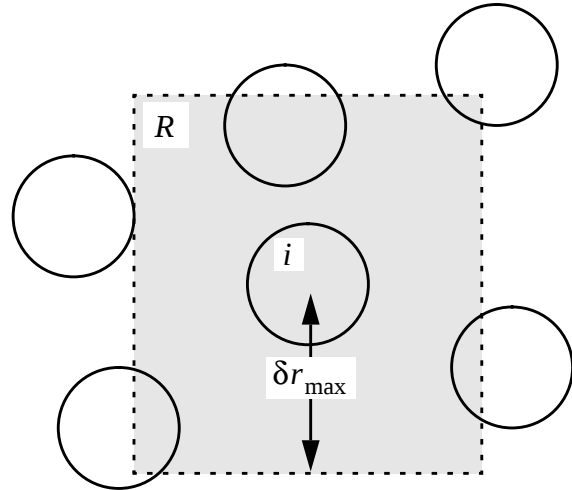
Stokastisen matriisin α avulla pääsemme tilasta m mihin tahansa sen naapuritilaan n yhtäsuurilla todennäköisyyksillä. Ainoa rajoitus oli symmetrisyys $\alpha_{mn} = \alpha_{nm}$.

Naapuritila voidaan määritellä kuvan mukaan.

Kuusi atomia ovat tilassa m . Uusi tila n muodostetaan valitsemalla satunnaisesti jokin atomeista (i) ja siirtämällä sitä paikasta $\mathbf{r}_i^m$ satunnaisesti yhtäsuurella todennäköisyydellä mihin tahansa paikkaan $\mathbf{r}_i^n$ neliön R sisällä.

Tietokoneessa on äärellinen määrä N_R paikkoja neliön alueella, joten muodollisesti matriisi α on

$$\alpha_{mn} = \begin{cases} 1/N_R, & \mathbf{r}_i^n \in R \\ 0, & \mathbf{r}_i^n \notin R \end{cases} . \quad (4.32)$$



Kuva 4.6: Naapuritilat atomisysteemin Monte Carlo -simulaatiossa.

Maksimisiirtymä $\delta r_{\max}$ kontrolloi Markovin ketjun suppenemista. C-koodina asia näyttää seuraavanlaiselta

```
rxinew = rx[i] + (2.0*ran(&seed)-1.0)*drmax
ryinew = ry[i] + (2.0*ran(&seed)-1.0)*drmax
rzinew = rz[i] + (2.0*ran(&seed)-1.0)*drmax
```

Siirtymätodennäköisyys π_{mn} riippuu tilojen m ja n todennäköisyystiheyksien ρ_n ja ρ_m suhteesta

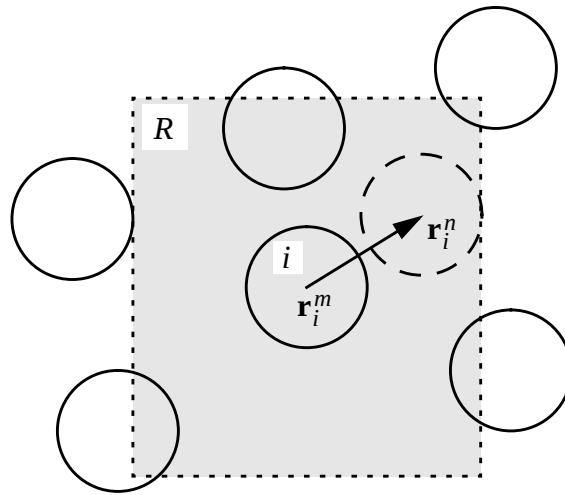
$$\frac{\rho_n}{\rho_m} = e^{-\delta U_{nm}/k_B T} , \quad (4.33)$$

missä eri tilojen potentiaalienergioiden erotus on

$$\delta U_{nm} = U(n) - U(m) . \quad (4.34)$$

Tämän laskemiseen ei tarvitse käydä kaikkia atomipareja kuten (4.31):ssä vaan riittää laskea atomin i vuorovaikutus muiden atomien kanssa:

$$\delta U_{nm} = \sum_{j=1}^N U(r_{ij}^n) - \sum_{j=1}^N U(r_{ij}^m) \quad (4.35)$$



Kuva 4.7: Yritesiirto atomisysteemin MC-simulaatiossa.

Jos ehdotettu siirtymä on alamäkeen energiassa ($\delta U_{nm} \leq 0$), uusi tila hyväksytään. Tämä vastaa ensimmäistä tapausta kaavassa (4.24), eli tila hyväksytään todennäköisyydellä $\pi_{mn} = \alpha_{mn}$.

Jos taas $\delta U_{nm} > 0$, uusi tila hyväksytään todennäköisyydellä ρ_n/ρ_m kaavan (4.24) toisen rivin mukaan. Tekijä α_{nm} on tässäkin tapauksessa mukana. Tiheyksien suhde on tietysti

$$\frac{\rho_n}{\rho_m} = \frac{Z_{NVT}^{-1} e^{-U(n)/k_B T}}{Z_{NVT}^{-1} e^{-U(m)/k_B T}} = \frac{e^{-U(n)/k_B T}}{e^{-U(m)/k_B T} e^{\delta U_{nm}/k_B T}} = e^{-\delta U_{nm}/k_B T}. \quad (4.36)$$

Hyväksyminen todennäköisyydellä $\exp(-\delta U_{nm}/k_B T)$ toteutetaan seuraavasti:

- i. Generoidaan tasaisesti välille $[0, 1)$ jakautunut satunnaisluku ξ
- ii. Jos $\xi < \exp(-\delta U_{nm}/k_B T)$, uusi tila hyväksytään.

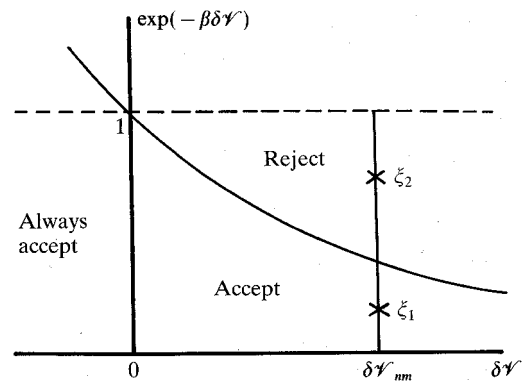
Seuraava kuva selvittää asiaa.

Jos tämä ylämäkeen tehtävä siirto hylätään, systeemi pysyy vanhassa tilassaan, ja tämä tila lasketaan uudestaan mukaan.

Siis: jokainen ehdotettu tila hyväksytään todennäköisyydellä

$$\min(1, e^{-\delta U_{nm}/k_B T}). \quad (4.37)$$

Alla on esitetty tyypillisen Metropolis-MC-koodin ydin.



Kuva 4.8: Yritesiirron hyväksyminen MC-simulaatiossa

```

deltv = vnew-vold
deltvb = beta*deltv

if (deltvb < 75.0) {
    if (deltvb <= 0.0) {
        v = v+deltv;
        rx[i] = rxinew;
        ry[i] = ryinew;
        rz[i] = rzinew;
        naccpt++;
    } else if (exp(-deltvb) > ran(seed)) {
        v = v+deltv;
        rx[i] = rxinew;
        ry[i] = ryinew;
        rz[i] = rzinew;
        naccpt++;
    }
}
ntrial++;
/*... laske keskiarvoja ... */

```

Tässä $\text{deltv} = \delta U_{nm}$ ja $\text{beta} = \beta = 1/k_B T$.

Ehdolla $\text{deltvb} < 75.0$ varmistetaan se, ettei eksponenttifunktion laskussa tule ongelmia. Siis konfiguraatiot, joissa atomit menevät liikaa päällekkäin hylätään suoraan.

Miten sitten valitaan parametri $\delta r_{\max}$?

Jos se on liian pieni, suurin osa yritetiloista hyväksytään, mutta faasiavaruutta käydään läpi hyvin hitaasti: peräkkäiset tilat ovat hyvin korreloituneita.

Jos taas se on liian suuri, huomattava osa yritetiloista hylätään seurauksena taas liian hidas eteneminen faasiavaruudessa. Parametrin arvo voidaan simulaation kuluessa helposti asettaa haluttuun arvoon:

```

if (ntrial%nadjst==0) {
    ratio = (float)naccpt/(float)nadjst;
    if (ratio > 0.5) {
        drmax*=1.05;
    } else {
        drmax*=0.95;
    }
    naccpt=0;
}

```

Tässä asetetaan hyväksymisosuus arvoon 0.5 . Tämä on yleinen arvo, mutta ei ole täysin selvää, että se on optimaalinen. Kvantitatiivinen peruste $\delta r_{\max}$:n valinnalle voisi olla atomien paikkojen poikkeamien neliöiden summa simulaatioaskeleen n funktiona

$$(\Delta \mathbf{r})^2 = \frac{1}{N} \sum_{i=1}^N [\mathbf{r}_i^{(0)} - \mathbf{r}_i^{(n)}]^2. \quad (4.38)$$

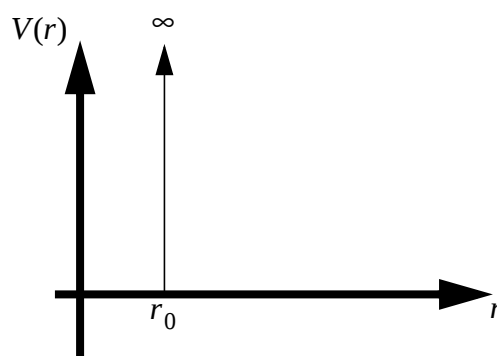
Mitä suurempi tämä poikkeama, sitä parempi otos saadaan faasiavaruudesta tietyllä simulaatioaskeleiden määrällä.

Alunperin Metropolis-MC:ssä yhdellä simulaatioaskeleella siirrettiin vain yhtä atomia. Toinen vaihtoehto on siirtää systeemin kaikkia atomeja kerralla. Yhden atomin siirrossa systeemin atomeja voidaan käydä läpi tietyssä järjestyksessä tai satunnaisesti. Yleensä, kun puhutaan simulaatioaskeleesta, tarkoitetaan N yhden atomin siirtoa.

Jos systeemin partikkeleiden potentiaali on ns. kovan pallon (hard sphere), potentiaali

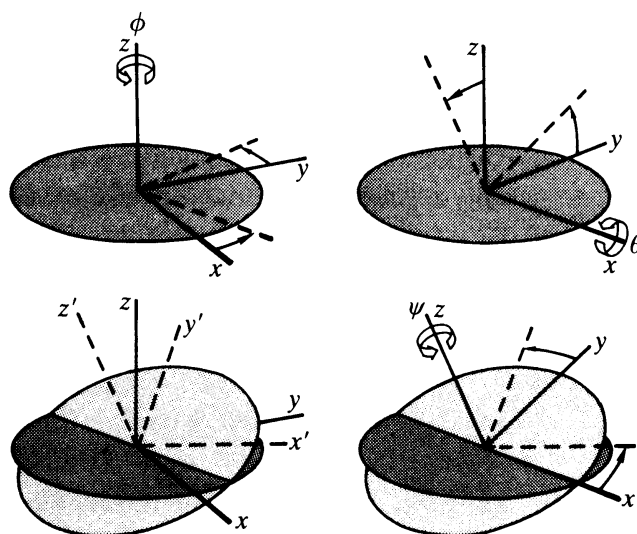
$$U(r_{ij}) = \begin{cases} \infty, & r_{ij} \leq r_0 \\ 0, & r_{ij} > r_0 \end{cases} \quad (4.39)$$

redusoituu Metropolis-MC-simulaatio yksinkertaiseen muotoon. Jos atomin siirrossa kahden atomin välimatka $r_{ij} \leq r_0$, on tekijä $\exp(-\delta U_{nm}/k_B T)$ nolla, eikä uutta tilaa hyväksytä. Eli siirtoyrityksessä tarkistetaan, ettei siirrettävä atomi mene päällekkäin muiden kanssa. Huomaa myös, että lämpötila ei vaikuta simulaation kulkuun millään tavalla.



Kuva 4.9: Kovan pallon potentiaali.

Jos simuloitava systeemi koostuu molekyyleistä, joiden atomien välillä on kiinteät sidokset, on myös niiden orientaatio otettava huomioon. Molekyylin translaatio toteutetaan edelläesitettyyn tapaan siirtämällä sen massakeskipistettä. Orientaatio esitetään usein Eulerin kulmien ϕ , θ , ψ avulla.



Kuva 4.10: Eulerin kulmat.

Molekyylin satunnainen kierto toteutetaan muuttamalla Eulerin kulmia satunnaisesti:

$$\begin{aligned}\phi_i^n &= \phi_i^m + (2\xi_1 - 1)\delta\phi_{\max} \\ \theta_i^n &= \theta_i^m + (2\xi_2 - 1)\delta\theta_{\max} , \\ \psi_i^n &= \psi_i^m + (2\xi_3 - 1)\delta\psi_{\max}\end{aligned}\quad (4.40)$$

missä ξ_j ovat välille $[0, 1]$ tasaisesti jakautuneita satunnaislukuja ja $\delta\phi_{\max}$, $\delta\theta_{\max}$ ja $\delta\psi_{\max}$ ovat Eulerin kulmien maksimimuutokset.

MC-algoritmissa tilojen m ja n todennäköisyyksien suhde on nyt

$$\frac{\rho_n}{\rho_m} = \frac{\exp[-\beta(U_m + \delta U_{nm})] d\mathbf{r}^n d\Omega^n}{\exp[-\beta U_m] d\mathbf{r}^m d\Omega^m} , \quad (4.41)$$

missä avaruuskulmaelementit ovat

$$d\Omega^n = \prod_{i=1}^N d\Omega_i^n , \quad d\Omega_i^n = (\sin\theta_i^n d\theta_i^n d\psi_i^n d\phi_i^n) / \Omega . \quad (4.42)$$

Tässä $\Omega = 8\pi^2$ ei-lineaarisille ja $\Omega = 4\pi$ lineaarisille molekyyille. Yhden molekyylin pyöräytyksessä suhde voidaan kirjoittaa muodossa

$$\frac{\rho_n}{\rho_m} = \exp[-\beta\delta U_{nm}] \frac{\sin\theta_i^n}{\sin\theta_i^m} . \quad (4.43)$$

Tuon sinien suhteen, joka ottaa huomioon avaruuskulmaelementin koon, on esiinnyttävä joko siirron hyväksymisosassa tai jo yritesiirtotodennäköisyydessä α_{nm} . Jälkimmäisessä tapauksessa kierto kulman θ verran valitaan siten, että $\cos\theta$ on tasaisesti jakautunut:

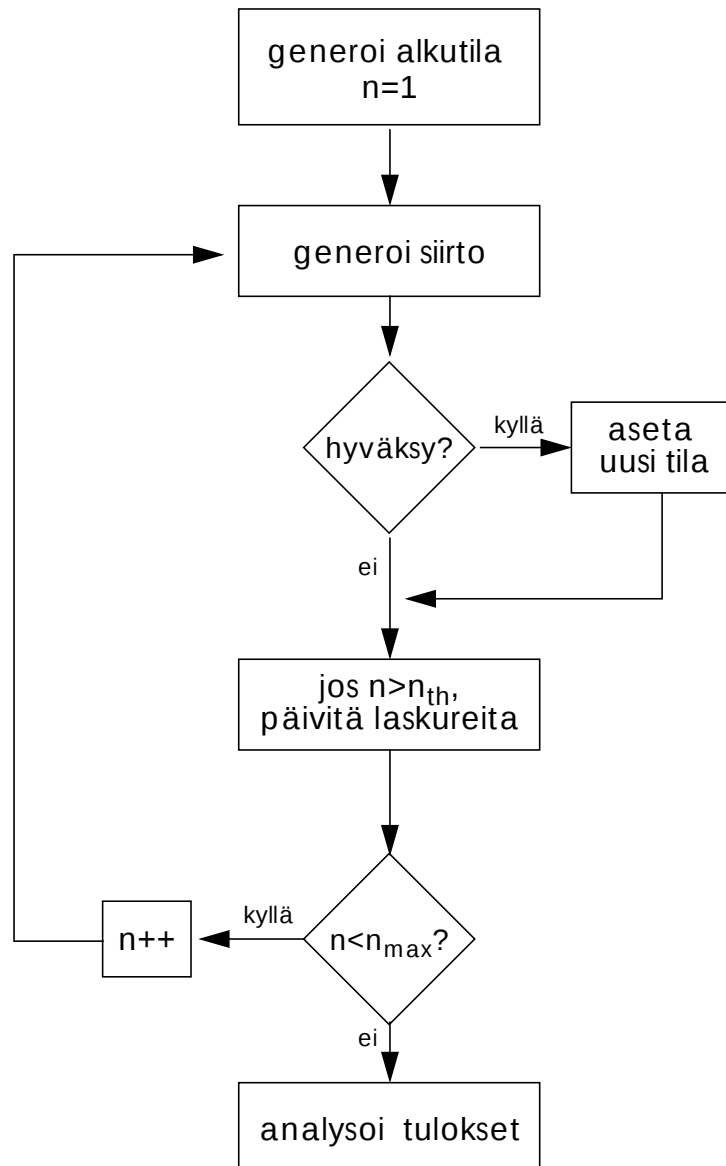
$$\cos\theta_i^n = \cos\theta_i^m + (2\xi_2 - 1)\delta(\cos\theta)_{\max} , \quad (4.44)$$

ja algoritmin hyväksymistodennäköisyys on normaali $\min(1, \exp(-\beta\delta U_{nm}))$.

Toinen tapa tehdä pyöräytys on valita satunnaisesti jokin kolmesta koordinaattiakselista ja pyöräyttää molekyyliä sen ympäri.

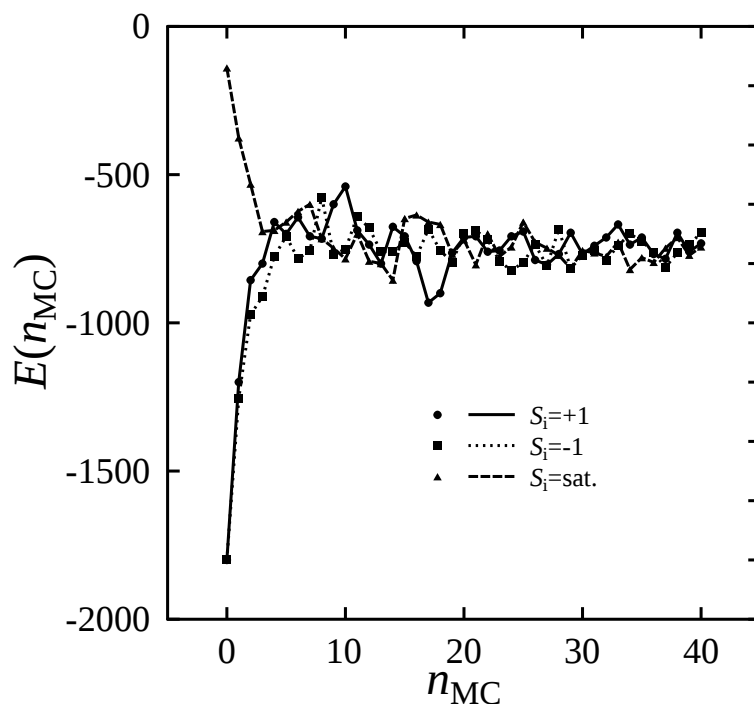
Jos molekyyliä ei voi mallintaa täysin jäykkänä kappaleena eli joitain sisäisiä vapausasteita on otettava huomioon, voi tilanne tulla hyvinkin monimutkaiseksi. Asiasta kiinnostuneet voivat tutustua *Allen-Tildesleyn* kappaleeseen 4.7.2.

Lopuksi voisimme hahmotella tyypillisen simulointiajon eri vaiheet oheisessa kuvassa.



Kuva 4.11: Tyypillisen Monte Carlo -simulaation kulku.

Alkutilan generointi riippuu tietysti käytettävästä systeemistä. Sen valinta ei ole kuitenkaan kovin kriittinen, sillä MC-algoritmi vie systeemin tasapainotilaan riippumatta alkutilasta (ainakin periaatteessa). Kuvassa 4.12 on esitetty, miten Ising-systeemin energia kehittyy simuloinnin kuluessa, kun alkutilana on joko kaikilla hilapisteillä sama tai satunnaisesti arvottu spinin arvo. Energia saavuttaa nopeasti tasapainoarvonsa.



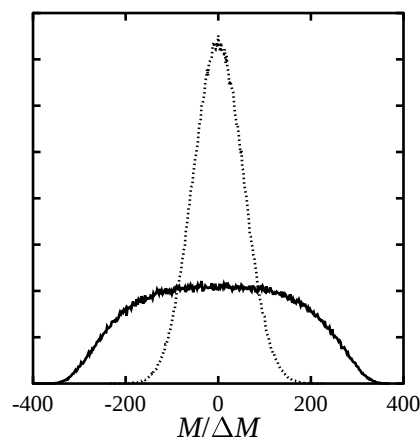
Kuva 4.12: Ising mallin MC-simuloinnin alkutilan valinta. $T = 4.0$, $J = 1.0$, 30×30 -hila.

4.3 Metropolis-algoritmin parannuksia

Joskus joudumme tutkimaan systeemejä, joissa normaali MC-algoritmi ei tunnu käyvän läpi faasiavaruutta tarpeeksi tehokkaasti, jotta pystyisimme laskemaan järkeviä keskiarvoja.

Esimerkiksi Isingin mallin Curie-pisteen läheisyydessä magnetoituman flukтуаatiot kasvavat hyvin suuriksi (ks. oheinen kuva). Normaalilla MC-algoritmillä pystytään yhdellä askeleella (spiniä kohti) muuttamaan magnetoitumaa enintään kahdella. Jos magnetoituman jakauma levenee, joudutaan laskemaan enemmän MC-siirtoja, jotta saadaan riittävän hyvä otos keskiarvojen laskemiseen.

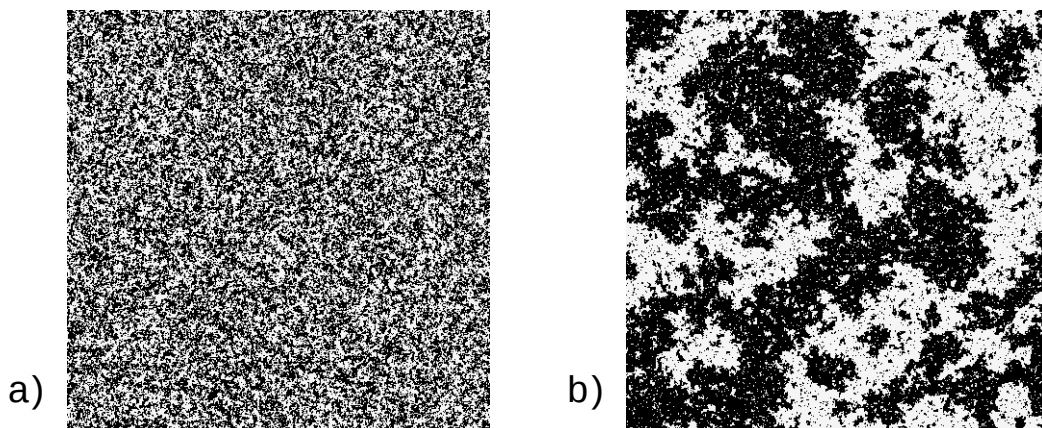
Kun Ising-systeemin lämpötila on korkea, spinit ovat varsin sekaisin, ja yhden spinin kääntöryityksellä on suurehko todennäköisyys onnistua. Lämpötilan laskiessa lähelle Curie-



Kuva 4.13: Magnetoituman M jakauma 20×20 :n suuruudessa Isingin mallissa. 10% (leveä jakauma) ja 45% (kapea jakauma) Curie-pisteen yläpuolella.

$\Delta M = 2$ on magnetoituman maksimimuutos yhdessä MC-siirrossa.

pistettä, systeemiin syntyy alueita, joilla spinit ovat samansuuntaisia. Tällaisella alueella olevan spinin käännön hyväksyminen ei olekaan kovin todennäköistä.



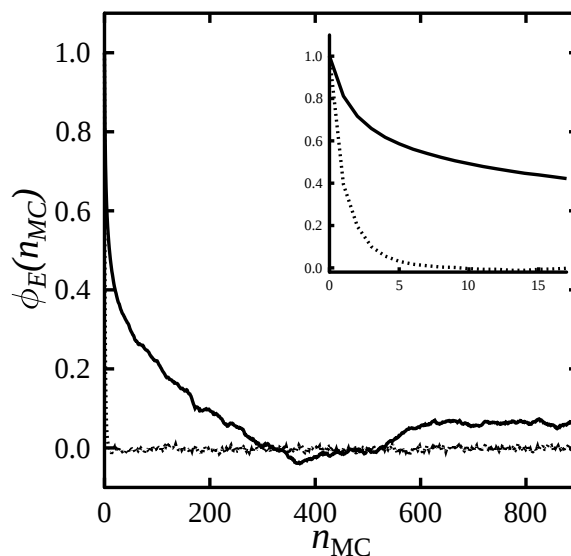
Kuva 4.14: Kaksiulotteisen Isingin mallin MC-simuloinnilla generoituja konfiguraatioita. a) $T = 1.6T_c$, b) $T = T_c$. Laskettu ohjelmalla `xpotts`, joka kuuluu `xtoys`-pakettiin (<http://penguin.phy.bnl.gov/www/xtoys/xtoys.html>).

Tällöin peräkkäiset tilat ovat vahvasti korreloituneita, koska uusia tiloja hyväksytään vähän. Kvantitatiivisesti tämä saadaan näkyviin laskemalla esimerkiksi systeemin energian normitettu autokorrelaatiofunktio

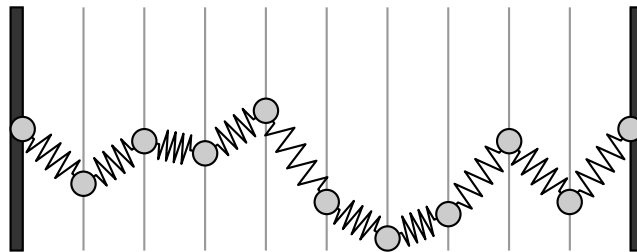
$$\phi_E(i) = \frac{\langle E(0)E(i) \rangle - \langle E \rangle^2}{\langle E^2 \rangle - \langle E \rangle^2}, \quad (4.45)$$

missä indeksi i viittaa simulointiaskeleeseen. Jos peräkkäiset tilat olisivat täysin satunnaisia, olisi autokorrelaatiofunktio deltapiikki: $\phi_E(i) = \delta_{i0}$. Kuvaan 4.15 on piirretty Isingin mallin ϕ_E sekä korkeassa että lähelle Curie-pistettä sijoittuvassa lämpötilassa.

Otetaan yksinkertainen esimerkki tapauksesta, jossa normaali Metropolis-algoritmi toimii huonosti¹. Olkoon meillä N kappaletta, jotka on kytketty toisiinsa jousilla. Kappaleet voivat liikkua z -suunnassa.



Kuva 4.15: Isingin mallin energian autokorrelaatiofunktio lämpötiloissa $T = 2.27 \approx T_c$ (yhtenäinen viiva) ja $T = 4.0$ (katkoviiva).



Kuva 4.16: Kytetyt harmoniset värähtelijät.

Systeemin potentiaalienergia on

$$U = \frac{1}{2} \sum_{i=1}^N (z_i - z_{i-1})^2, \quad (4.46)$$

missä z_0 ja z_N on kiinnitetty nollaan.

Tämän systeemin MC-simulointi Metropolis-algoritmilla on yksinkertainen ohjelmointitehtävä. Ohjelman tärkeimmät rivit voisivat näyttää seuraavalta:

```
for (imc=1;imc<=nmc;imc++) {
  for (i=2,i<nchain;i++) {
    znew=z[i]+eps*2.0*(ranf(&seed)-0.5);
    Eold=0.5*(pow(z[i]-z[i-1],2)+pow(z[i+1]-z[i],2));
    Enew=0.5*(pow(znew-z[i-1],2)+pow(z[i+1]-znew,2));
    dE=Enew-Eold;
    if (dE<0.0) {
      z[i]=znew;
      E+=dE;
    } else {
      if (ranf(seed)<exp(-beta*dE)) {
        z[i]=znew; E+=dE;
      }
    }
  }
}
```

Mutta ohjelma on varsin hidas faasiavaruuden läpikäymiseen kuten oheinen kuva havainnollistaa. Vaikka simulaatio on hyvinkin pitkä ei faasiavaruudesta eli tässä tapauksessa keskimmäisen värähtelijän paikasta saada kovin hyvää otosta. Tämä johtuu värähtelijän paikan jakauman leveydestä.

-
1. Esimerkki lainattu W. Krauthin julkaisusta *Introduction to Monte Carlo Algorithms*, Advances in Computer Simulation, toim. János Kertész, Imre Kondor, Lecture Notes in Physics **501** (Springer Verlag, Berlin, 1998), saatavissa myös preprint-palvelimesta <http://babbage.sissa.it/>, julkaisu cond-mat/9612186

os haluamme uusien tilojen hyväksymistodennäköisyydeksi puolikkaan, on maksimi-siirron pituus (eps edellisen sivun listauksessa) pieni verrattuna värähtelijän $z_{N/2}$ paikkajakaumaan. Tämä jakauma taas on leveä, koska koko ketjun poikkeama on luokkaa $\sqrt{N}$. Ongelma johtuu tässä tapauksessa siitä, että värähtelijät on vahvasti kytketty toisiinsa. Useamman värähtelijän poikkeutus samalla kertaa ei auta tilannetta, sillä uuden tilan energia on hyvin todennäköisesti niin suuri, että se hylätään.

Ratkaisuna on pyrkiä valitsemaan yritesiirrot siten, että sen lisäksi, että ne usein hyväksytään, ne käyvät läpi faasiavaruutta tarpeeksi tehokkaasti.

Kuvan mukaan, jos yritämme kasvattaa yritesiirron pituutta, tulee normaalin Metropolis-algoritmin yritesiirron jakauma α_{mn} on hyvin erilaiseksi verrattuna lopulliseen hyväksymistodennäköisyyteen.

Yhtälön (4.24) mukaisesti siirtymätodennäköisyys tilasta m tilaan n voidaan kirjoittaa muodossa

$$\pi_{mn} = \alpha_{mn} p_{mn} , \quad (4.47)$$

missä p_{mn} on yritesiirron hyväksymistodennäköisyys. Tässä tapauksessa (ja edellisen luvun atomisysteemin tapauksessa) yritesiirtodennäköisyyden jakauma oli tasainen:

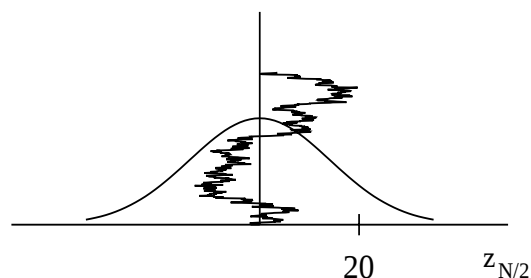
$$\alpha_{mn} = \begin{cases} 1, & \delta z \leq \epsilon \\ 0, & \delta z > \epsilon \end{cases} . \quad (4.48)$$

Tästä johtuu hylkäyksien suuri määrä.

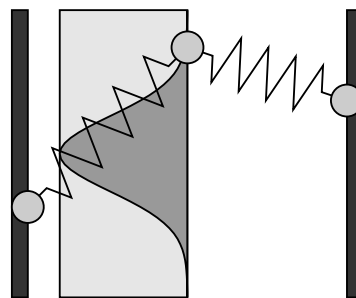
Kirjoitetaan detaljibalanssi (4.18) siten, että siirtymätodennäköisyys on muotoa (4.47):

$$\rho_m \alpha_{mn} p_{mn} = \rho_n \alpha_{nm} p_{nm} . \quad (4.49)$$

On helppo nähdä, että oikea jakauma saavutetaan, kun hyväksymistodennäköisyydeksi otetaan



Kuva 4.17: Kytettyjen värähtelijöiden keskimmäisen kappaleen paikan jakauma (tasainen käyrä), kun $N = 40$. Lisäksi kuvaan on piirretty värähtelijän polku 400 000 simuloitiaskeleen aikana.



Kuva 4.18: Värähtelijät z_{i-1} , z_i ja z_{i+1} . Äärimmäiset värähtelijät on nyt kiinnitetty. Tumma alue kuvaa z_i :n jakaumaa, vaalea alue normaalin Metropolis-algoritmin yritesiirron jakau-

$$p_{mn} = \min(1, \frac{\rho_n \alpha_{nm}}{\alpha_{mn} \rho_m}) . \quad (4.50)$$

Meillä on siis vapaat kädet valita sopiva yritesiirtotodennäköisyys α_{mn} . Sen painotus otetaan huomioon yhtälön (4.50) mukaisella hyväksymistodennäköisyydellä.

Lisähankaluuksia voi kuitenkin tuoda se, että algoritmissa on pystyttävä laskemaan yritetodennäköisyydet sekä siirtymälle $m \rightarrow n$ että siirtymälle $n \rightarrow m$.

Entä jos asetamme yritetodennäköisyyksiksi $\alpha_{mn} = \rho_n$ ja $\alpha_{nm} = \rho_m$? Tällöin hyväksymistodennäköisyys on ykkönen, eli yhtään siirtoa ei hylätä. Toisaalta, Metropolis-algoritmi otettiin käyttöön juuri sen vuoksi, että otoksen laskeminen jakaumasta ρ_m on mahdotonta.

Joskus tämä on kuitenkin mahdollista. Otetaan taas esimerkiksi nuo värähtelijät. Jos z_{i-1} ja z_{i+1} on kiinnitetty, on z_i :n jakauma

$$\rho_{i|i-1, i+1} \propto \exp[-\beta(z_i - \bar{z}_i)^2] , \quad (4.51)$$

missä,

$$\bar{z}_i = \frac{z_{i-1} + z_{i+1}}{2} .$$

Todennäköisyyttä (4.51) voimme käyttää yritetodennäköisyytenä. Toisaalta siinä tarvitaan gaussisesti jakautuneita satunnaislukuja. Näiden generoinnissa käytetään usein hylkäystekniikkaa.

Voimme kuitenkin laajentaa yhtälö (4.51) integroimalla useamman värähtelijän yli:

$$\rho_{i|i-2, i+2} \propto \exp[-2\beta(z_i - \bar{z}_i)^2] , \quad (4.52)$$

missä

$$\bar{z}_i = \frac{z_{i-2} + z_{i+2}}{2} .$$

Kaavalla (4.52) voimme arpoa z_i :n uuden paikan ja kaavalla (4.51) z_{i-1} :n ja z_{i+1} :n paikat. Algoritmia voidaan jatkaa siten, että koko ketjun värähtelijöiden paikkoja päivitetään kerralla, eikä hylkäyksiä tarvita (paitsi generoitaessa gaussisia satunnaislukuja).

Käytännössä asiat eivät ole näin hyvin. Mutta usein algoritmi (4.50) toimii hyvin, vaikka α ja ρ eroavat toisistaan. Käytetään taas esimerkkinä värähtelijöitä. Lisätään energiaan (4.46) pieni häiriö

$$U_1 = \gamma \sum_{i=1}^N f(z_i) . \quad (4.53)$$

Jokaisen konfiguraatio energia on nyt

$$\tilde{U}(z_1, \dots, z_N) = U_0 + U_1 , \quad (4.54)$$

missä U_0 on kaavan (4.46) antama. Käyttämällä yritetodennäköisyyttä $\alpha_{mn} = \exp[-\beta U_0^n]$ (joka on ideaalinen häiriintymättömälle systeemille) saadaan hyväksymistodennäköisyydeksi

$$p_{mn} = \min(1, \frac{\exp[-\beta U_1^n]}{\exp[-\beta U_1^m]}) . \quad (4.55)$$

Tämähän on tuttu Metropolis, mutta mukana on vain energian häiriötermi.

Yritetodennäköisyyden muokkaamista simuloinnin nopeuttamiseksi on sovellettu luonnollisesti hyvin monenlaisissa systeemeissä. Esimerkiksi atomisysteemien simuloinneissa yritesiirrot voidaan painottaa atomiin kohdistuvan voiman mukaan. Puhutaan voimia suosivasta MC:stä.

Spinsysteemien simuloinnissa ns. klusterialgoritmi nopeuttaa simulointia huomattavasti. Tutkitaan Isingin mallia. Systeemin energia on muotoa

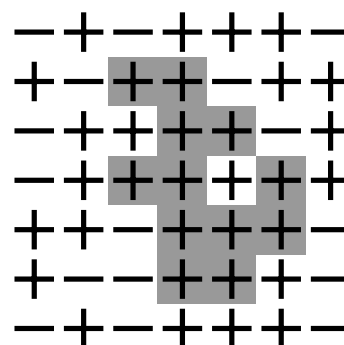
$$E = -J \sum_{\langle i, j \rangle} S_i S_j ; S_i = \pm 1 . \quad (4.56)$$

Systeemillä (kun $J > 0$) on olomuodon muutos pisteessä $\beta = \ln(1 + \sqrt{2})/2$. Tällöin esimerkiksi magnetoituman fluktuaatiot kasvavat suuriksi (ks. kuva 4.13). Jos algoritmi kääntää vain yhden spinin kerrallaan, on otoksen kerääminen hidasta. Kuten kytkettyjen värähtelijöiden tapauksessa, ei yksinkertainen usean spinin kääntäminen toimi.

Klusterialgoritmissa muodostetaan samansuuntaisista spineistä klusteri lähtemällä jostain spinistä ja lisäämällä siihen todennäköisyydellä P samansuuntaisia naapureita.

Oletetaan, että lähdemme rakentamaan klusteria oheisen kuvan harmaalta osalta. Yritetodennäköisyys α_{mn} koostuu termistä, joka kertoo klusterin sisäisten ‘++’-linkkien osuuden, α_{mn}^{int} , ja klusterin reunaa kuvaavasta termistä. Tämä reuna koostuu kahden tyyppisistä linkeistä:

sisä	ulko	lukumäärä
+	-	c_1
+	+	c_2



Kuva 4.19: Klusterin muodostaminen.

Reunan osuus systeemin energiassa on

$$U_{\partial C} = -c_2 + c_1 \quad (J = 1) . \quad (4.57)$$

Kuvan 4.19 esimerkissä $c_1 = 10$ ja $c_2 = 14$. Yritetodennäköisyys on nyt

$$\alpha_{mn} = \alpha_{mn}^{\text{int}} (1 - P)^{c_2} . \quad (4.58)$$

Todennäköisyyssiehyksiä ρ_m laskettaessa emme tarvitse termejä, jotka eivät muutu klusterin käännön seurauksena. Eli vain reunan energia otetaan huomioon:

$$\rho_m = \exp[-\beta(c_1 - c_2)] . \quad (4.59)$$

Päinvastaisessa siirtymässä ($n \rightarrow m$, ks kuva 4.20) reuna koostuu seuraavanlaisista linkeistä:

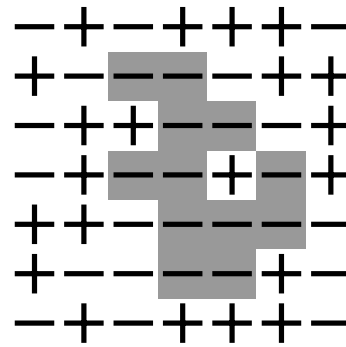
sisä	ulko	lukumäärä
-	-	c_1
-	+	c_2

Reunan osuus energiasta on

$$U_{\partial C} = -c_1 + c_2 . \quad (4.60)$$

Yritetodennäköisyys on nyt

$$\alpha_{nm} = \alpha_{nm}^{\text{int}} (1 - P)^{c_1} . \quad (4.61)$$



Kuva 4.20: Toteutettu klusterisiirto.

Termi α_{nm}^{int} on sama kuin päinvastaisessa siirtymässä. Kuten edellä todennäköisyyssiheys on

$$\rho_n = \exp[-\beta(c_2 - c_1)] . \quad (4.62)$$

Detaljibalanssi tulee muotoon

$$e^{-\beta(c_1 - c_2)} (1 - P)^{c_2} p_{mn} = e^{-\beta(c_2 - c_1)} (1 - P)^{c_1} p_{nm} . \quad (4.63)$$

Tästä saamme lopulta hyväksymistodennäköisyyden

$$p_{mn} = \min\left(1, \frac{e^{-\beta(c_2 - c_1)} (1 - P)^{c_1}}{e^{-\beta(c_1 - c_2)} (1 - P)^{c_2}}\right) . \quad (4.64)$$

Algoritmi on seuraavanlainen:

- i. Ota systeemistä yksi spin satunnaisesti.
- ii. Lisää siihen viereisiä samansuuntaisia spinejä todennäköisyydellä P .
- iii. Lopeta klusterin kasvattaminen, kun yhtäkään sen reuna ylittävää linkkiä ei ole valittu.
- iv. Käännä klusterin spinin todennäköisyydellä (4.64).

Huomaa, että jos asetamme $1 - P = \exp(-2\beta)$, saamme algoritmin, jossa ei ole ollenkaan hylkäyksiä.

5 Monte Carlo -simulaatiot eri ensembleissä

5.1 Yleistä

Eri ensemblit vastaavat erilaisia ulkoisia olosuhteita. Edellä kävimme läpi kanonisen ensemblin (eli N , V ja T vakioita) MC-algoritmin. Usein koeolosuhteissa on tilavuuden sijasta paine vakio. Tällöin on kyseessä NPT-ensemble. Suurkanonista ensembleä tarvitaan, kun hiukkasten lukumäärä voi vaihdella. Esimerkiksi adsorptiossa kaasufaasia ei kannata simuloida, vaan käytetään suurkanonista ensembleä ja etukäteen määrättyä lämpötilaa ja kemiallista potentiaalia.

Eri ensembleiden antamat keskiarvot periaattessa lähestyvät samaa arvoa termodynaamisella rajalla eli, kun systeemin koko kasvaa rajatta. Fluktuaatioille ei kuitenkaan käy näin. Tästä seuraa, että useat vastefunktiot ovat eri ensembleille erilaiset.

Seuraavassa esitellään eri ensembleiden MC-simulointi lähinnä atomisysteemien kannalta.

5.2 Mikrokanoninen ensemble

Mikrokanonisessa ensemblessä (NVE) systeemin kokonaisenergia säilyy vakiona. Suoraviivaisin tapa simuloida tätä ensembleä olisi NVE-molekyylidynamiikka, jossa liikeyhtälöt pitävät huolen energian säilymisestä¹.

Mikrokanonisen ensemblin MC-simuloinnissa meidän on saatava aikaiseksi satunnaiskävely faasiavaruuden vakioenergiapinnoilla.

NVE-ensemblin todennäköisyysstiheys voidaan kirjoittaa muodossa

$$\rho_{NVE} = Z_{NVE}^{-1} \delta(H(\Gamma) - E) , \quad (5.1)$$

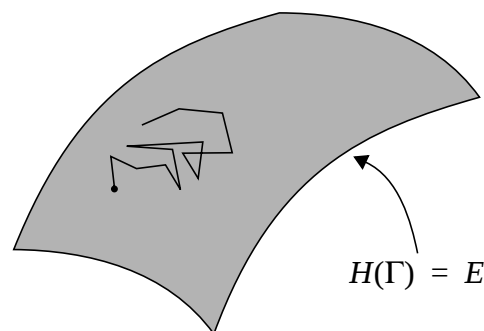
missä partitiofunktio on

$$Z_{NVE} = \int_{\Omega} d\Gamma \delta(H(\Gamma) - E) . \quad (5.2)$$

Ja Hamiltonin funktio

$$H(\Gamma) = K(\mathbf{p}) + U(\mathbf{r}) . \quad (5.3)$$

MC-simuloinnissa ei ole mukana kineettistä energiaa, joten otetaan sitä kuvaamaan ylimääräinen vapausaste, *demoni*², jolla on energia E_D . Todennäköisyysstiheys on nyt



Kuva 5.1: Vakioenergiapinta faasiavaruudessa.

1. Jos tutkimme systeemiä, jolla on liikeyhtälöt.
2. M. Creutz, *Phys. Rev. Lett.* **50** (1983) 1411.

$$\rho_{NVE} \propto \delta(U(\mathbf{r}) + E_D - E) . \quad (5.4)$$

Demonin energia on rajoitettu positiivisiin arvoihin.

Simulaatioalgoritmi on seuraavanlainen:

- i. Konstruoidaan alkutila, jonka (potentiaali)energia E .
- ii. Asetetaan demonille energia (esim. $E_D = 0$).
- iii. Muutetaan systeemin tilaa (atomin paikan muutos) $m \rightarrow n$.
- iv. Lasketaan energiaero uuden ja vanhan tilan välillä $\delta U_{nm} = U(n) - U(m)$.
- v. Jos energia pienenee eli $\delta U_{nm} < 0$, hyväksy uusi tila ja kasvata demonin energiaa:
 $E_D \rightarrow E_D + |\delta U_{nm}|$.
- vi. Jos taas energia kasvaa (tai pysyy samana) eli $\delta U_{nm} \geq 0$, hyväksy uusi tila ainoastaan, jos demonilla on tarpeeksi energiaa eli $E_D - \delta U_{nm} > 0$. Tällöin vähennä demonin energiaa: $E_D \rightarrow E_D - \delta U_{nm}$.
- vii. Palaa askeleeseen iii.

Voidaan osoittaa, että demonin energiajakauma lähestyy Boltzmannin jakaumaa

$$\rho_{E_D} \propto e^{-E_D/k_B T} . \quad (5.5)$$

Tästä voimme periaatteessa laskea systeemin lämpötilan. (Vrt. lämpötilan laskeminen molekyyliidynamiikkasimulaatioissa.)

Tätä NVE-MC-menetelmää ei ole juurikaan käytetty atomi- tai molekyyliysteemien simulointiin.

Etuina voisi ehkä mainita satunnaislukujen vähäisen käytön. Jos atomit käydään läpi järjestyksessä, ei satunnaislukuja tarvita ollenkaan. Hilasimulaatioissa (esim. Ising-malli) menetelmä voi nopeuttaa laskentaa huomattavastikin¹.

5.3 Vakiopaine-ensemble

Vakio-NPT-ensemble on kokeelliselta kannalta tärkeä. NPT -ensemblissä paineen odotusarvo on vakio ja tilavuus fluktuoi. Tarkastellaan tilannetta atomi- tai molekyyliysteemissä. Todennäköisyystiheyden konfiguraatio-osa on

$$\rho_{NPT} \propto e^{-\beta(PV + U(\mathbf{r}^N))} \quad (5.6)$$

ja ensemblekeskiarvo (tai sen konfiguraatio-osa) on nyt muotoa

1. M. Creutz *Phys. Rev. Lett.* **69** (1992) 1002.]

$$\langle A \rangle_{NPT} = \frac{\int_0^\infty dV e^{-\beta PV} \int_V d\mathbf{r}^N A(\mathbf{r}^N, V) e^{-\beta U(\mathbf{r}^N)}}{\int_0^\infty dV e^{-\beta PV} \int_V d\mathbf{r}^N e^{-\beta U(\mathbf{r}^N)}} . \quad (5.7)$$

Oletetaan, että simulaatiokoppi on kuutio, jonka sivut ovat $L = V^{1/3}$ ja tehdään muuttujanvaihdos:

$$\mathbf{s}^N = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N) = L^{-1} \mathbf{r}^N \quad (5.8)$$

$$\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N) . \quad (5.9)$$

Tällöin odotusarvo (5.7) tulee muotoon

$$\begin{aligned} \langle A \rangle_{NPT} &= \frac{\int_0^\infty dV e^{-\beta PV} \int_\omega L^{3N} d\mathbf{s}^N A(\mathbf{s}^N, V) e^{-\beta U(\mathbf{s}^N)}}{\int_0^\infty dV e^{-\beta PV} \int_\omega L^{3N} d\mathbf{s}^N e^{-\beta U(\mathbf{s}^N)}} \quad (5.10) \\ &= \frac{\int_0^\infty dV e^{-\beta PV} V^N \int_\omega d\mathbf{s}^N A(\mathbf{s}^N, V) e^{-\beta U(\mathbf{s}^N)}}{\int_0^\infty dV e^{-\beta PV} V^N \int_\omega d\mathbf{s}^N e^{-\beta U(\mathbf{s}^N)}} \\ &= \frac{\int_0^\infty dV e^{-\beta PV} e^{\beta(N \ln V / \beta)} \int_\omega d\mathbf{s}^N A(\mathbf{s}^N, V) e^{-\beta U(\mathbf{s}^N)}}{\int_0^\infty dV e^{-\beta PV} e^{\beta(N \ln V / \beta)} \int_\omega d\mathbf{s}^N e^{-\beta U(\mathbf{s}^N)}} \end{aligned}$$

Tässä ω tarkoittaa integrointia yksikkökuution yli.

Järjestelemällä termejä saadaan lopulta muoto

$$\langle A \rangle_{NPT} = \frac{\int_0^\infty \int_\omega dV d\mathbf{s}^N A(\mathbf{s}^N, V) e^{-\beta \left[PV - \frac{N}{\beta} \ln V + U(\mathbf{s}^N) \right]}}{\int_0^\infty \int_\omega dV d\mathbf{s}^N e^{-\beta \left[PV - \frac{N}{\beta} \ln V + U(\mathbf{s}^N) \right]}} . \quad (5.11)$$

Eli meillä on nyt systeemi, jossa on $3N + 1$ vapausastetta $\{V, \mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N\}$, ja jonka faasia-varuuden todennäköisyystiheysfunktio on

$$\rho \propto e^{-\beta \left[PV - \frac{N}{\beta} \ln V + U(\mathbf{s}^N) \right]}. \quad (5.12)$$

MC-algoritmissa uusi yrittäjä generoidaan siirtämällä atomin paikka ja/tai muuttamalla tilavuutta

$$\mathbf{s}_i^n = \mathbf{s}_i^m + \delta \mathbf{s}_{\max} (2\xi - 1) \quad (5.13)$$

$$V_n = V_m + \delta V_{\max} (2\xi - 1), \quad (5.14)$$

missä ξ on vektori, jonka komponentit ovat välillä $[0, 1)$ tasaisesti jakautuneita satunnaislukuja.

Parametrit $\delta \mathbf{s}_{\max}$ ja $\delta V_{\max}$ määräävät maksimimuutokset paikoissa ja tilavuudessa. Kuten kanonisessakin ensemblessä nämä parametrit valitaan siten, että saadaan uusien tilojen hyväksymissuhteeksi n. 0.5.

Toisaalta meillä on nyt kaksi parametria, joten haluttu hyväksymissuhde voidaan saada aikaiseksi erilaisilla kombinaatioilla. Simulaatio-ohjelma voi automaattisesti säätää näitä parametreja halutun suhteen aikaansaamiseksi.

Kun uusi tila on generoitu, lasketaan energianmuutos δH

$$\delta H_{nm} = \delta U_{nm} + P(V_n - V_m) - Nk_B T \ln \left(\frac{V_n}{V_m} \right). \quad (5.15)$$

Tilavuuden muutoksesta seuraavan energianmuutoksen laskeminen on yleisesti ottaen laskennallisesti kallista. Molekyylin paikan muutoksesta aiheutuvan energianmuutoksen δU_{nm} laskemiseen riittää enintään $2(N-1)$ potentiaalin laskua. Tilavuuden muutos taas vaatii (paripotentiaaleilla) $(1/2)N(N-1)$ vuorovaikutuksen laskemisen.

Yksinkertaisimmille potentiaaleille voidaan kuitenkin energianmuutos laskea skaalaamalla. Otetaan esimerkiksi Lennard-Jones¹, jolle systeemin potentiaalienergia tilassa m on

1. $U(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]$

$$U_m = 4\epsilon \sum_{i=1}^N \sum_{j>i} \left(\frac{\sigma}{L_m s_{ij}^m} \right)^{12} - 4\epsilon \sum_{i=1}^N \sum_{j>i} \left(\frac{\sigma}{L_m s_{ij}^m} \right)^6 . \quad (5.16)$$

$$\equiv U_m^{(12)} + U_m^{(6)}$$

Jos tilasta toiseen ($m \rightarrow n$) muuttuu vain tilavuus eli $L_m \rightarrow L_n$, saadaan uuden tilan n energia skaalaamalla:

$$U_n = U_m^{(12)} \left(\frac{L_m}{L_n} \right)^{12} + U_m^{(6)} \left(\frac{L_m}{L_n} \right)^6 \quad (5.17)$$

ja energiaero

$$\delta U_{nm} = \delta U_{nm}^{\text{vol}} = U_m^{(12)} \left[\left(\frac{L_m}{L_n} \right)^{12} - 1 \right] + U_m^{(6)} \left[\left(\frac{L_m}{L_n} \right)^6 - 1 \right] . \quad (5.18)$$

Jos yhtäaikaa siirretään atomia ja muutetaan tilavuutta on potentiaalienergian muutos

$$\delta U_{nm} = \delta U_{nm}^{\text{dis}} + \delta U_{nm}^{\text{vol}} , \quad (5.19)$$

missä

$$\delta U_{nm}^{\text{dis}} = U_n(L_n) - U_m(L_n) . \quad (5.20)$$

Eli tässä tapauksessa ensin muutetaan tilavuutta ja sen jälkeen hiukkasen paikkaa.

Jos potentiaali on monimutkaisempi, on tilavuudenmuutosten energiaeron laskeminen kallista ja niitä on tehtävä harvemmin.

NPT-ohjelmassa voidaan luonnollisesti laskea paine viriaalista. Tämän paineen keskiarvon tulisi luonnollisesti olla sama kuin asetetun ulkoisen paineen, joka esiintyy yhtälössä (5.15).

Lennard-Jones systeemin NPT-MC-ohjelman tilavuudenmuutososa voisi näyttää seuraavalta:

```

/** attempt a box move */
boxnew=box+(2.0*ranf(dummy)-1.0)*dboxmx;
ratbox=box/boxnew; rbox= 1.0/ratbox;
rcutn=rcut*rbox;

/** calculate scaling parameters */
rat3=rarbox*ratbox*ratbox;
rat6=rat3*rat3; rat12=rat6*rat6;

/** scale energy, and virial including lrc */
v12new=v12*rat12; v6new=v6*rat6;
w12new=w12*rat12; w6new=w6*rat6;

/** calculate change in energy and volume */
deltv=v12new+v6new-v12-v6;
dpv=presur*(boxnew*boxnew*boxnew-vol);
dvol=3.0*temp*(float)n*log(ratbox);
delthb=beta*(deltv+dpv+dvol);

/** check for acceptance */
if (delthb<75.0) {
    if (delthb<=0.0) {
        v12=v12new; v6=v6new;
        w12=w12new; w6=w6new;
        for (i=1;i<=n;i++) {
            rx[i]=rx[i]*rbox;
            ry[i]=ry[i]*rbox;
            rz[i]=rz[i]*rbox;
        }
        box=boxnew; rcut=rcutn; acboxa=acboxa+1.0;
    } else if (exp(-delthb)>ranf(dummy)) {
        v12=v12new; v6=v6new;
        w12=w12new; w6=w6new;
        for ( i=1;i<=n,i++) {
            rx[i]=rx[i]*rbox;
            ry[i]=ry[i]*rbox;
            rz[i]=rz[i]*rbox;
        }
        box=boxnew; rcutn=rcut; acboxa=acboxa+1.0;
    }
}
}
boxinv=1.0/box; vol=box*box*box;
dens=(float)n/vol;

/** calculate energy and pressure */
vn=(v12+v6)/(float)n;
pres=dens*temp+(w12+w6)/vol;

/** increment accumulators */
acm=acm+1.0; acv=acv+vn;
acp=acp+pres; acd=acd+dens;
acvsq=acvsq+vn*vn;
acpsq=acpsq+pres*pres;
acdsq=acdsq+dens*dens;

```

5.4 Suurkanoninen ensemble

Suurkanonisessa enesemblessä on systeemin kemiallinen potentiaali on vakio ja hiukkasten lukumäärä fluktuoi. Vakioita ovat siis μ , V ja T .

Todennäköisyystiheyden konfiguraatio-osa on¹

$$\rho_{\mu VT} \propto \frac{1}{N! \Lambda^{3N}} e^{-\beta[U(\mathbf{r}^N) - \mu N]}, \quad (5.21)$$

missä Λ on terminen de Broglien aallonpituus

$$\Lambda = \left(\frac{h^2}{2\pi m k_B T} \right)^{1/2}. \quad (5.22)$$

Kuten muistamme, suureen A ensemblekeskiarvon konfiguraatio-osa on

$$\begin{aligned} \langle A \rangle_{\mu VT} &= \frac{\sum_{N=0}^{\infty} \int d\mathbf{r} A(\mathbf{r}) (N! \Lambda^{3N})^{-1} e^{-\beta[U(\mathbf{r}^N) - \mu N]}}{Q_{\mu VT}} \\ &= \frac{\sum_{N=0}^{\infty} (N! \Lambda^{3N})^{-1} e^{\beta\mu N} \int d\mathbf{r} A(\mathbf{r}) e^{-\beta U(\mathbf{r}^N)}}{Q_{\mu VT}}. \end{aligned} \quad (5.23)$$

Konfiguraatio-osaan voidaan taas lisätä liikemääristä aiheutuva osuus. On kuitenkin huomattava, että tämä osuus riippuu atomien lukumäärästä, joten $\langle N \rangle$ on laskettava simulaatiosta.

Myös suurkanonisessa enesemblessä on käytännöllistä tehdä yhtälön (5.8) mukainen muuttujanvaihdos. Tästä saamme tulokseksi tekijän V^N :

1. Nyt emme voi unohtaa N :stä riippuvaa vakiokerrointa.

$$\begin{aligned}
\langle A \rangle_{\mu VT} &= \frac{\sum_{N=0}^{\infty} (N! \Lambda^{3N})^{-1} V^N e^{\beta \mu N} \int_{\omega} d\mathbf{s} A(\mathbf{s}) e^{-\beta U(\mathbf{s})}}{Q_{\mu VT}} \quad . \quad (5.24) \\
&= \frac{\sum_{N=0}^{\infty} e^{-\ln N!} e^{-3N \ln \Lambda} e^{N \ln V} e^{\beta \mu N} \int_{\omega} d\mathbf{s} A(\mathbf{s}) e^{-\beta U(\mathbf{s})}}{Q_{\mu VT}} \\
&= \frac{\sum_{N=0}^{\infty} \int_{\omega} d\mathbf{s} A(\mathbf{s}) \exp \{ -\beta [U(\mathbf{s}) - N\mu] - \ln N! - 3N \ln \Lambda + N \ln V \}}{Q_{\mu VT}}
\end{aligned}$$

Eli suurkanonisessa Monte Carlo -simulaatiossa tulisi saada aikaan Markov-ketju, jonka tasapainojakauma on

$$\rho \propto \exp \{ -\beta [U(\mathbf{s}) - N\mu] - \ln N! - 3N \ln \Lambda + N \ln V \} . \quad (5.25)$$

Koska myös hiukkasten lukumäärä voi fluktuoida, on algoritmin avulla pystyttävä myös poistamaan tai lisäämään atomeja.

Käytännössä Markov-ketju toteutetaan seuraavalla tavalla.

Käytetään kolmenlaisia siirtymiä tilasta toiseen:

- i. Atomin siirto.
- ii. Atomin tuhoaminen.
- iii. Atomin luominen satunnaiseen paikkaan systeemissä.

Atomin siirto toteutetaan kanonisesta ensemblestä tuttuun tapaan.

Jos atomi tuhotaan, on tilojen todennäköisyyksien suhde (tilassa m alunperin N atomia)

$$\begin{aligned}
\frac{\rho_n}{\rho_m} &= \frac{e^{-\beta [U(\mathbf{s}^n) - (N-1)\mu] - \ln(N-1)! - 3(N-1) \ln \Lambda + (N-1) \ln V}}{e^{-\beta [U(\mathbf{s}^m) - N\mu] - \ln N! - 3N \ln \Lambda + N \ln V}} \quad , \quad (5.26) \\
&= e^{-\beta \delta U_{nm}} e^{-\beta \mu} \frac{N \Lambda^3}{V}
\end{aligned}$$

missä $\delta U_{nm} = U(\mathbf{s}^n) - U(\mathbf{s}^m)$ on potentiaalienergiaero. Määritellään aktiivisuus

$$z = \frac{e^{\beta \mu}}{\Lambda^3} , \quad (5.27)$$

jolloin (5.26) tulee muotoon

$$\frac{\rho_n}{\rho_m} = \exp \left\{ -\beta \delta U_{nm} + \ln \left(\frac{N}{zV} \right) \right\} = e^{-\beta \delta D_{nm}}, \quad (5.28)$$

missä δD_{nm} on *tuhoamisfunktio*:

$$\delta D_{nm} = \delta U_{nm} - \beta^{-1} \ln \left(\frac{N}{zV} \right). \quad (5.29)$$

Yhtälö (5.28) on samanmuotoinen kuin (4.33), joten yritetty tuhoamissiirto hyväksytään (kanonisen ensemblen tapaan) todennäköisyydellä

$$\min(1, e^{-\beta \delta D_{nm}}). \quad (5.30)$$

Samalla lailla yritettäessä luoda atomia on todennäköisyyksiä suhde

$$\begin{aligned} \frac{\rho_n}{\rho_m} &= \frac{e^{-\beta[U(s^n) - (N+1)\mu] - \ln(N+1)! - 3(N+1)\ln\Lambda + (N+1)\ln V}}{e^{-\beta[U(s^m) - N\mu] - \ln N! - 3N\ln\Lambda + N\ln V}} \\ &= e^{-\beta \delta U_{nm}} e^{\beta \mu} \frac{V}{(N+1)\Lambda^3}. \end{aligned} \quad (5.31)$$

Aktiivisuuden avulla lausuttuna suhde on

$$\frac{\rho_n}{\rho_m} = e^{-\beta \delta U_{nm}} \frac{zV}{N+1} = \exp \left\{ -\beta \delta U_{nm} + \ln \left(\frac{zV}{N+1} \right) \right\} = e^{-\beta \delta C_{nm}}, \quad (5.32)$$

missä on määritelty *luomisfunktio* δC_{nm} :

$$\delta C_{nm} = \delta U_{nm} - \beta^{-1} \ln \left(\frac{zV}{N+1} \right). \quad (5.33)$$

Ja tavalliseen tapaan yritetty luomisoperaatio hyväksytään todennäköisyydellä

$$\min(1, e^{-\beta \delta C_{nm}}). \quad (5.34)$$

Toteutettaessa atomin tuhoamisia ja luomisia on muistettava pitää huoli siitä, että yritetodennäköisyysmatriisit α_{mn}^d ja α_{mn}^c ovat symmetrisiä ($\alpha_{mn} = \alpha_{nm}$). Tästä seuraa mikroskooppinen reversiibeliys (4.18). Tämä saavutetaan asettamalla luomisen ja tuhoamisen todennäköisyydet samoiksi.

Kolmen eri operaation yritetodennäköisyyksien (α_{mn}^m , α_{mn}^d , α_{mn}^c) asettaminen samaksi on todettu tehokkaimmaksi Markovin ketjun konvergenssin kannalta, eli

$$\alpha_{mn}^m = \alpha_{mn}^d = \alpha_{mn}^c = 1/3 . \quad (5.35)$$

Algoritmi on siis lopulta muotoa:

- i. Mene satunnaisesti johonkin askeleista ii. - iv.
- ii. Siirrä satunnaisesti atomia i . Hyväksy uusi tila todennäköisyydellä (4.37). Mene askeleeseen v.
- iii. Lisää atomi satunnaiseen paikkaan systeemissä. Hyväksy uusi tila todennäköisyydellä (5.34). Mene askeleeseen v.
- iv. Poista satunnaisesti valittu atomi systeemistä. Hyväksy uusi tila todennäköisyydellä (5.30).
- v. Päivitä keskiarvomuuttujia.
- vi. Mene askeleeseen i.

Tyypillisesti simulaation kuluessa lasketaan potentiaalienergian, paineen ja tiheyden ensemblekeskiarvoja. Eräs suurkanonisen ensemblen eduista on se, että Helmholtzin vapaa energia A pystytään laskemaan suoraan. Kuten termodynamiikasta muistamme suurkanonisen ensemblen termodynaaminen potentiaali on ns. suuri potentiaali Ω

$$\Omega = E - TS - \mu N = A - \mu N . \quad (5.36)$$

Toisaalta

$$\Omega = -PV . \quad (5.37)$$

Näistä saamme yhteyden

$$A = \Omega + \mu N = \mu N - PV , \quad (5.38)$$

eli merkitsemällä eksplisiittisesti ensemblekeskiarvot

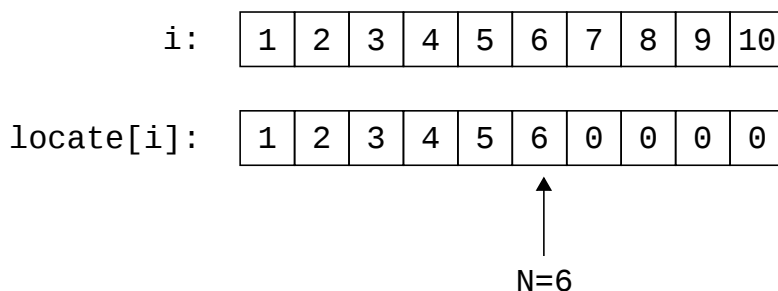
$$\frac{A}{N} = \mu - \frac{\langle P \rangle_{\mu VT} V}{\langle N \rangle_{\mu VT}} . \quad (5.39)$$

Kuten vakiopainesimuloinneissa, myös suurkanonisessa tapauksessa tiheys vaihtelee simuloinnin kuluessa. Tästä syystä myös potentiaalienergian pitkän kantaman korjaukset tulee laskea δU_{mn} :n laskun yhteydessä uudestaan¹. Sama pätee muidenkin suureiden pitkän kantaman korjauksiin.

Ohjelmointiteknisesti suurkanoninen MC-simulointi teettää hieman enemmän työtä johtuen atomien lisäämisestä ja poistamisesta. Atomien kirjanpito voidaan toteuttaa esimerkiksi seuraavaan tapaan².

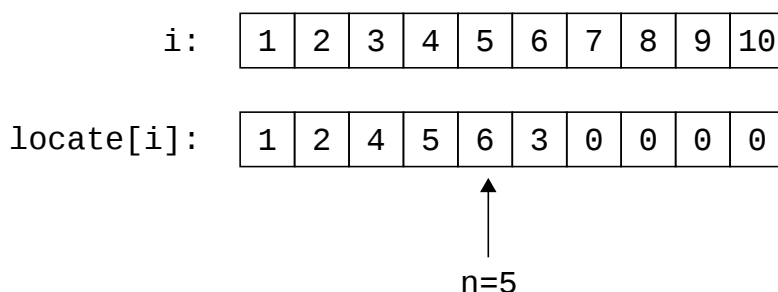
1. Pitkän kantaman korjauksista puhutaan lisää myöhemmin.

Oletetaan, että meillä on alussa 6 atomia ja suurin mahdollinen määrä on 10. Taulukossa `locate` on tieto siitä, mitkä atomit ovat 'elossa' kullakin simulaatioaskeleella. Ensimmäisellä askeleella taulukko näyttää seuraavalta



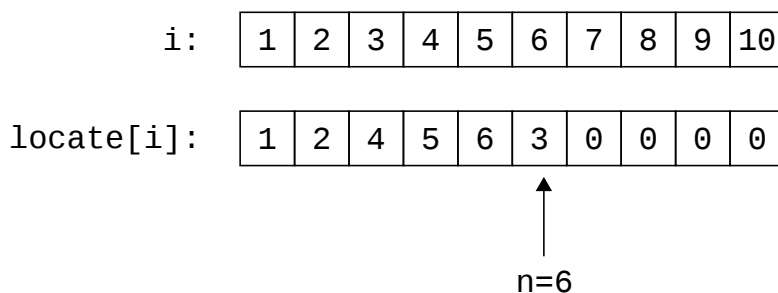
Kuva 5.2: Suurkanonisen simuloinnin atomien kirjanpito.

Jos atomi numero 3 tuhotaan, taulukko päivitetään ja atomi numero 3 siirretään taulukon päähän ('kuolleeseen' osaan):



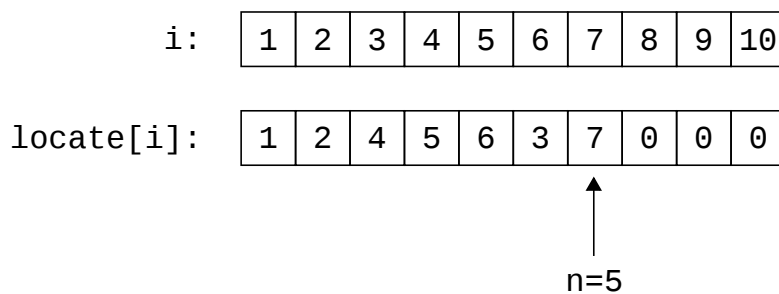
Kuva 5.3: Suurkanonisen simuloinnin atomien kirjanpito (jatkoa).

Jos nyt uusi atomi luodaan, se viedään paikkaan `locate[n+1]`:



Kuva 5.4: Suurkanonisen simuloinnin atomien kirjanpito (jatkoa).

Jos tämän jälkeen taas edelleen luodaan uusi atomi, se viedään paikkaan `locate[n+1]`, ja uuden atomin indeksiksi asetetaan `n+1`:



Kuva 5.5: Suurkanonisen simuloinnin atomien kirjanpito (jatkoa).

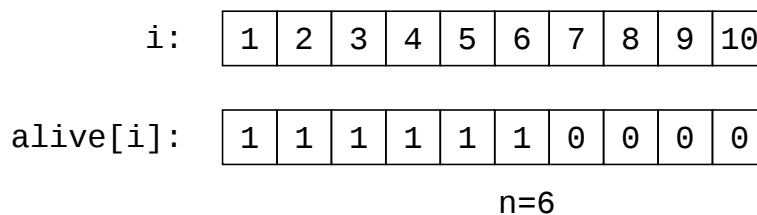
Ohjelman silmukat, jotka käyvät atomit läpi ovat nyt seuraavannäköisiä:

```
for (i=1;i<=n;i++) {
    ia = locate[i];
    rxi = rx[ia];
    ryi = ry[ia];
    rzi = rz[ia];
    /* ... laske haluttuja asioita ... */
}
```

Allen-Tildesleyn esimerkkiohjelmat¹ **F.13** ja **F.14** sisältävät suurkanonisessa simulointiohjelmassa tarvittavia rutiineja.

Katso myös D. Frenkelin ja B. Smitin kirjan WWW-sivun Case Studies -ohjelmia².

Ylläesitellyssä menetelmässä joudutaan taulukko **locate** järjestelemään jokaisen tuhoamisen jälkeen uudestaan. Toinen tapa on käyttää loogista taulukko **alive**, joka kertoo, onko kyseinen atomi systeemissä. Oletetaan samanlainen tilanne kuin aikaisemmin (6 atomia, maksimilukumäärä 10)



Kuva 5.6: Suurkanonisen simuloinnin atomien kirjanpito käyttäen loogista taulukkoa.

Jos atomi numero 3 tuhotaan, asetetaan taulukon **alive**, alkio arvoon **false** (eli 0).

1. <http://www.lce.hut.fi/~akuronen/MDkurssi/allen-tildesley> . Ohjelmat on kirjoitettu Fortran77:lla (valittavasti).
 2. <http://cpcoup5.tn.tudelft.nl/people/smit1/README.html>

i:	1	2	3	4	5	6	7	8	9	10
alive[i]:	1	1	0	1	1	1	0	0	0	0

n=5

Kuva 5.7: Suurkanonisen simuloinnin atomien kirjanpito käyttäen loogista taulukkoa (jatkoa).

Jos nyt uusi atomi luodaan, se viedään ensimmäiseen vapaaseen paikkaan taullukossa **alive**:

i:	1	2	3	4	5	6	7	8	9	10
alive[i]:	1	1	1	1	1	1	0	0	0	0

n=6

Kuva 5.8: Suurkanonisen simuloinnin atomien kirjanpito käyttäen loogista taulukkoa (jatkoa).

Jos tämän jälkeen taas edelleen luodaan uusi atomi, se viedään paikkaan **alive[7]**, ja uuden atomin indeksiksi asetetaan:

i:	1	2	3	4	5	6	7	8	9	10
alive[i]:	1	1	1	1	1	1	1	0	0	0

n=7

Kuva 5.9: Suurkanonisen simuloinnin atomien kirjanpito käyttäen loogista taulukkoa (jatkoa).

Ohjelman silmukat, jotka käyvät atomit läpi ovat nyt seuraavannäköisiä:

```

for (i=1;i<=nmax;i++) {
  if (alive[i]) {
    rxi = rx[ia];
    ryi = ry[ia];
    rzi = rz[ia];
    /* ... laske haluttuja asioita ... */
  }
}

```

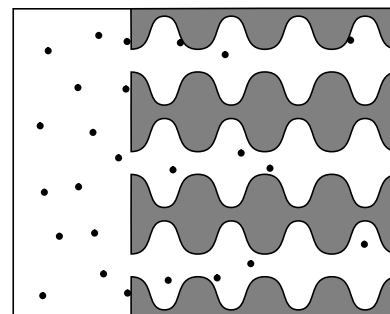
Etuna edelliseen menetelmään on se, että muistia tarvitaan vain yksi looginen taulukko. Jos muisti on rajoittava tekijä voidaan tämä taulukko toteuttaa bittitaulukkona.

Haittana on ylimääräinen ehtolause ohjelman sisimmissä silmukoissa.

Esitetään lopuksi yksi esimerkki suurkanonisen MC-simuloinnin sovelluskohteesta¹.

Adsorptiotutkimuksissa haluamme tietää aineen huokosiin kiinnittyneiden atomien tai molekyylien lukumäärän ulkoisen paineen ja lämpötilan funktiona.

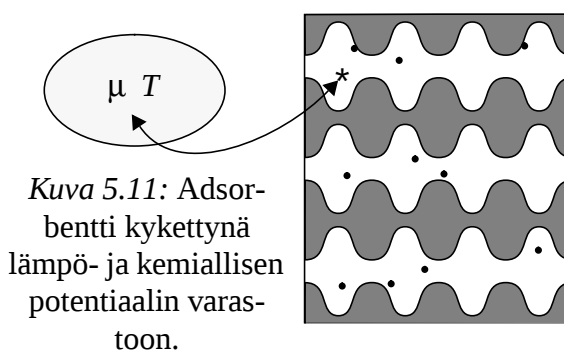
Yksi menetelmä on tietysti molekyyldynamiikka, mutta usein systeemin aikaskaala on makroskooppinen: tasapainotilan saavuttaminen voi vaatia minutteja tai jopa tunteja. Lisäksi kaasufaasin simulaatioon kuluu laskentatehoa, vaikka emme ole siitä kiinnostuneita.



Kuva 5.10: Adsorbentti kaasuatmosfäärissä.

Systeemissä vallitsee tasapaino ympäröivön kaasun ja aineen huokosissa olevien kaasumolekyylien välillä, jolloin niiden kemialliset potentiaalit ja lämpötilat ovat samat. Suurkanonisessa MC-simulaatiossa mallinamme ympäröivää kaasua asettamalla kaasumolekyylien kemiallisen potentiaalin μ ja lämpötilan T haluttuihin arvoihin. Kemiallisen potentiaalin arvo saadaan periaatteessa ulkoisen kaasun paineesta sen tilanyhtälön avulla.

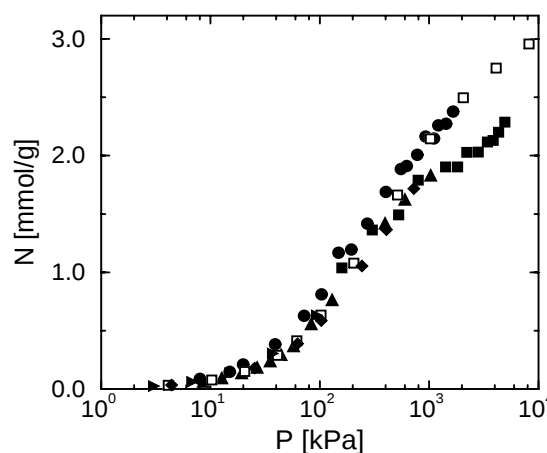
Otetaan esimerkiksi zeoliitit. Ne ovat hyvin huokoisia aineita, joilla on sovelluksia esimerkiksi petrokemian teollisuudessa katalyyttisinä materiaaleina. Ne ovat kiteisiä SiO_4 - ja AlO_4 tetrahedreista rakentuvia materiaaleja, joita nykyään tunnetaan n. 600 erilaista. Sovellusten kannalta on tärkeää tietää, miten ympäristön atomit tai molekyylit kiinnittyvät zeoliitin huokosiin.



Kuva 5.11: Adsorbentti kykettynä lämpö- ja kemiallisen potentiaalin varastoon.

Kuvassa esimerkkinä metaanin adsorptio erään zeoliitin (silikaliitti) pinnalle ulkoisen paineen funktiona.¹ Simulaatiotulokset vastaavat varsin hyvin koetuloksia.

Kuva 5.12: Metaanin adsorptio silikaliittiin metaanin paineen funktiona. Mustat symbolit ovat mitattuja arvoja, avoimet suurkanonisen MC-simulaation tuloksia.



1. Ks. http://www.hpcn.tudelft.nl/frenkel_smit.html

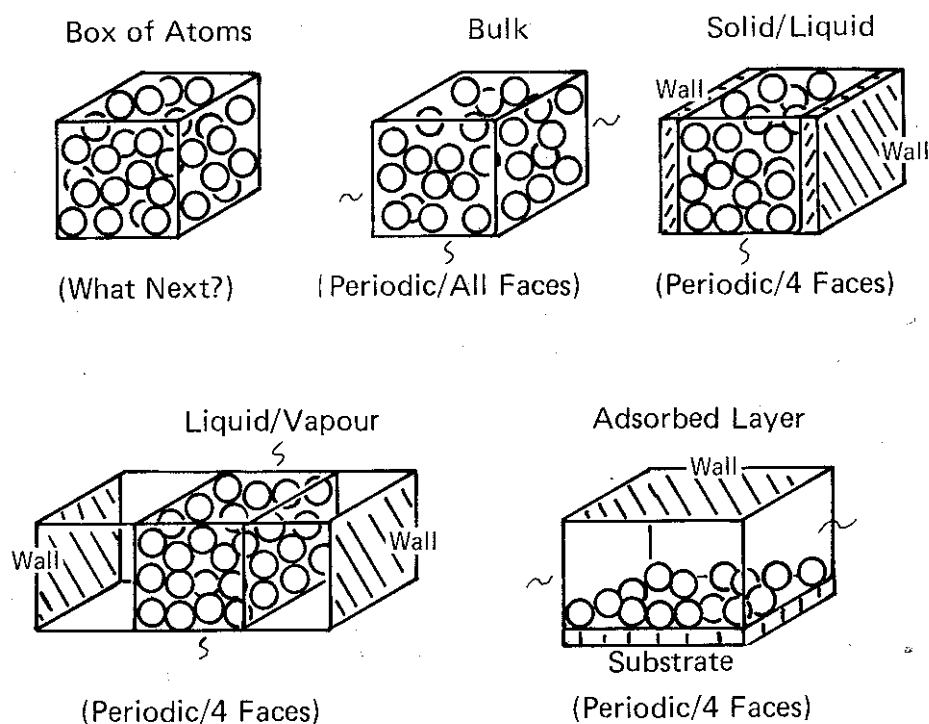
1. Frenkel-Smit kappale 5.6.4

6 Simulaatio-ohjelma käytännössä

Tässä luvussa käymme läpi tiettyjä simulaatioteknisiä asioita kuten reunaehdot, vuorovaikutusten laskeminen tehokkaasti ja niin edelleen. Tarkastellaan edelleen lähinnä atomisysteemejä.

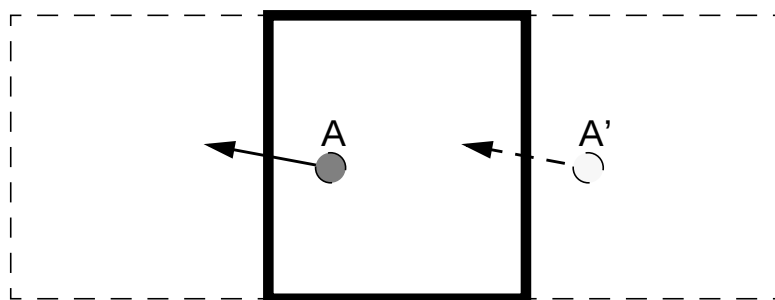
6.1 Reunaehdot

Simuloitava systeemi koostuu luonnollisesti äärellisestä määrästä partikkeleita (simulointikoppi), joten systeemissä on reunat. Mitä tulisi tehdä reunoilla? Se riippuu tietysti tutkittavasta ilmiöstä.



Kuva 6.1: Atomisysteemin reunaehdot.

Jos tutkitaan ääretöntä ainetta (bulk) käytetään ns. periodisia reunaehtoja. Eli kopin reunan yli menevä atomi tulee toiselta puolelta takaisin. Avaruus on siis täytetty simulointikopin kopioilla.



Kuva 6.2: Periodiset reunaehdot.

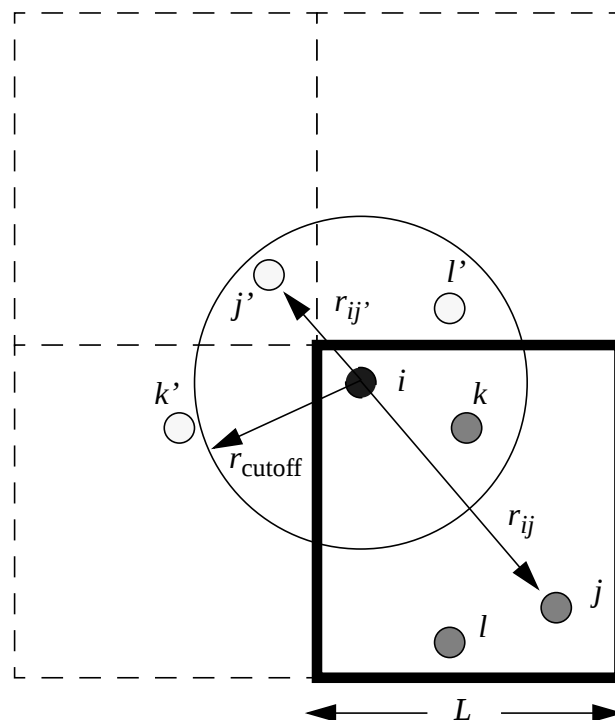
Käytännössä tämä voidaan toteuttaa esimerkiksi seuraavanlaisella C-rutiinilla:

```
void bcond(float *x, int *ibc, float *box)

/* boundary conditions: */
/*      x(3)  : particle coordinate */
/*      ibc   : =1-periodic, >1-free */
/*      box(3) : simulationcell edge */

{
  int j;
  for (j=1;j<=3;j++) {
    if (ibc[j]==1) {
      if (x[j]<0.0)  x[j] = x[j]+box[j];
      if (x[j]>box[j]) x[j] = x[j]-box[j];
    }
  }
}
```

Potentiaalienergian laskussa periodiset reunaehdot tulee myös ottaa huomioon (tarvitaan atomien välisiä etäisyyksiä; ns. *minimum image convention*).



Kuva 6.3: Periodiset reunaehdot potentiaalienergian laskemisessa.

Esimerkiksi kuvan 6.3 tapauksessa, koska $r_{ij} > r_{ij'}$, käytetään atomien i ja j välisenä etäisyytenä vektoria $r_{ij'}$.

C-kielellä asia voidaan hoitaa seuraavasti:

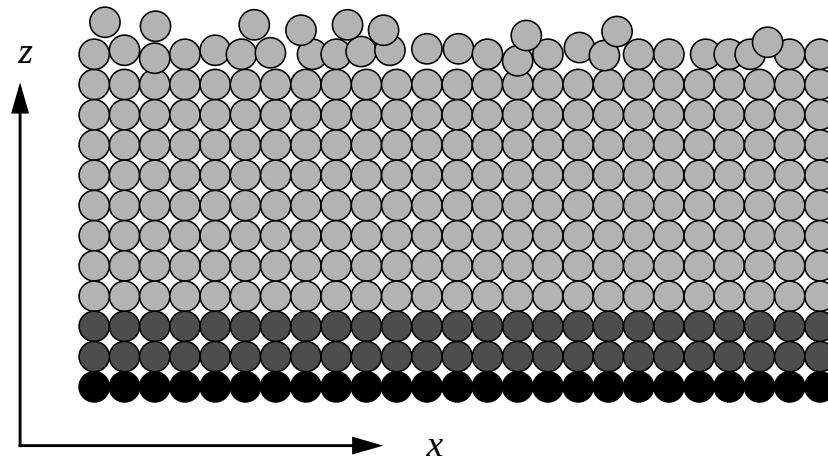
```

if (rijx > box[1]/2.0) rijx=rijx-box[1]
if (rijy > box[2]/2.0) rijy=rijy-box[2]
if (rijz > box[3]/2.0) rijz=rijz-box[3]
if (rijx < -box[1]/2.0) rijx=rijx+box[1]
if (rijy < -box[2]/2.0) rijy=rijy+box[2]
if (rijz < -box[3]/2.0) rijz=rijz+box[3]

```

Edelläolevasta seuraa, että suurin pituus systeemissä on $L/2$. Tämä on otettava huomioon laskehtaessa suureita, jotka ovat atomien välimatkojen funktioita [esim. parikorrelaatiofunktio $g(r)$ tai rakennetekijä $\rho(\mathbf{k})$].

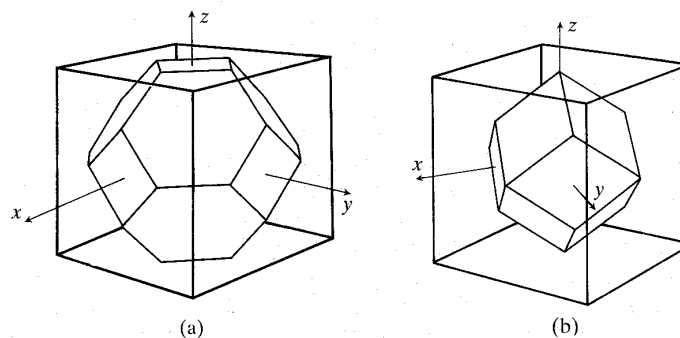
Lisäksi on yleensäkin oltava tarkkana, ettei reunaehtojen aiheuttama periodisuus vaikuta tuloksiin. Esimerkiksi simuloitaessa kidevaurioita on kopin oltava niin suuri, etteivät vaurioiden kuvat pääse vuorovaikuttamaan toistensa kanssa. Tutkittaessa materiaalien pintoja periodisia reunaehtoja käytetään vain x - ja y -suunnissa.



Kuva 6.4: Reunaehdot pintasimulaatiossa.

Systeemin ‘pohja’ voi olla joko vapaa (kalvo) tai kiinnitetty (mallintaa bulkkia).

Yleensä simulaatiokoppi on kuutio tai suorakulmainen. On kuitenkin mahdollista käyttää ei-suorakulmaisia koppeja: esim. monokliininen kiderakenne vaatii tämän. Voidaan käyttää myös muita säännöllisiä (avaruuden täyttäviä) monitahokkaita.



Kuva 6.5: Ei-suorakulmaisia simulointikoppeja.

Esimerkiksi nesteiden simuloinneissa pallomaisempi symmetria voi olla parempi. Periodisten reunaehtojen soveltaminen on tietysti hankalampaa¹.

Olivatpa reunaehdot mitkä hyvänsä, on aina tutkittava niiden mahdollista vaikutusta tuloksiin. Eli suoritetaan simulaatioita erikokoisilla systeemeillä ja toivon mukaan tulokset saturoituvat tiettyyn arvoon, kun koko kasvaa.

6.2 Potentiaalienergian laskeminen

Useimmiten suurin osa tietokoneajasta menee systeemin potentiaalienergian ja voimien² laskemiseen. Tästä syystä yleensä kannattaa keskittyä ohjelman tämän osan optimointiin.

Olkoon meillä paripotentiaali $v(r)$. Systeemin potentiaalienergia on

$$U = \sum_{i=1}^{N-1} \sum_{j=i+1}^N v(r_{ij}), \quad (6.1)$$

$$\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j \quad (6.2)$$

Atomiin i atomin j kohdistama voima on³

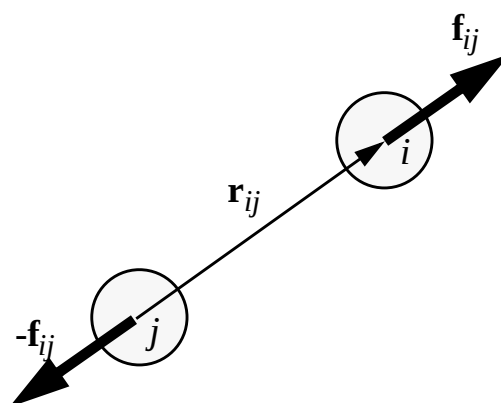
$$\mathbf{f}_{ij} = -\nabla_{\mathbf{r}_i} v(r_{ij}) = -\nabla_{\mathbf{r}_{ij}} v(r_{ij}), \quad (6.3)$$

eli

$$\mathbf{f}_{ij} = -\left[\frac{dv}{dr}\right]_{r=r_{ij}} \times \frac{\mathbf{r}_{ij}}{r_{ij}} \quad (6.4)$$

Käytännössä potentiaalilla on aina katkaisusäde r_c . Atomit joiden välinen etäisyys $> r_c$ eivät vuorovaikuta.

Yleensä $r_c \approx$ muutama Å. Katkaisusäteen valinnan vaikutusta tuloksiin pitäisi aina tutkia.



Kuva 6.6: Paripotentiaalın välityksellä vuorovaikuttavat atomit.

Energiaan (ja muihin laskettaviin suureisiin) voidaan ottaa mukaan pitkän kantaman korjaus (lasketaan olettaen aine jatkuvaksi), joka tarvi-

1. Allen-Tildesleyn esimerkkiohjelmasta F.1 löytyy periodisten reunaehtojen soveltaminen muutamalla ei-suorakulmaiselle kopille. Linkki ohjelmiin löytyy kurssin kotisivulta.
2. Voimia MC-ohjelmassa tarvitaan viriaalin laskemiseen.
3. $\nabla_{\mathbf{r}_{ij}} = \mathbf{i} \frac{\partial}{\partial x_{ij}} + \mathbf{j} \frac{\partial}{\partial y_{ij}} + \mathbf{k} \frac{\partial}{\partial z_{ij}}, \quad \mathbf{r}_{ij} = x_{ij}\mathbf{i} + y_{ij}\mathbf{j} + z_{ij}\mathbf{k}$

taan vuorovaikutuksen katkaisun takia. Esim. Lennard-Jonesille

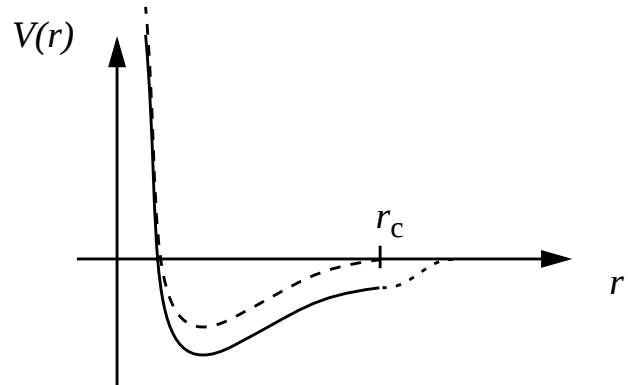
$$E = E_c + E_{\text{LRC}} = E_c + 2\pi\rho \int_{r_c}^{\infty} r^2 V(r) dr \quad (6.5)$$

Katkaisusäteestä johtuen potentiaalilla on epäjatkuvuus r_c :n kohdalla.

Potentiaali voidaan viedä nollaan alueella $[r_c - \delta r, r_c]$, siten, että esim. potentiaali ja voima ovat jatkuvia. (3. asteen polynomi)

Tai potentiaalia voidaan siirtää; energian nollakohtahan on mielivaltainen.

Potentiaalienergia (ja viriaali eli voimat) lasketaan siis kaksinkertaisessa silmukassa:



Kuva 6.7: Potentiaalin katkaisu

```
for (i=1;i<N;i++) {
  for (j=i+1;j<=N;j++) {
    rijx = rx[i]-rx[j];
    rijy = ry[i]-ry[j];
    rijz = rz[i]-rz[j];
    rijsq = rijx*rijx+rijy*rijy+rijz*rijz;
    rij = sqrt(rijsq)
    if (rijsq<rcut) {
      vij = ...potentiaalienergia...;
      wij = ...viriaali...;
      v+=vij;
      w+=wij;
    }
  }
}
```

Potentiaalienergian lasku vie koneaikaa $O(N^2)$ eli ennen pitkää laskenta-aika kasvaa liian suureksi. Avuksi otamme käyttöön Verlet'n naapurilistan ja solumenetelmän.

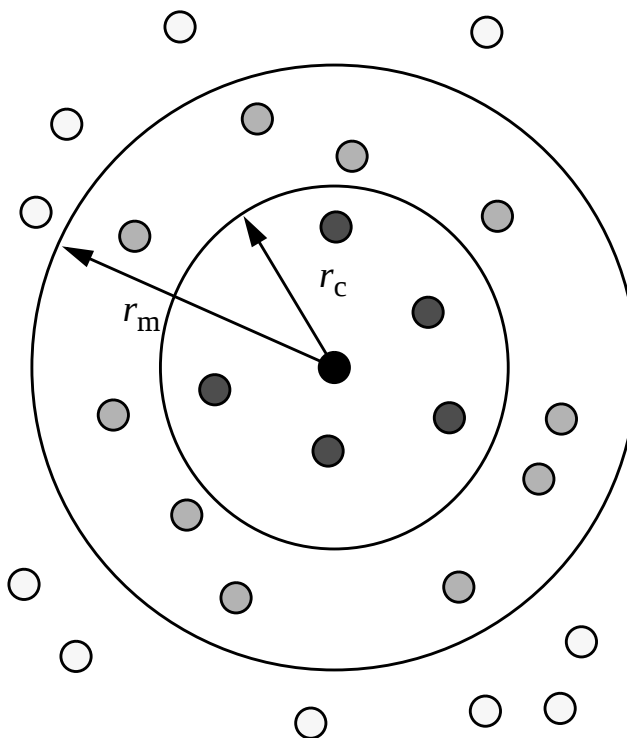
Verlet'n naapurilista on tietorakenne, jossa säilytetään jokaista atomia varten niiden naapurien indeksit, jotka ovat lähempänä kuin tietty etäisyys r_m . Lista päivitetään vain joka N_m :s simulaatioaskel. Parametrit r_m ja N_m valitaan siten, että

$$r_m - r_c > N_m \delta r, \quad (6.6)$$

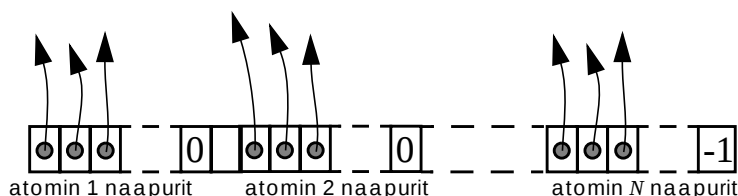
missä δr on tyypillinen atomin poikkeama simulaatioaskeleella.

Käytännössä naapurilista voisi olla kuvan 6.9 mukainen yksiulotteinen taulukko.

Naapurilistaakin käyttämällä simulaatio käyttäytyy kuten $O(N^2)$: Kun N kasvaa, jossain vaiheessa naapurilistan rakentaminen vien huomattavan osan laskenta-ajasta.



Kuva 6.8: Verlet'n naapurilista.



Kuva 6.9: Verlet'n naapurilistan toteutus yksiulotteisena taulukkona

Solunenetelmässä simulaatiokoppi jaetaan osiin. Olkoon meillä $M \times M \times M$ solua.

Etsittäessä atomin i kanssa vuorovaikuttavia naapureita käydään läpi vain naapurisolut. Kaksiulotteisena esimerkkinä otamme kuvan 6.10 mukaisen systeemin. Solussa 13 olevan atomin Verlet'n lista rakennetaan ottaen huomioon vain soluissa 7, 8, 9, 12, 14, 17, 18, ja 19 sijaitsevat atomit.

Solun koko on valittava siten, että

$$l = \frac{L}{M} > r_c, \quad (6.7)$$

missä L = simulointikopin sivun pituus.

21	22	23	24	25
16	17	18	19	20
11	12	13	14	15
6	7	8	9	10
1	2	3	4	5

Kuva 6.10: Solunenetelmä.

Keskimääräinen atomien lukumäärä solussa on $N_c = N/M^3$, mistä seuraa, että on käytävä läpi $27NN_c$ atomiparia $N(N-1):n$ sijaan.

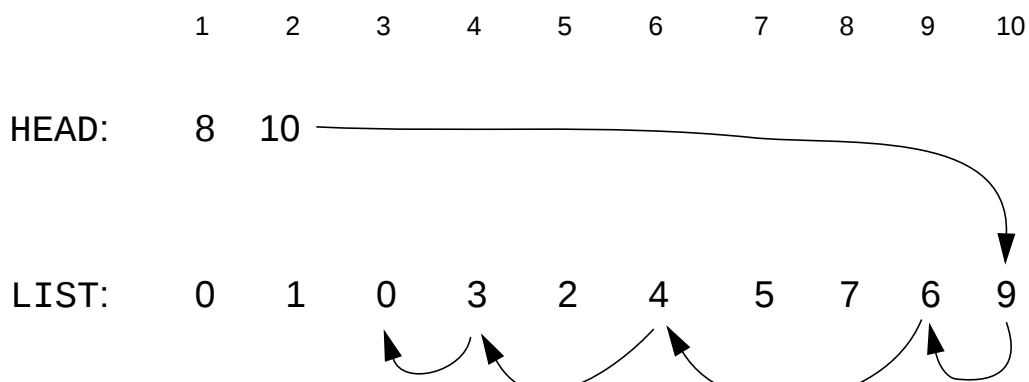
Otetaan esimerkki käytännön toteutuksesta *Allen-Tildesleyn* henkeen¹:

taulukko **HEAD**:

- koko = M^3
- alkio i = osoitin taulukkoon **LIST**
- kertoo solun i atomilistan alun

taulukko **LIST**:

- koko = N
- alkio j kertoo solun seuraavan atomin indeksin.



Kuva 6.11: Solumenetelmän toteutus käyttäen **HEAD**- ja **LIST**-taulukoita.

Listan rakentaminen koodina näyttää seuraavalta²:

MD-kopin sivun pituus = 1.0

solun sivun pituus = $1/M$

```

celli = (float)M;
head = 0;
for (i=1;i<=n;i++) {
    icell = 1 + (int)((rx[i]+0.5)*celli)
              (int)((ry[i]+0.5)*celli) * M
              (int)((rz[i]+0.5)*celli) * M * M;
    list[i] = head[icell];
    head[icell] = i;
}

```

1. Allen-Tildesleyn esimerkkiohjelmassa F.20 on Fortran77:lla tehty toteutus.

2. Simulointikopin sivun pituus on 1.0 ja solun sivun pituus on $1/M$.

6.3 Pitkän kantaman potentiaalit

Yleisesti pitkän kantaman voima on voima, joka pienenee hitaammin kuin r^{-d} , jossa d on systeemin dimensioiden lukumäärä.

Tärkein tapaus on ionien varausten välinen Coulombin vuorovaikutus¹

$$V^{zz}(r_{ij}) = \frac{z_i z_j e^2}{4\pi\epsilon_0 r_{ij}}. \quad (6.8)$$

Ioniyhdisteiden² ionien välistä vuorovaikutusta mallinnetaan usein käyttämällä Coulombin vuorovaikutusta, lyhyen kantaman repulsiota (SR) sekä van der Waalsin attraktiota (vdW):

$$V(r_{ij}) = V^{zz}(r_{ij}) + V^{SR}(r) + V^{vdW}(r), \quad (6.9)$$

$$V^{SR}(r) = Ae^{-r/\rho}, \quad V^{vdW}(r) = -\frac{C}{r^6}. \quad (6.10)$$

Koska Coulombin potentiaalin kantama on pitkä, emme voi käyttää normaalia katkaisusädettä $r_c \approx$ muutama Å, eikä systeemin kokoa voi kasvattaa niin suureksi, että voisimme turvallisesti katkaista potentiaalin. Lisäksi potentiaalin katkaisu voisi tehdä systeemistä varatun.

Yleisin ratkaisu tähän ongelmaan on käyttää ns. Ewaldin summaa.

Otetaan huomioon kaikkien periodisten kuva-atomien vuorovaikutukset (unohdetaan $4\pi\epsilon_0$, jotta olisi vähemmän kirjoitettavaa). Coulombin vuorovaikutuksesta johtuva systeemin potentiaalienergia on siten

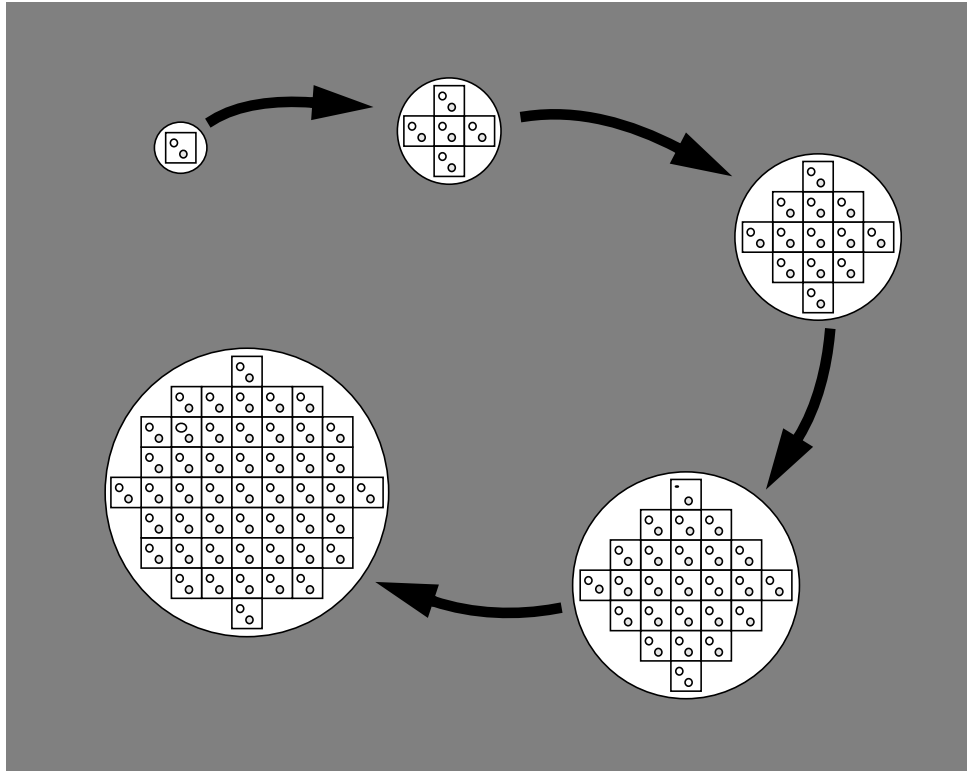
$$V^{zz} = \frac{1}{2} \sum_{\mathbf{n}} \left[\sum_{i=1}^N \sum_{j=1}^N \frac{z_i z_j}{|\mathbf{r}_{ij} - \mathbf{n}|} \right] \quad (6.11)$$

Tässä vektori $\mathbf{n}$ käy läpi kaikki simulaatiokopin kuvat ja indeksit i ja j atomit kopin sisällä. Huomaa, että $\mathbf{n}$ -summassa ei oteta mukaan termiä $i = j$, kun $\mathbf{n} = 0$.

Tämä summa on vain ehdollisesti suppeneva: tulos riippuu siitä, missä järjestyksessä termejä summaamme.

Luonnollinen valinta on lisätä säteittäisesti koppeja origon ympärille (ks. kuva 6.12).

-
1. Käytämme potentiaalienergian symbolina nyt kirjainta V , koska tässä kappaleessa sitä ei voi sotkea tilavuuteen.
 2. Suolat ja monet oksidit.



Kuva 6.12: Systeemin rakentaminen simulointikopeista.

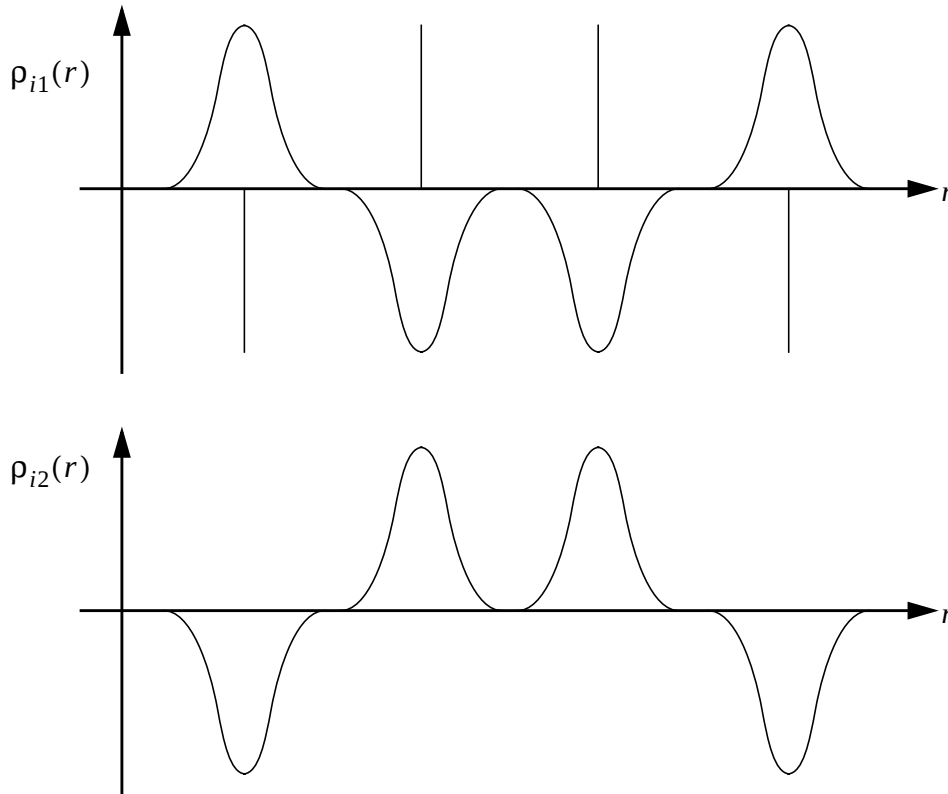
Ympäröivä aine vaikuttaa energiaan. Metallille ($\epsilon = \infty$) ja tyhjiölle ($\epsilon = 1$) tulokset ovat erilaiset; tyhjiössä pallon pinnalla on dipolikerros:

$$V^{zz}(\epsilon = \infty) = V^{zz}(\epsilon = 1) - \frac{2\pi}{3L^3} \left| \sum_i z_i \mathbf{r}_i \right|^2. \quad (6.12)$$

Ewaldin summalla voimme laskea $V^{zz}(\epsilon = \infty)$:n.

Jos haluamme systeemimme olevan tyhjiössä, voimme lisätä tuon dipolitermin energiaan.

Ewaldin summamenetelmässä ilmaistaan varaukset z_i varaustiheyden ρ_i^z muodossa.



Kuva 6.13: Ionien varauksen ilmaiseminen varausjakauman avulla Ewaldin summassa.

Jokaisen ionin varaus lausutaan summana gaussisesta ja deltapiikkitiheydestä:

$$\rho_i^z(\mathbf{r}) = \rho_{i1}^z(\mathbf{r}) + \rho_{i2}^z(\mathbf{r}) \quad (6.13)$$

$$\rho_{i1}^z(\mathbf{r}) = z_i [\delta(\mathbf{r} - \mathbf{r}_i) - \kappa^3 \pi^{-3/2} e^{-\kappa^2(\mathbf{r} - \mathbf{r}_i)^2}] \quad (6.14)$$

$$\rho_{i2}^z(\mathbf{r}) = z_i \kappa^3 \pi^{-3/2} e^{-\kappa^2(\mathbf{r} - \mathbf{r}_i)^2} . \quad (6.15)$$

Koska nyt ρ_{i1}^z :llä on äärellinen kantama voimme laskea siitä aiheutuvan energian ja voiman soveltaen katkaisusädettä.

Johtuen ρ_{i2}^z :n 'tasaisuudesta' se kannattaa laskea käänteisavaruudessa: atomien ρ :n Fourier-muunnokset summataan ja tästä otetaan käänteinen Fourier-muunnos.

Eli lopputulokseksi (*after a few steps of algebra*) saadaan lauseke, jossa on $\mathbf{r}$ -avaruuden termi, $\mathbf{k}$ -avaruuden termi sekä itseisenergian vastaluku ja pintatermi:

$$V^{zz}(\epsilon = 1) = V_{\text{real}}^{zz} + V_{\text{recip}}^{zz} - V_{\text{self}}^{zz} + V_{\text{surf}}^{zz} \quad , \quad (6.16)$$

$$= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \left[\sum_{|\mathbf{n}|=0}^{\infty} z_i z_j \frac{\text{erfc}(\kappa |\mathbf{r}_{ij} + \mathbf{n}|)}{|\mathbf{r}_{ij} + \mathbf{n}|} \right. \\ \left. + \frac{1}{\pi L^3} \sum_{\mathbf{k} \neq 0} z_i z_j \frac{4\pi^2}{k^2} e^{-k^2/4\kappa^2} \cos(\mathbf{k} \cdot \mathbf{r}_{ij}) \right] \\ - \frac{\kappa}{\pi^{1/2}} \sum_{i=1}^N z_i^2 \\ + \frac{2\pi}{3L^3} \left| \sum_{i=1}^N z_i \mathbf{r}_i \right|^2$$

missä

$$\text{erfc}(x) = \frac{2}{\pi^{1/2}} \int_x^{\infty} e^{-t^2} dt \quad (6.17)$$

on komplementäärinen virhefunktio ja L on simulointikopin sivun pituus. Oletamme tässä, että koppi on kuutiollinen. Taaskaan $\mathbf{n}$ -summassa ei oteta mukaan termiä $i = j$, kun $\mathbf{n} = 0$.

Toinen termi on summa yli käänteisavaruuden vektoreiden $\mathbf{k} = (2\pi/L)\mathbf{n}$. Toiseksi viimeinen termi on ρ_{i2}^z :n itseisenergia, joka on vähennettävä, koska se lasketaan mukaan V_{recip}^{zz} -osassa.

Asettamalla κ tarpeeksi suureksi voimme rajoittaa vain termiin $\mathbf{n} = 0$, joka vastaa normaalia 'minimum image'-konventiota.

Reaaliavaruuden termi voidaan laskea samassa silmukassa kuin lyhyen kantaman voimat. Tällöin on V_{real}^{zz} muotoa

$$V_{\text{real}}^{zz} = \sum_{i < j} \left[z_i z_j \frac{\text{erfc}(\kappa r_{ij})}{r_{ij}} \right] \quad (6.18)$$

Kompleksilukuja käyttäen käänteisavaruuden termi tulee yksinkertaisempaan muotoon

$$V_{\text{recip}}^{zz} = \sum_{\mathbf{k}} A(\mathbf{k}) \left| \sum_i z_i e^{i\mathbf{k} \cdot \mathbf{r}_i} \right|^2 \quad (6.19)$$

$$A(\mathbf{k}) = \frac{2\pi \exp(-k^2/4\kappa^2)}{L^3 k^2} \quad (6.20)$$

Vastaavasti tästä aiheutuva voima on ioniin i on

$$\mathbf{f}_{\text{recip}}^i = -2z_i \sum_{\mathbf{k} \neq 0} \mathbf{k} A(\mathbf{k}) \text{Im} \left\{ e^{-i\mathbf{k} \cdot \mathbf{r}_i} \sum_j z_j e^{i\mathbf{k} \cdot \mathbf{r}_j} \right\} \quad (6.21)$$

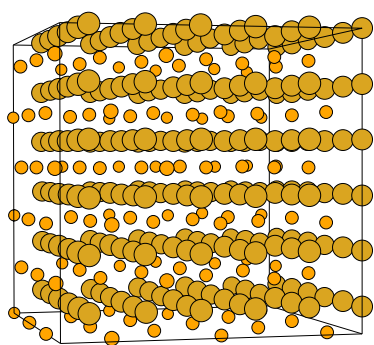
Huomaa, että potentiaalienergian (ja voimien) lasku vie nyt aikaa kuten $O(N^2)$. Lausekkeet on helppo yleistää ei-kuutiollisiin koppeihin.

Valittavana on nyt kolme potentiaalienergian (ja voimien) laskuun liittyvää parametria:

- i. katkaisusäde r_c
- ii. varausjakauman terävyys κ
- iii. $\mathbf{k}$ -summauksen yläraja $|\mathbf{k}|_{\text{max}}^2$

Parametrien valinnassa lienee hyvä aloittaa asettamalla r_c suurehkoksi: esim. $L/2$. Tästä saadaan sopiva κ :n arvo, jonka perusteella lasketaan sopiva $\mathbf{k}$ -summauksen raja. Tyypillisesti $\kappa \sim 5/L$, jolloin laskenta painottuu $\mathbf{k}$ -avaruuteen. $\mathbf{k}$ -summaukseen otetaan mukaan 100-200 vektoria.

Otetaan esimerkiksi ioniyhdiste EuF_2 .



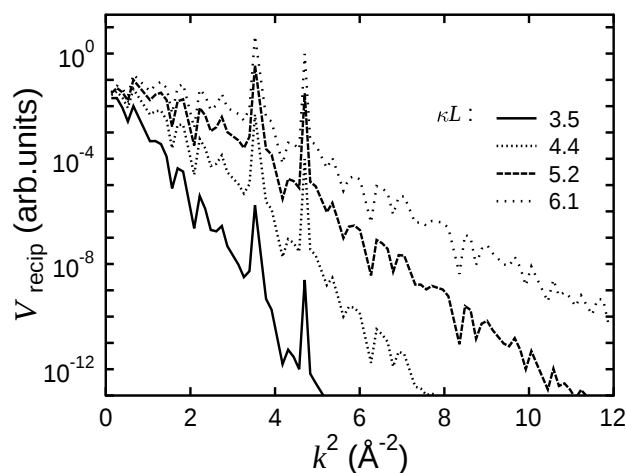
Kuva 6.15: EuF_2 :n kiderakenne.

Suuret pallot ovat fluoriatomeja. Asiaan liittyvät simulointiparametrit olivat:

$$N = 324, \quad L = 17.4 \text{ \AA}$$

$$\kappa L = 5.2, \quad r_c = L/2,$$

$$|\mathbf{k}|_{\text{max}}^2 = 5 \text{ \AA}^{-2}$$



Kuva 6.14: Europiumdifluoridisysteemin potentiaalienergia. Piirrettynä on energian käänteisavaruusosuus neljälle eri parametrin κ arvolle.

Jos Ewaldin summan aiheuttama periodisuus tuottaa ongelmia, voidaan käyttää hiukkas-hilamenetelmää¹. Käänteisavaruuden osuus lasketaan tasoittamalla ionien varaukset säännölliseen hilaan ja ratkaisemalla potentiaali Poissonin yhtälöstä $\nabla^2\phi = -\rho/\epsilon_0$ Fourier-menetelmillä. Etuna on myös laskennan käyttäytyminen kuten $O(N)$. Haittana ohjelman monimutkaistuminen.

Reaktiokenttämenetelmässä² atomista etäisyyttä r_c kauempana olevat naapurit approksimoidaan jatkuvaksi aineeksi (ϵ_s). Onkalon sisällä potentiaalienergia (ja voimat) lasketaan normaaliin tapaan. Ongelmana on tuo ϵ_s , joka tulee tietää etukäteen.

6.4 Alkuehdot

Vaikka simulointialgoritmi periaatteessa vie systeemin tasapainotilaan riippumatta alkutilasta, kannattaa alkutilan valinta tehdä järkevästi.

Jos tutkimme kiinteää kiteistä ainetta, on luonnollisin valinta kiderakenteen määräämät paikat. Termisen tasapainon saavuttamista voi hieman nopeuttaa poikkeuttamalla atomeja satunnaisesti kidepaikoiltaan. Poikkeama on yleensä gaussisesti jakautunut ja jakauman leveys voidaan arvioida materiaalin Debyen lämpötilan avulla.

Esimerkiksi atomien asettaminen sivukeskisen kuutiorakenteen määräämiin paikkoihin voidaan tehdä seuraavalla C-rutiinilla:

```
#include <stdio.h>
#include <math.h>
#include "mcnvt.h"
/* Lattice constant of Ne */
#define A 4.43

void createlattice(int ix, int iy, int iz, int *npart,
                  double **x, double **y, double **z, double *box)
{
    int i,j,k,iat;
    double xo,yo,zo,o1;

    *npart=ix*iy*iz*4; /* 4 atoms in a unit cell in FCC */
    *x= (double *) malloc((size_t) ((*npart) *sizeof(double)));
    *y= (double *) malloc((size_t) ((*npart) *sizeof(double)));
    *z= (double *) malloc((size_t) ((*npart) *sizeof(double)));
    box[0]=ix*A;
    box[1]=iy*A;
    box[2]=iz*A;

    o1=0.50*A;
    iat=-1;
    for (i=0;i<ix;i++) {
        xo=A*i;
        for (j=0;j<iy;j++) {
```

-
1. R.W.Hockney, J.W.Eastwood, *Computer Simulation Using Particles*, Adam Hilger, 1988, kappale 8.
 2. Allen-Tildesley, kappale 5.5.3.

```

yo=A*j;
for (k=0;k<iz;k++) {
    zo=A*k;
    iat++; (*x)[iat]=xo ; (*y)[iat]=yo ; (*z)[iat]=zo;
    iat++; (*x)[iat]=xo+o1 ; (*y)[iat]=yo+o1 ; (*z)[iat]=zo;
    iat++; (*x)[iat]=xo+o1 ; (*y)[iat]=yo ; (*z)[iat]=zo+o1;
    iat++; (*x)[iat]=xo ; (*y)[iat]=yo+o1 ; (*z)[iat]=zo+o1;
}
}
}
*npart=iat+1;
}

```

Rutiinin kutsu on seuraavanlainen:

```

#define double double
...
int npart,ix,iy,iz;
double *x,*y,*z,T,box[3];
...
createlattice(ix,iy,iz,&npart,&x,&y,&z,box);

```

Nestemäisen aineen atomien alkupaikoiksi voidaan myös asettaa kyseessä olevan aineen kiteisen muodon kidepaikat. Simulaation alkuvaiheessa korkea lämpötila pitää huolen kiderakenteen rikkoontumisesta. Periaatteessa mikä muu kiderakenne käy myös, kunhan aineen tiheys asetetaan oikeaksi.

Amorfinen materiaali saadaan aikaan simuloimalla jonkin aikaa nestettä ja tämän jälkeen laske-
malla lämpötilaa¹jolloin saavutetaan amorfinen tila.

Huomaa, että nesteen ja amorfisen aineen tapauksessa ei atomien paikkojen satunnainen arpo-
minen simulointikopin sisällä ole kovin hyvä ratkaisu. Tällä tavalla syntyvällä tilalla voi hel-
posti olla suuri potentiaalienergia ja tasapainon saavuttamiseen kuluu paljon
simulointiaskeleita.

Koska tasapaino on sitten saavutettu? Kriteerinä voi käyttää tiettyjen suureiden asettumista
alkutransientin jälkeen. Esimerkiksi simuloitaessa kiinteää ainetta voimme tarkastella atomien
keskimääräistä poikkeamaa alkuperäisiltä (kide)paikoiltaan $\sqrt{\langle (\mathbf{r}_i - \mathbf{r}_0)^2 \rangle}$.

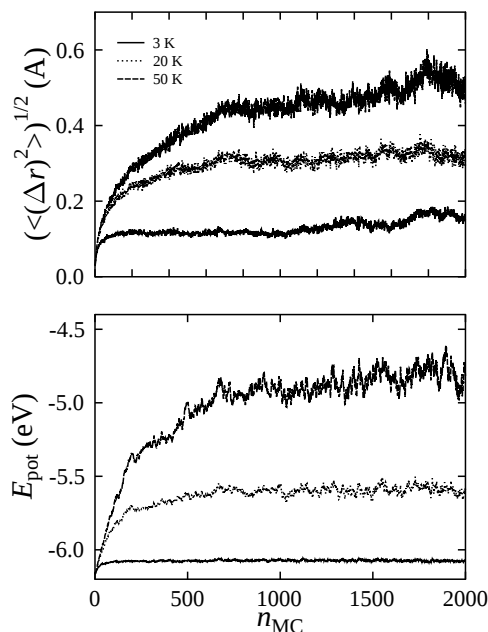
1. Riittävän nopeasti, sillä amorfinen tila on yleensä metastabiili.

Kuvassa 6.16 on esitetty, miten 256 atomin Lennard-Jones-systeemin¹ atomien keskimääräinen poikkeama ja potentiaalienergia käyttäytyvät simulaation alussa eri lämpötiloissa. Lämpötilassa 50 K systeemi voi olla jo nestemäinen, joten atomien keskimääräinen poikkeama ei asetu tiettyyn arvoon, vaan jatkaa diffuusionomaista käyttäytymistä.

Nesteen tapauksessa tasapainon saavuttaminen vie kauemmin kuin kiinteän aineen tapauksessa. Tällöin hyvä kriteeri on käänteishilavektorille $\mathbf{K}$ lasketun rakennetekijän

$$\rho(\mathbf{K}) = \sum_{j=1}^N e^{i\mathbf{K} \cdot \mathbf{r}_j} \quad (6.22)$$

asettuminen arvoon, joka on paljon ykköstä pienempi².



Kuva 6.16: Lennard-Jones-systeemin MC-simuloinnin alku.

6.5 Tulosten analysoinnista

Simulaation kuluessa laskemme erinäisistä suureista keskiarvoja. Näille keskiarvoille tulisi pystyä arvioimaan jonkinlaisia epätarkkuuksia.

Tuloksissa voi olla systemaattisia virheitä: äärellisen koon vaikutus (*finite size effects*), huonon satunnaislukugeneraattorin vaikutus, väärät simulaatioparametrit jne. Näiden vaikutuksen arvioiminen on tietysti tärkeää.

Lisäksi, koska simulaatiossa otetaan äärellinen otos, on tuloksissa statistista epätarkkuutta.

Useimmiten voimme olettaa laskiessamme ensemblekeskiarvon $\langle A \rangle$ tai sen fluktuaatioiden $\langle (\delta A)^2 \rangle$ epätarkkuutta, että prosessi $A(t)$ on gaussinen. Tämä tarkoittaa sitä, että A :n jakauman määrää täysin sen kaksi ensimmäistä momenttia: keskiarvo ja varianssi.

Eli yleisesti

1. Systeemin kuvaa neonia.

2. Käänteishilavektorille $\exp(i\mathbf{K} \cdot \mathbf{r}_i) = 1$ eli täydelliselle kiteelle $\rho(\mathbf{K}) = N$.

$$\langle \delta A(t_1) \delta A(t_2) \dots \delta A(t_n) \rangle, \quad (6.23)$$

$$= \begin{cases} 0, & n \text{ pariton} \\ \sum_{\text{parit}} \langle \delta A(t_i) \delta A(t_j) \rangle \langle \delta A(t_k) \delta A(t_l) \rangle \dots, & n \text{ parillinen} \end{cases}$$

missä summa käy läpi kaikki mahdolliset parit ajanhetkistä¹ $t_1, t_2, \dots, t_n$. Ylläolevassa yhtälössä on argumenttina aika, mutta se pätee myös MC-simulaatioihin, jolloin ajan tilalla on simulaatioaskel.

Oletus gaussisuudesta perustuu keskeisraja-arvoteoreemaan, josta puhuttiin aikaisemmin luennoilla (kappale 3.2). Sen mukaan satunnaismuuttujan, joka on summa monesta satunnaismuuttujasta, jakauma on lähellä gaussista, kun summan termien lukumäärä on suuri. Tämä pätee riippumatta summan termien jakaumista. Ehtona on vain yksittäisten jakaumien momenttien olemassaolo.

Tästä syystä simulaatiotulosten, jotka ovat systeemin ‘kollektiivisia’ ominaisuuksia (eli ovat summia yli atomien), voidaan olettaa noudattavan gaussista jakaumaa. Tietysti on suureita, joiden jakauma on täsmälleen gaussinen: esimerkiksi yksittäisten atomien nopeus, joka noudattaa Boltzmannin jakaumaa.

Eli ongelmaksi jää pitkästä, mutta äärellisestä simulaatioajasta lasketun keskiarvon varianssin arvioiminen.

Huomaa, että suureen A ensemblekeskiarvo $\langle A \rangle_{\text{ens}}$ on tarkasti määritelty huolimatta siitä, että suureella on fluktuaatioita $\sigma^2(A)$:

$$\sigma^2(A) = \langle \delta A^2 \rangle_{\text{ens}} = \langle A^2 \rangle_{\text{ens}} - \langle A \rangle_{\text{ens}}^2, \quad (6.24)$$

$$\delta A = A - \langle A \rangle_{\text{ens}}. \quad (6.25)$$

Se mitä tässä ajamme takaa on keskiarvo $\langle A \rangle_{\text{ens}}$ approksimaation $\langle A \rangle_{\text{run}}$ varianssi. Voimme ajatella yhtä laskettua keskiarvoa $\langle A \rangle_{\text{run}}$ otoksena äärellisistä simulaatiodatoista laskettujen keskiarvojen joukoista.

Oletetaan, että ajon pituus on n_t simulaatioaskelta. Suureen A keskiarvo $\langle A \rangle_{\text{run}}$ (ajokeskiarvo) saadaan tietysti seuraavasti

-
1. Monte Carlo -simulaatioiden yhteydessä aika tarkoittaa simulaatioaskelta, Markovin prosessin indeksii. Tietyllä tavalla voimme ajatella ajan olevan mukana myös MC-simulaatiossa. Generoimalla Markovin ketjua ratkaisemme masteryhtälöä

$$\frac{dP(\mathbf{r}, t)}{dt} = - \sum_{\mathbf{r}'} W(\mathbf{r} \rightarrow \mathbf{r}') P(\mathbf{r}, t) + \sum_{\mathbf{r}'} W(\mathbf{r}' \rightarrow \mathbf{r}) P(\mathbf{r}', t).$$

Tämä aika ei kuitenkaan ole sama kuin systeemin liikeyhtälöiden (jos sellaiset ovat olemassa) kuvaama aika.

$$\langle A \rangle_{\text{run}} = \frac{1}{n_t} \sum_{i=1}^{n_t} A_i, \quad (6.26)$$

missä A_i on suureen A arvo simulaatioaskeleella i .

Entä tämän varianssi? Se voidaan kirjoittaa seuraavaan muotoon

$$\begin{aligned} \langle (\delta A)^2 \rangle &= \left\langle \left[\frac{1}{n_t} \sum_{i=1}^{n_t} (A_i - \langle A \rangle) \right]^2 \right\rangle \\ &= \frac{1}{n_t^2} \sum_{i=1}^{n_t} \langle (A_i - \langle A \rangle)^2 \rangle + \frac{2}{n_t^2} \sum_{i \neq j} \langle (A_i - \langle A \rangle)(A_j - \langle A \rangle) \rangle \\ &= \frac{1}{n_t^2} \sum_{i=1}^{n_t} \langle (A_i - \langle A \rangle)^2 \rangle + \frac{2}{n_t^2} \sum_{i=1}^{n_t-1} \sum_{j=i+1}^{n_t} (\langle A_i A_j \rangle - \langle A \rangle^2) \end{aligned} \quad (6.27)$$

Tässä kulmasulut tarkoittavat ensemblekeskiarvoa. Ylläoleva yhtälö voidaan summausindeksiä muuttamalla kirjoittaa muotoon

$$\langle (\delta A)^2 \rangle = \frac{1}{n_t} \left[\langle A^2 \rangle - \langle A \rangle^2 + 2 \sum_{i=1}^{n_t} \left(1 - \frac{i}{n_t} \right) (\langle A_0 A_i \rangle - \langle A \rangle^2) \right]. \quad (6.28)$$

Tämän voimme muuntaa integraaliksi merkitsemällä muuttamalla summausindeksin jatkuvaksi aikamuuttujaksi $t = i \cdot \delta t$, missä δt on peräkkäisten konfiguraatioiden välinen aika:

$$\begin{aligned} \langle (\delta A)^2 \rangle &= \frac{1}{t_n} \left[\langle A^2 \rangle - \langle A \rangle^2 + \frac{2}{\delta t} \int_0^{t_n} \left(1 - \frac{t}{t_n} \right) [\langle A(0)A(t) \rangle - \langle A \rangle^2] dt \right] \\ &= \frac{1}{t_n} [\langle A^2 \rangle - \langle A \rangle^2] \left[1 + \frac{2}{\delta t} \int_0^{t_n} \left(1 - \frac{t}{t_n} \right) \frac{\langle A(0)A(t) \rangle - \langle A \rangle^2}{\langle A^2 \rangle - \langle A \rangle^2} dt \right] \end{aligned} \quad (6.29)$$

Tässä integroinnin yläraja $t_n = n \cdot \delta t$. Määritellään suureen A normitettu autokorrelaatiofunktio¹

$$\phi_A(t) \equiv \frac{\langle A(0)A(t) \rangle - \langle A \rangle^2}{\langle A^2 \rangle - \langle A \rangle^2}, \quad (6.30)$$

jolloin yhtälö (6.29) tulee muotoon

1. Joissain teoksissa käytetään termiä relaksaatiofunktio.

$$\langle (\delta A)^2 \rangle = \frac{1}{n} [\langle A^2 \rangle - \langle A \rangle^2] \left[1 + \frac{2}{\delta t} \int_0^{t_n} \left(1 - \frac{t}{t_n} \right) \phi_A(t) dt \right]. \quad (6.31)$$

Autokorrelaatiofunktiolle pätee¹:

$$\phi_A(0) = 1, \quad \lim_{t \rightarrow \infty} \phi_A(t) = 0. \quad (6.32)$$

Suureen A korrelaatioaika τ_A määritellään seuraavasti

$$\tau_A \equiv \int_0^{\infty} \phi_A(t) dt. \quad (6.33)$$

Korrelaatioaika kertoo aikaskaalan, jossa korrelaatiofunktion arvo on hyvin lähellä nollaa.

Jos oletamme, että otoksemme on niin suuri, että $t_n \gg \tau_A$, voimme unohtaa termin t/t_n integraalissa (6.31) ja ulottaa integraalin äärettömyyteen:

$$\langle (\delta A)^2 \rangle = \frac{1}{n} [\langle A^2 \rangle - \langle A \rangle^2] \left[1 + \frac{2}{\delta t} \int_0^{\infty} \phi_A(t) dt \right]. \quad (6.34)$$

Eli saamme lopulta

$$\langle (\delta A)^2 \rangle = \frac{1}{n} [\langle A^2 \rangle - \langle A \rangle^2] \left[1 + \frac{2\tau_A}{\delta t} \right]. \quad (6.35)$$

Jos peräkkäisten konfiguraatioiden välinen aika δt on paljon suurempi kuin korrelaatioaika $\delta t \gg \tau_A$, saadaan tuttu korreloimattomalle datalle pätevä tulos

$$\langle (\delta A)^2 \rangle \approx \frac{1}{n} [\langle A^2 \rangle - \langle A \rangle^2]. \quad (6.36)$$

Päinvastaisessa tapauksessa $\delta t \ll \tau_A$, saadaan virhe muotoon

$$\langle (\delta A)^2 \rangle \approx \frac{2\tau_A}{n\delta t} [\langle A^2 \rangle - \langle A \rangle^2] = \frac{2\tau_A}{\tau_{\text{obs}}} [\langle A^2 \rangle - \langle A \rangle^2], \quad (6.37)$$

missä τ_{obs} on keskiarvon laskemiseen käytetty aikajakso. Eli virhe on riippumaton peräkkäisten konfiguraatioiden välisestä ajasta. Jos pidämme ajan τ_{obs} samana, mutta lisäämme otoksen suuruutta pienentämällä väliä δt , emme silti saavuta mitään: statistinen virhe pysyy samana. Vain suhde τ_A/τ_{obs} merkitsee.

1. Esimerkkinä autokorrelaatiofunktioista ks. kuva 4.15.

Eli periaatteessa simuloinneista laskettujen ensemblekeskiarvojen virheen laskemiseksi olisi laskettava kyseisen suureen korrelaatioaika ja siten autokorrelaatiofunktio. Tässä on ylimääräistä työtä melko paljon, joten tehtävään on kehitetty muita menetelmiä.

Ns. blokkimenetelmässä lasketaan simulointidatasta τ_B :n pituisia keskiarvoja

$$\langle A \rangle_B = \frac{1}{\tau_B} \int_0^{\tau_B} A(t) dt . \quad (6.38)$$

Blokkikeskiarvot eripituisilla blokeilla voidaan helposti laskea simuloinnin kuluessa tai jälkeensä raakadatasta. Blokkikeskiarvo varianssi on

$$\sigma_B^2(A) = \frac{1}{n_B} \sum_{b=1}^{n_B} (\langle A \rangle_B - \langle A \rangle)^2 , \quad (6.39)$$

missä n_B on blokkien lukumäärä. Jos τ_B on paljon suurempi kuin korrelaatioaika τ_A ja lisäksi näytteitä blokkikeskiarvoon on otettu niin tiheään, että $\delta t \ll \tau_A$, pätee tällöin

$$\sigma_B^2(A) \approx \frac{2\tau_A}{\tau_B} [\langle A^2 \rangle - \langle A \rangle^2] . \quad (6.40)$$

Emme kuitenkaan tiedä korrelaatioaikaa τ_A . Siksi laskemme suureen

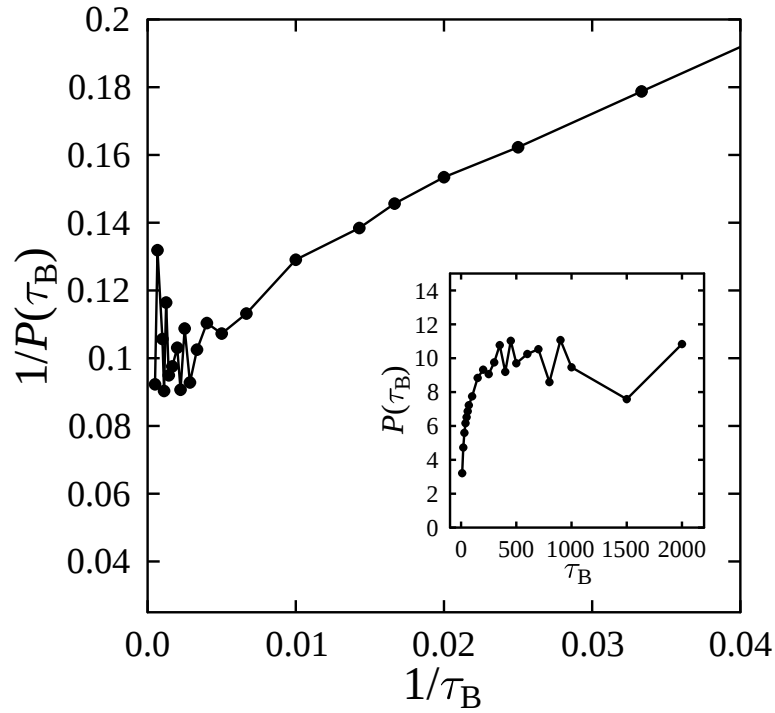
$$P(\tau_B) \equiv \frac{\tau_B}{2} \frac{\sigma_B^2(A)}{\langle A^2 \rangle - \langle A \rangle^2} . \quad (6.41)$$

Rajalla $\tau_B \gg \tau_A$ suure $P(\tau_B)$ lähestyy korrelaatioaikaa τ_A . Eli piirrämme $P(\tau_B)$:n τ_B :n funktiona¹ ja arvioimme raja-arvon $\lim_{\tau_B \rightarrow \infty} P(\tau_B) = \tau_A$.

Blokkimenetelmää voi käyttää myös tutkittaessa, onko simulaation pituus riittävä. Jos huomaamme, että rajalla $\tau_B \rightarrow \tau_{\text{obs}}$ suure $P(\tau_B)$ vielä riippuu voimakkaasti blokin τ_B :stä, on ajomme liian lyhyt.

Kuvassa 6.17 on esitetty 256 neonatomin systeemin potentiaalienergian blokkikeskiarvojen käyttäytyminen blokin koon funktiona.

1. Tai mieluummin $1/P(\tau_B)$:n $1/\tau_B$:n funktiona.



Kuva 6.17: Neonatomisysteemin potentiaalienergian blokkikeskiarvosuureen (6.41) käyttäytyminen blokin koon funktiona.

Suure $P(\tau_B)$ saturoituu suunnilleen arvoon $P(\tau_B \rightarrow \infty) \approx 10$. Korrelaatioaika on siis 10 eli simulaatiodatasta vain joka kymmenennen konfiguraation voidaan katsoa olevan riippumattomia.

6.6 Satunnaisluvuista

Simuloinneissa tarvitaan suuria määriä eri tavoin jakautuneita satunnaislukuja. Eri tavalla jakautuneet luvut saadaan tekemällä esimerkiksi jokin muunnos tasaisesti välille $[0, 1]$ jakautuneille luvuille. Käsitlemme ensi niitä.

6.6.1 Tasaisesti jakautuneiden satunnaislukujen generointi¹

Satunnaisluvut voidaan periaatteessa jakaa kolmeen luokkaan

- i. *Aidot satunnaisluvut* tuotetaan jonkin fysikaalisen prosessin pohjalta (esim. radioaktiivinen hajoaminen tai sähköinen kohina). Tämä on kuitenkin melko epäkäytännöllinen tapa.
- ii. *Pseudosatunnaisluvut* tuotetaan deterministisen algoritmin avulla tietokoneella eli ne eivät ole aidosti satunnaisia, mutta voivat approksimoida aitoja satunnaislukuja hyvin.
- iii. *Valesatunnaisluvut* tuotetaan myös algoritmin avulla. Niillä ei kuitenkaan pyritä aitoon satunnaisuuteen vaan ne on yleensä räätälöity tiettyyn sovellutukseen. Esimerkiksi Monte Carlo -integroinnissa voidaan käyttää valesatunnaislukuja peittämään tietty avaruuden osa yhä tiheämmällä ja tiheämmällä pistejoukolla. Näin voidaan saada tuloksen statistinen virhe pienemmäksi.

Julkaistuja satunnaislukutaulukoita käytettiin aikaisemmin satunnaislukujen lähteenä. Niihin luvut oli saatu esim. fysikaalisista prosesseista, puhelin- ja väestöluetteloista jne.

Käytännössä kaikki simulaatioissa käytettävät satunnaisluvut ovat pseudosatunnaislukuja eli ne tuotetaan satunnaislukugeneraattorilla simulointiohjelmassa.

Hyvältä satunnaislukugeneraattorilta vaaditaan mm. seuraavia ominaisuuksia

- i. Luvuilla tulee olla hyvä jakauma eli niiden tulee olla lähellä satunnaista. Simulaatioissa tärkeä ominaisuus on etteivät satunnaislukujonon luvut saa olla millään tavalla korreloituneita. Saman tulee päteä myös n :n peräkkäisen luvun jonoille. Numeerisessa integroinnissa taas tärkein ominaisuus on jakauman tasaisuus.
- ii. Lukujonolla tulee olla pitkä periodi. Generaattoreiden toteutustavasta johtuen ne alkavat tietyn jakson jälkeen toistamaan itseään.
- iii. Lukujonojen tulisi olla toistettavia. Usein halutaan tutkia tulosten riippuvuutta tie-

1. I. Vattulainen *ym.* (HIP) ovat tutkineet (pseudo)satunnaislukugeneraattoreiden ominaisuuksia. Hyviä referenssejä löytyy heidän WWW-sivulta <http://www.physics.helsinki.fi/~vattulai/publications.html>. Tämän luvun materiaali perustuu pääosin yllämainitun ryhmän julkaisuihin. Lisäksi voisi mainita seuraavat julkaisut: I. Vattulainen, *ym.* *CSC Research Reports*, **R05/92**. T. E. Hull, A. R. Dobell, *SIAM Review* **4** (1962) 230. S. L. Anderson, *SIAM Review* **32** (1990) 221. P. L'Ecuyer, *Ann. Oper. Res.* **53** (1994) 77. A. Compagner, *Am. J. Phys.* **59** (1991) 700. F. James, *Comput. Phys. Commun.* **60** (1990) 329.

tyistä parametreista, jolloin halutaan ajaa useita ajoja täsmälleen samalla lukujonolla. Joissain tapauksissa halutaan aloittaa simulaatioajo tietystä kohdasta, jolloin generaattorin tila on tallennettava. Tila voi koostua yhdestä tai useammasta kokonaisluvusta.

- iv. Generaattorin tulisi olla helposti siirrettävissä koneympäristöstä toiseen. Algoritmi tulee siten laatia jollain korkean tason kielellä.
- v. Generaattorin tulee olla luonnollisesti riittävän nopea. Tämän ominaisuuden merkitys on ehkä tietokoneiden ja simulointialgoritmien kehittyessä hieman vähentynyt. (Yhtä satunnaislukua kohti tehdään yhä enemmän asioita.)
- vi. Algoritmin tulisi olla rinnakkaistettavissa.

Satunnaislukugeneraattorit toteutetaan useimmiten kokonaislukuaritmetiikalla, ja haluttu välille $[0, 1)$ sjoittuva reaaliluku saadaan skaalaamalla. Seuraavassa esitellään lyhyesti yleisimmät algoritmit.

Lineaarinen kongruentiaaligeneraattori on vanhin tyyppi. Se on muotoa

$$x_{i+1} = (ax_i + c) \bmod m, \quad (6.42)$$

missä luvut x_i muodostavat halutun satunnaislukujonon. Oleelliset parametrit ovat a , c ja m ja siitä käytetään merkintää $LCG(a, c, m)$.

Parametri m on yleensä tietokoneen suurin mahdollinen kokonaisluku tai sitä hiukan pienempi ja se määrää generaattorin jakson pituuden P . Sille pätee $P \leq m$. Esimerkiksi 32-bittisellä koneella yleinen valinta on $2^{31} - 1$ tai 2^{32} . Tulos saadaan välille $[0, 1)$ skaalaamalla: x_i/m .

Vakio c voidaan myös asettaa nolllaksi, jolloin käytetään merkintää $MLCG(a, m)$.

Lukujonon aloittamiseen tarvitaan siemenluku x_0 . Se voidaan antaa ohjelmalle syöttötietona tai sen voi laskea esimerkiksi kellonajan perusteella. Alla C-kielinen esimerkki.

```
#include <sys/time.h>
int getseed()
{
    int i;
    struct timeval tp;
    if (gettimeofday(&tp, (struct timezone *)NULL)==0) {
        i=tp.tv_sec+tp.tv_usec;
        i=(i%10000000)|1;
        return i;
    } else {
        return -1;
    }
}
```

Ja itse generaattori [esimerkkinä $LCG(69069, 1, 2^{32})$]:

```

float lcg(long int *seed)
{
    static long int a=69069,c=1,m=4294967296; /* 2^32 */
    static float rm=4294967296.0;

    *seed=(*seed*a+c)%m;
    return (float)*seed/rm;
}

```

Ja lyhyt pääohjelma, joka käyttää edellämainittuja rutiineja:

```

#include <stdio.h>
#include <stdlib.h>

long int getseed();
float lcg(long int *iseed);

main (int argc, char **argv)
{
    long int seed;
    int i;
    float r;

    seed=getseed();
    fprintf(stderr,"Seed  %ld\n",seed);

    for (i=1;i<10001;i++) fprintf(stdout,
                                   "%g %g\n",lcg(&seed),lcg(&seed));

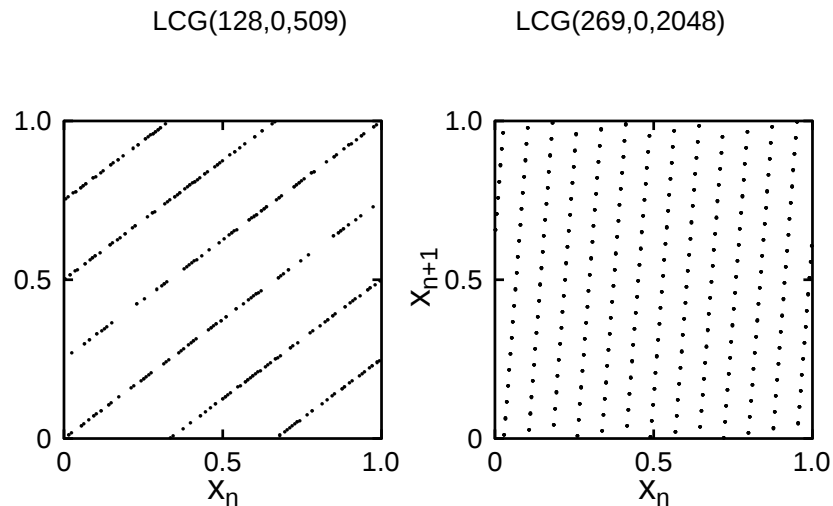
    return(0);
}

```

LCG-generaattoreiden ominaisuudet tunnetaan kohtalaisen hyvin.

Niiden ehkä suurin heikkous on se, että satunnaislukujen d -kertojen $\{x_{i+1}, x_{i+2}, \dots, x_{i+d}\}$ muodostamat pisteet asettuvat d -ulotteisessa avaruudessa lukuisiin samansuuntaisiin hypertasoihin, joiden keskimääräinen etäisyys toisistaan on vakio. Tasojen lukumäärän teoreettinen maksimi d -ulotteisessa hyperkuutiossa $[0, 1)^d$ on $[d!m]^{1/d}$. Jos lukumäärä on tätä oleellisesti pienempi, on pisteiden jakauman tasaisuus heikko. Tavoitteena on etsiä parametrit (a, m) , joiden avulla maksimoidaan hypertasojen lukumäärä.

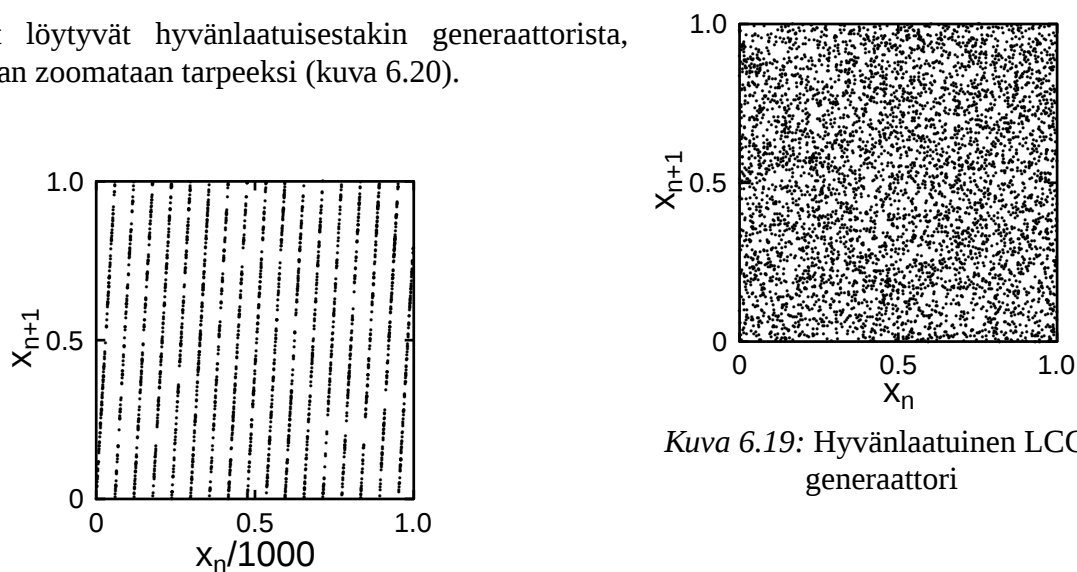
Alla esimerkkinä tapaus $d = 2$ kahdella eri parametrijoukolla.



Kuva 6.18: Huonolaatuinen LCG-generaattori.

Esimerkki käytännön satunnaislukugeneraattorista on kuvassa 6.19 oleva $\text{LCG}(69069, 1, 2^{32})$.

Tasot löytyvät hyvänlaatuisestakin generaattorista, kunhan zoomataan tarpeeksi (kuva 6.20).



Kuva 6.19: Hyvänlaatuinen LCG-generaattori

Kuva 6.20: Hyvänlaatuisen LCG-generaattorin korrelaatiot. Generaattori on sama, jota käytettiin laskuharjoitusten Ising-malliohjelmassa.

Generaattorin lukujonon laatu riippuu siis parametreista. Niiden valinta on melko hankalaa, joten parasta lienee tukeutua generaattoreihin, joita löytyy kirjallisuudesta.

Viiveistettyjen Fibonacci-generaattoreiden (LF) toiminta perustuu Fibonacci-lukujonon $x_i = x_{i-1} + x_{i-2}$ yleistykseen

$$x_i = (x_{i-q} \otimes x_{i-r}) \bmod m, \quad (6.43)$$

missä parametrit q ja r ovat viiveitä ($q > r$) ja $\otimes$ on jokin binäärioperaatio seuraavista vaihtoehtoista

- $+$ yhteenlasku
- $-$ erotus
- $\times$ tulo
- $\oplus$ XOR (exclusive OR), jos m on kahden potenssi.

Generaattorista käytetään merkintää $LF(q, r, \otimes)$. Sen alustaminen vaatii q kappaletta siemenlukuja, jotka voidaan esimerkiksi tuottaa jollain toisella generaattorilla.

LF-generaattoreiden käyttö on melko uutta, joten niiden teoreettisista ominaisuuksista ei kovin paljon tiedetä. Niiden jakso pystytään laskemaan ja niiden etuna onkin pitkä jakso. Tietyillä ehdoilla¹ jakson pituudeksi tulee peräti $(2^p - 1)(2^m - 1)$.

Kirjallisuudessa esiintyy suosituksia, että operaatiota $\oplus$ ei kannata käyttää, koska siinä ei tapahdu tarpeeksi lukujen bittien sekoittumista.

Esimerkki LF-generaattorista on kirjan *Numerical Recipes* generaattori **RAN3**, joka on muotoa $LF(55, 24, -)$.

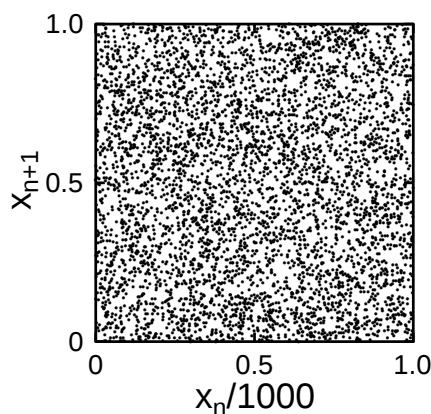
Samantyyppisiä korrelaatioita kuin LCG-generaattoreilla ei esimerkiksi **RAN3**:lla esiinny (kuva 6.21).

Kolmas yleinen tapa on **siirtorekisterit**. Ne voi hahmottaa LF-generaattorin erikoistapauksena $m = 2$. Ne perustuvat bittijonoon $b = \{b_i\}$, jonka termin määritellään rekursiivisesti:

$$b_i = (c_1 b_{i-1} + c_2 b_{i-2} + \dots + c_w b_{i-w}) \bmod 2. \quad (6.44)$$

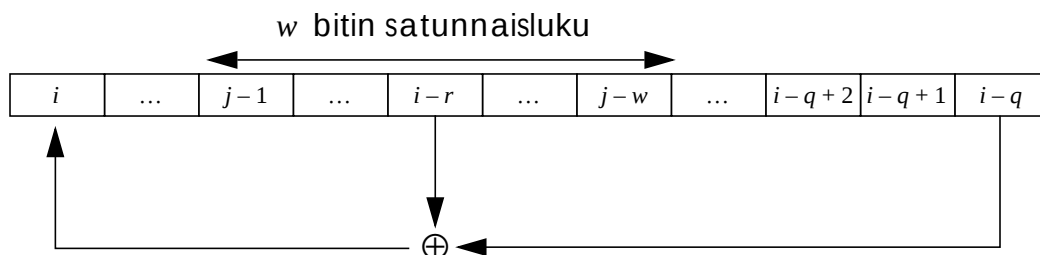
Kertoimet c_j ($j = 1, \dots, w-1$) ovat joko nollia tai ykkösiä ja $c_w = 1$, jolloin binäärijonon w -kertojen $\{b_{i-1}, b_{i-2}, \dots, b_{i-w}\}$ periodi on maksimissaan $2^w - 1$. Satunnaiset kokonaisluvut saadaan näistä w -kerroista.

Käytännössä jonon $\{b_i\}$ kertoimet c_j valitaan yhtälön (6.43) mukaisesti. Kertoimet c_q ja c_r ovat siten ykkösiä ja muut nollia. Valitsemalla binäärioperaatioksi $\oplus$ ja asettamalla $m = 2$, jonon muodostuminen on kuvan (6.22) mukainen.



Kuva 6.21: Generaattori RAN3.

1. D. E. Knuth, *Semi-Numerical Algorithms, vol. 2: The Art of Computer Programming*, 3rded. (Addison-Wesley, Redding, MA, 1998).

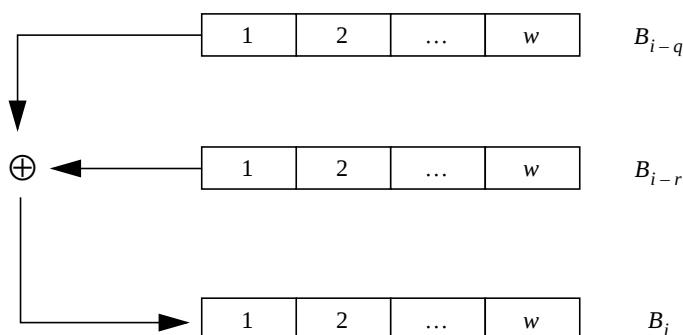


Kuva 6.22: Siirtorekisterigeneraattori.

Tämä generaattorin ongelma on periodin lyhyys, joten siitä on kehitetty yleistetty versio: yleistetty takaisinkytketty siirtorekisteri (GFSR). Idean on käsitellä kokonaislukuja vastaavia bittijonoja $B_i = \{b_{j+1}, b_{j+2}, \dots, b_{j+w}\}$ ja suorittaa binäärioperaatio $\oplus$ näiden jonojen välillä yhtälöä (6.44) vastaavalla tavalla:

$$B_i = (c_1 B_{i-1} + c_2 B_{i-2} + \dots + c_q B_{i-q}) \bmod 2. \quad (6.45)$$

Kertoimet valitaan yleensä yhtälön (6.43) mukaan. Eli kuvana



Kuva 6.23: Yleistetty takaisinkytketty siirtorekisteri.

GFSR-generaattorin etuna on periodin pituus, jonka määrää takaisinkytkennän pituus q . Generaattoreista on kuitenkin löytynyt korrelaatioita, jotka hyvin vaativissa simulaatioissa voivat vaikuttaa tuloksiin¹. Esimerkkinä GFSR-generaattorista olkoon **R250**:

/*

r250 random number generator

*A past shift-register sequence random number generator with maximum cycle length $2^{**}250$. The algorithm is based on the formula represented by S. Kirkpatrick and E. Stoll (see Journal of Computational Physics 40, 527 (1981), pages 517-526).*

parameters:

x integer input-output array of dimension $n + 250$. before

1. I. Vattulainen ym., *Phys. Rev. Lett.* **73** (1994) 2513.

first call to the subroutine the elements 1-250 of the array must be filled with seeds.

r - floating point output array of dimension n . r contains uniform distributed random numbers between $[0,1[$.
 n - integer input scalar, the dimension of r .

Example call (function `ran` returns a random number in $[0,1]$):

```
#define N250 10251
#define N 10000
#define N1 10001
...
int i;
int x[N250];
float r[N1];
...
for (i=1;i<=250;i++) x[i]=2147483647.0*ran(&seed);
r250(x,r,N);
...

*/

void r250(int *x,float *r,int n)
{
    static int q=103,p=250;
    static double rmaxin=2147483648.0; /* rmaxin = 2**31 to avoid */
    /* the return of one (32-bit machine) */
    int i,k;

    for (k=1;k<=n;k++) {
        x[k+p]=x[k+p-q]^x[k];
        r[k]=(float)x[k+p]/rmaxin;
    }
    for (i=1;i<=p;i++) x[i]=x[n+i];
}
```

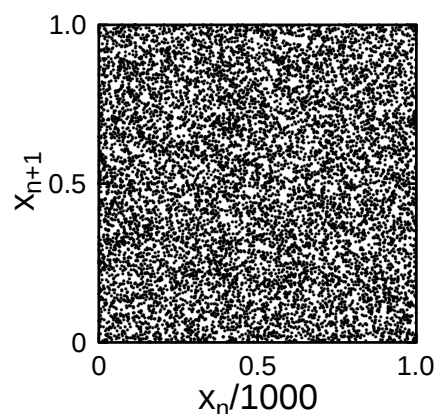
R250:ssä ei ole yhtä pahoja 2-kertakorrelaatioita kuin LCG-generaattoreissa (ks. kuva (6.24)).

Eri generaattoreita ja menetelmiä voidaan tietysti myös yhdistellä satunnaisuuden parantamiseksi.

Yksinkertaisessa **sekoitusmenetelmässä** käytetään kahta satunnaislukugeneraattoria, jotka tuottavat tasaisesti välille $[0, 1)$ jakautuneet lukujonot $\{x'_i\}$ ja $\{y'_i\}$. Menetelmässä poimitaan lukujonon $\{y'_i\}$ avulla näytteitä jonosta $\{x'_i\}$.

Kokeellisesti on havaittu, että yhdistämällä kahden tai useamman satunnaislukugeneraattorin tulokset voidaan saada lukujono, jonka satunnaisuus voi olla sen osia parempi.

Olkoot kombinaation osina lukujonot $\{x_i\}$ ja $\{y_i\}$,. Näiden maksimiavot ovat X ja Y ja



Kuva 6.24: Generaattorin **R250** korrelaatiot.

$m = \max(X, Y)$. Näiden kombinaatio on

$$z_i = (x_i \otimes y_i) \bmod m, \quad (6.46)$$

jossa $\otimes$ on jokin operaatioista $(+, -, \times, \oplus)$. Luvut z_i normalisoidaan välille $[0, 1)$: z_i/m .

Jos lukujonot $\{x_i\}$ ja $\{y_i\}$ on jo normitettu välille $[0, 1)$, voidaan kombinaatio muodostaa esimerkiksi seuraavasti

$$z_i = \begin{cases} x_i + y_i - |x_i + y_i|, & \text{jos operaatio on } + \\ |x_i - y_i|, & \text{jos operaatio on } - \\ x_i \times y_i, & \text{jos operaatio on } \times \end{cases}. \quad (6.47)$$

Kombinaatiogeneraattorien maksimiperiodi on varsin pitkä.

Esimerkkinä kombinaatiogeneraattorista on **RANMAR**. Ensimmäinen komponentti on LF(97, 33, $\bullet$), missä operaatio $\bullet$ kahden reaalisen satunnaisluvun välillä on

$$x_i \bullet x_j = \begin{cases} x_i - x_j, & \text{jos } x_i \geq x_j \\ x_i - x_j + 1, & \text{jos } x_i < x_j \end{cases} \quad (6.48)$$

Toisena generaattorina on yksinkertainen aritmeettinen jono moduluksen ollessa alkuluku $m = 2^{24} - 3 = 16777213$. 24-bittiselle realliluvuille tämä voidaan toteuttaa binäärioperaatiolla $y_i \cdot g$:

$$y_i \cdot g = \begin{cases} y_i - g, & \text{jos } y_i \geq g \\ y_i - g + 16777213/16777216, & \text{jos } y_i < g \end{cases}, \quad (6.49)$$

$$\text{missä } g = 7654321/16777216. \quad (6.50)$$

Kombinaation tulos saadaan näiden kahden operaation avulla:

$$\begin{aligned} x_i &= x_{i-97} \bullet x_{i-33} \\ y_i &= y_{i-1} \cdot g \\ z_i &= x_i \bullet y_i \end{aligned} \quad (6.51)$$

RANMARin jakso on pitkä $P \approx 2^{144} = 2 \times 10^{43}$, ja sillä voidaan toteuttaa lukuisa määrä erillisiä osajonoja. Viimeksi mainittu piirre on tärkeä simulointiohjelmia rinnakkaistettaessa. Lisäksi generaattori on täysin siirrettävissä koneesta toiseen. Toisaalta generaattorin tilan tallettamiseen tarvitaan 103 luvun talletus.

Generaattoreiden testaukseen ei ole olemassa kovin hyviä teoreettisia keinoja, joten testaus on tehtävä ‘empiirisesti’ käyttäen erilaisia mm. statistisia menetelmiä.

i. Tilastollisia testejä ovat mm.

- a) χ^2 -testi, Kolmogorov-Smirnov-testi
 - b) run-testi, jossa tutkitaan monotonisesti kasvavien lukujonojen pituuksia;
 - c) spektraalitestit, jossa tutkitaan d kertojen sjoittuminen d -ulotteiseen avaruuteen
- jne.

ii. Lisäksi voidaan tutkia lukujonojen satunnaisuutta bittitasolla, jos generaattori on toteutettu kokonaislukuaritmetiikalla.

iii. Lopulta voidaan tehdä visuaalisia testejä. Eräs yksinkertaisimmista ja ilmeisimmistä on jo aikaisemmin mainittu satunnaislukuparien sijoittuminen yksikköneliöön. Näin voidaan löytää sekä globaaleja että lokaaleja epäsatunnaisuuksia. Myös bittitasolla voidaan tehdä visuaalisia testejä.

iv. Viime aikoina on testaukseen käytetty myös tiettyjä fysikaalisia suureita¹. Esimerkkinä autokorrelaatioajat Ising-mallin MC-simulaatiossa. Näin on löydetty mm. lyhyen kantaman korrelaatioita siirtorekisterigeneraattoreista.

Edellämainitussa julkaisussa² testattiin kahdeksaa satunnaislukugeneraattoria tilastollisilla, bittitason ja visuaalisilla testeillä.

Yhteenvetona tässä tutkimuksessa todetaan, että generaattoriluokan valinta ei ole olennaisen tärkeää etsittäessä hyvää generaattoria.

Parhaiten testeissä menestyivät **R250** ja **RANMAR**. **R250** on kuitenkin alustettava kunnollisesti (ei bittitason korrelaatioita).

Lisäksi on huomattava, että tämän tutkimuksen jälkeen on GFSR-generaattoreista löydetty korrelaatioita, jotka voivat joissain tapauksissa vaikuttaa simulaatiotuloksiin.

Mikään yksittäinen testi ei voi kertoa, mikä generaattori on paras. Eri sovellutuksissa on erilaiset vaatimukset. Viime aikoina on kuitenkin pyritty kehittämään testejä, jotka testaavat juuri tietyissä sovellutuksissa vaadittavia ominaisuuksia. Jos epäilet satunnaislukugeneraattorisi laatua, on yksi hyvä tapa kokeilla jotain toista (eri generaattoriluokkaan) kuuluvaa generaattoria, ja katsoa muuttuvatko tulokset.

1. I. Vattulainen ym., *PRL* **73** (1994) 2513.

2. I. Vattulainen ym., *CSC Research Reports* **R05/92**

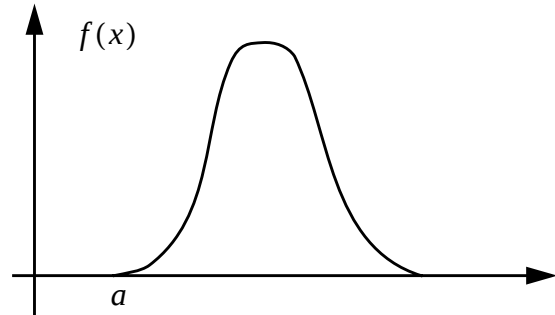
6.6.2 Erilaiset sattunnaislukujakaumat

Miten saamme aikaan halutulla tavalla jakautuneita satunnaislukuja olettaen, että käytössä on satunnaislukugeneraattori, joka antaa tasaisesti välillä $0 \rightarrow 1$ jakautuneita lukuja?

Olkoon haluttu todennäköisyysjakauma $f(x)$ välillä $[a, b]$. Sille tietysti pätee

$$f(x) \geq 0, \forall x \in [a, b] \quad (6.52)$$

$$\int_a^b f(x) dx = 1. \quad (6.53)$$

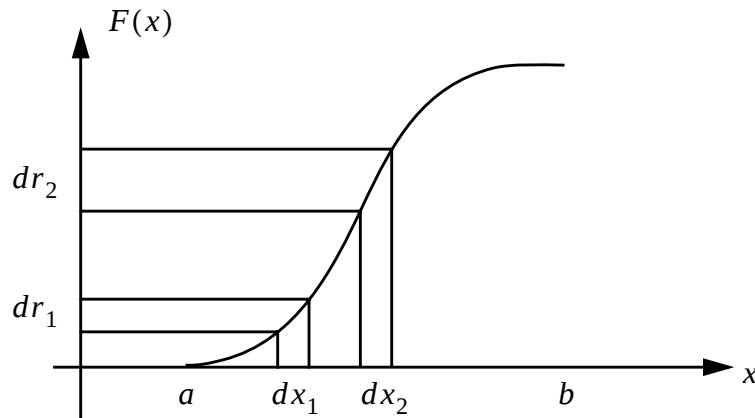


Olkoon $F(x)$ kumulatiivinen jakauma

Kuva 6.25: Todennäköisyysjakauma.

$$F(x) = \int_a^x f(x') dx' \equiv r. \quad (6.54)$$

Määritelmän mukaan voimme siis kuvata $F(x)$:n satunnaismuuttujalle $r \in [0, 1]$. Tarkastellaan kuvan mukaisia yhtäsuuria alueita dx_1 ja dx_2 paikoissa x_1 ja x_2 .



Kuva 6.26: Kumulatiivinen todennäköisyysjakauma.

Helposti näemme, että

$$\frac{dr_1}{dr_2} = \frac{(dF(x)/dx)|_{x=x_1}}{(dF(x)/dx)|_{x=x_2}} = \frac{f(x_1)}{f(x_2)}. \quad (6.55)$$

Eli valitsemalla tasaisesti jakautuneita satunnaislukuja, niiden lukujen lukumäärä, jotka osuvat välille dr_1 suhteessa välille dr_2 osuneisiin on näiden pisteiden todennäköisyyksiensä suhde.

Halutulla tavalla jakautuneita satunnaislukuja x saadaan tasaisesti jakautuneista luvuista r kumulatiivisen jakauman käänteisfunktion avulla

$$x = F^{-1}(r) . \quad (6.56)$$

Ja kaikilla laillisilla todennäköisyystiheysfunktioilla on käänteisfunktio.

Ensimmäisenä esimerkkinä olkoon fotonin kulkema matka z väliaineessa (yksiköissä keskimääräinen vapaa matka). Todennäköisyysjakauma on muotoa

$$f(z) = e^{-z} , \quad (6.57)$$

josta saadaan kumulatiivinen jakauma

$$F(z) = \int_0^z e^{-z'} dz' = 1 - e^{-z} . \quad (6.58)$$

Tästä edelleen

$$z = -\ln(1 - r) . \quad (6.59)$$

Koska $1 - r$ ja r ovat samalla tavalla jakautuneet voidaan käyttää muotoa

$$z = -\ln r . \quad (6.60)$$

Toisena esimerkkinä arvotaan R -säteisen pallon sisään tasaisesti jakautuneita pisteitä. Pallokoordinaatistossa ϕ ja $\cos\theta$ ovat tasaisesti jakautuneita. Tiheysfunktion säderiippuvuus $f(r)$ on

$$f(r) = A4\pi r^2 . \quad (6.61)$$

Normituksesta saadaan vakio A :

$$\int_0^R A4\pi r^2 dr = 1 \Rightarrow A = \frac{3}{4\pi} R^{-3} \Rightarrow f(r) = \frac{3}{R^3} r^2 . \quad (6.62)$$

Kumuloitunut jakauma on siten

$$F(r) = \int_0^r f(r') dr' = \frac{3}{R^3} \frac{1}{3} r^3 = \frac{r^3}{R^3} \equiv s = \text{sat.luku} . \quad (6.63)$$

Eli muunnoskaava tasaisesti jakautuneelle satunnaisluvulle s on

$$r = F^{-1}(s) = Rs^{1/3} . \quad (6.64)$$

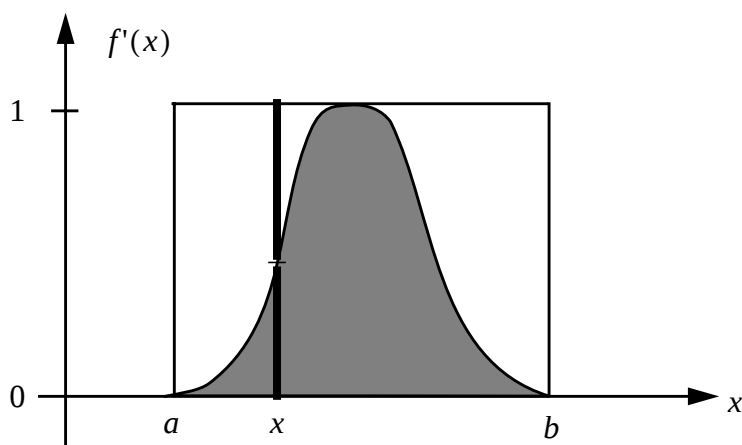
Aina ei ole mahdollista muodostaa funktiota F^{-1} . Tällöin voidaan käyttää hylkäysmenetelmää:

i. Muodostetaan todennäköisyysjakaumasta uusi funktio

$$f'(x) = \frac{f(x)}{f_{\max}} . \quad (6.65)$$

ii. Generoidaan tasaisesti jakautunut satunnaisluku $r_1 \in [0, 1]$. Normitetaan tämä välille $[a, b]$: $x = a + (b - a)r_1$. Jos väli $[a, b]$ ei ole äärellinen on tehtävä muunnos, jolla se palautetaan äärelliseksi. Esim. $x \in [a, \infty]$ voidaan palauttaa muunnoksella $x = a[1 - \ln(1 - y)]$ välille $y \in [0, 1]$.

iii. Generoidaan satunnaisluku $r_2 \in [0, 1]$. Jos $r_2 < f'(x)$, hyväksytään x , muuten palataan askeleeseen ii.



Kuva 6.27: Hylkäysmenetelmä

Tämähän on jo meille tuttu menetelmä. Sitä käytetään Monte Carlo -simulaation uuden tilan hyväksymisessä.

Menetelmässä tietysti tuhlataan satunnaislukuja; etenkin, jos $f(x)$ on kovin piikittynyt. Näiden kahden menetelmän yhdistelmää voidaan myös käyttää. Olkoon $f(x)$ piikittynyt ja vaikeasti käännettävissä ja integroitavissa. Voimme kuitenkin jakaa sen osiin

$$f(x) = g(x)h(x) , \quad (6.66)$$

missä $g(x)$ on helposti käännettävissä ja sisältää enimmäkseen piikittyneisyyden ja $h(x)$ on melko tasainen, mutta matemaattisesti monimutkainen. Satunnaisluvut voidaan generoida seuraavan reseptin mukaan.

- i. Skaalataan $g(x) \rightarrow \tilde{g}(x)$ siten, että $\int_a^b \tilde{g}(x) dx = 1$.
- ii. Skaalataan $h(x) \rightarrow \tilde{h}(x)$ siten, että $\tilde{h}(x) \leq 1 \quad \forall x \in [a, b]$.
- iii. Käyttäen edellämainittua suoraa menetelmää generoidaan satunnaisluku x , joka noudattaa jakaumaa $\tilde{g}(x)$.
- iv. Käyttäen satunnaislukua x sovelletaan hylkäysmenetelmää jakaumafunktiona nyt $\tilde{h}(x)$: Generoidaan satunnaisluku $r \in [0, 1]$; jos $\tilde{h}(x) \leq r$, hyväksy x , muuten mene askeleeseen iii.

Otetaan esimerkkinä vielä ympyräkollimoidun hiukkassuihkun generointi. Tässä tapauksessa normitettu todennäköisyystiheysfunktio on [sylinterikoordinaatistossa (ρ, ϕ)]

$$d^2 f(\rho, \phi) = \frac{1}{\pi \rho_0} \rho d\rho d\phi, \quad 0 \leq \rho \leq \rho_0, 0 \leq \phi \leq 2\pi. \quad (6.67)$$

Tässä ρ_0 on suihkun läpimitta ja ϕ atsimuuttikulma. Tämä jakauma separoituu

$$d^2 f(\rho, \phi) = df_1(\rho) df_2(\phi), \quad (6.68)$$

missä

$$df_1(\rho) = \frac{2}{\rho_0^2} \rho d\rho \quad (6.69)$$

ja

$$df_2(\phi) = \frac{1}{2\pi} d\phi. \quad (6.70)$$

Suoraa menetelmää käyttäen saamme satunnaisluvut seuraavasti. Kumuloituvat jakaumat ovat

$$F_1(\rho) = \frac{2}{\rho_0^2} \int_0^\rho \rho' d\rho' = \frac{\rho^2}{\rho_0^2} \quad (6.71)$$

$$F_2(\phi) = \frac{1}{2\pi} \int_0^\phi d\phi' = \frac{\phi}{2\pi}. \quad (6.72)$$

Kun nämä käännetään, saadaan

$$\rho = \rho_0 \sqrt{\xi_1} , \quad (6.73)$$

$$\phi = 2\pi\xi_2 , \quad (6.74)$$

missä ξ_i ovat tasaisesti välillä $[0, 1]$ jakautuneita satunnaislukuja.

Eli C-koodinpätkä, joka tekee tämän voisi näyttää seuraavalta

```
rho = rho0 * sqrt(rand(&seed));
phi = 2.0*pi * rand(&seed);
x = rho*cos(phi);
y = rho*sin(phi);
```

Funktio `rand(&seed)` antaa tasaisesti välille $[0, 1]$ jakautuneen satunnaisluvun. Hylkäysmenetelmää käyttäen arvomme tasaisesti pisteen neliön $-1 \leq x \leq 1$, $-1 \leq y \leq 1$ sisällä ja hyväksymme sen, jos se osuu yksikkösteisen ympyrän sisälle. Koodina se voisi näyttää seuraavalta:

```
do {
    x = 2.0*rand(&seed)-1.0;
    y = 2.0*rand(&seed)-1.0;
} while (x*x+y*y>1.0);

x = rho0*x;
y = rho0*y;
```

6.7 Atomien välisistä vuorovaikutuspotentiaaleista

6.7.1 Yleistä

‘Teoreettisissa’ simulaatioissa vuorovaikutuspotentiaali valitaan laskennallisesti helpoksi; yleensä paripotentiaaliksi.

Esimerkkinä Lennard-Jones (LJ)

$$V(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]. \quad (6.75)$$

Jos on tarkoitus saada koetuloksiin verrattavia tuloksia on käytettävä realistista potentiaalia. Simulaatioihinkin pätee tokaisu *garbage in–garbage out*.

Yleisesti N :n atomin tai molekyylin systeemin potentiaalienergia voidaan esittää sarjana:

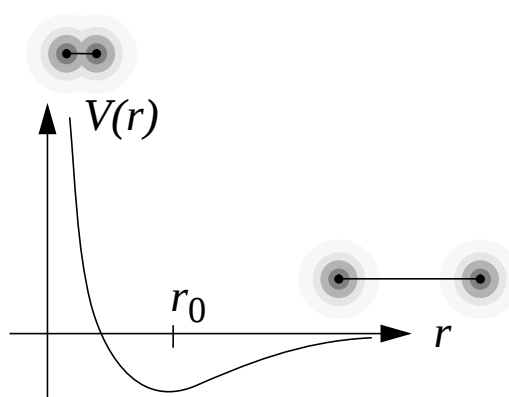
$$\begin{aligned} V = & \sum_{i=1}^N v_1(\mathbf{r}_i) \\ & + \sum_{i=1}^N \sum_{j>i}^N v_2(\mathbf{r}_i, \mathbf{r}_j) \\ & + \sum_{i=1}^N \sum_{j>i}^N \sum_{k>i,j}^N v_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + \dots \end{aligned} \quad (6.76)$$

v_1 on ulkoinen potentiaali, esimerkiksi sähkökenttä.

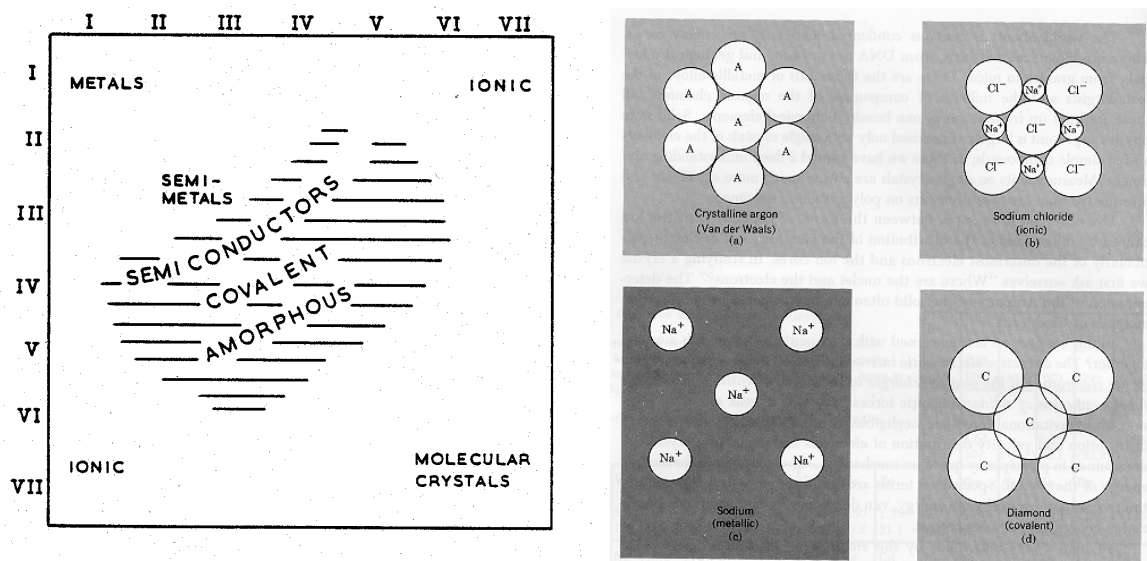
Paripotentiaaliapproksimaatiossa otetaan vain v_1 ja v_2 mukaan. Tällöin v_2 ei välttämättä ole sama kuin dimeerin potentiaali:

Yleisesti kahden atomin välinen potentiaali näyttää kuvan 6.28 mukaiselta. Pienillä etäisyyksillä vallitseeseen elektronikuorien päällekkäinmenosta johtuva repulsio. Suuremmilla etäisyyksillä eri syistä johtuva attraktio: van der Waals, Coulombin vuorovaikutus, metallisidos, kovalenttinen sidos. Potentiaalilla on minimi etäisyydellä r_0 .

Atomien välisten sidosten luonteen määrää se, mistä jaksollisen järjestelmän ryhmästä aineet ovat.



Kuva 6.28: Kahden atomin välinen vuorovaikutuspotentiaali.



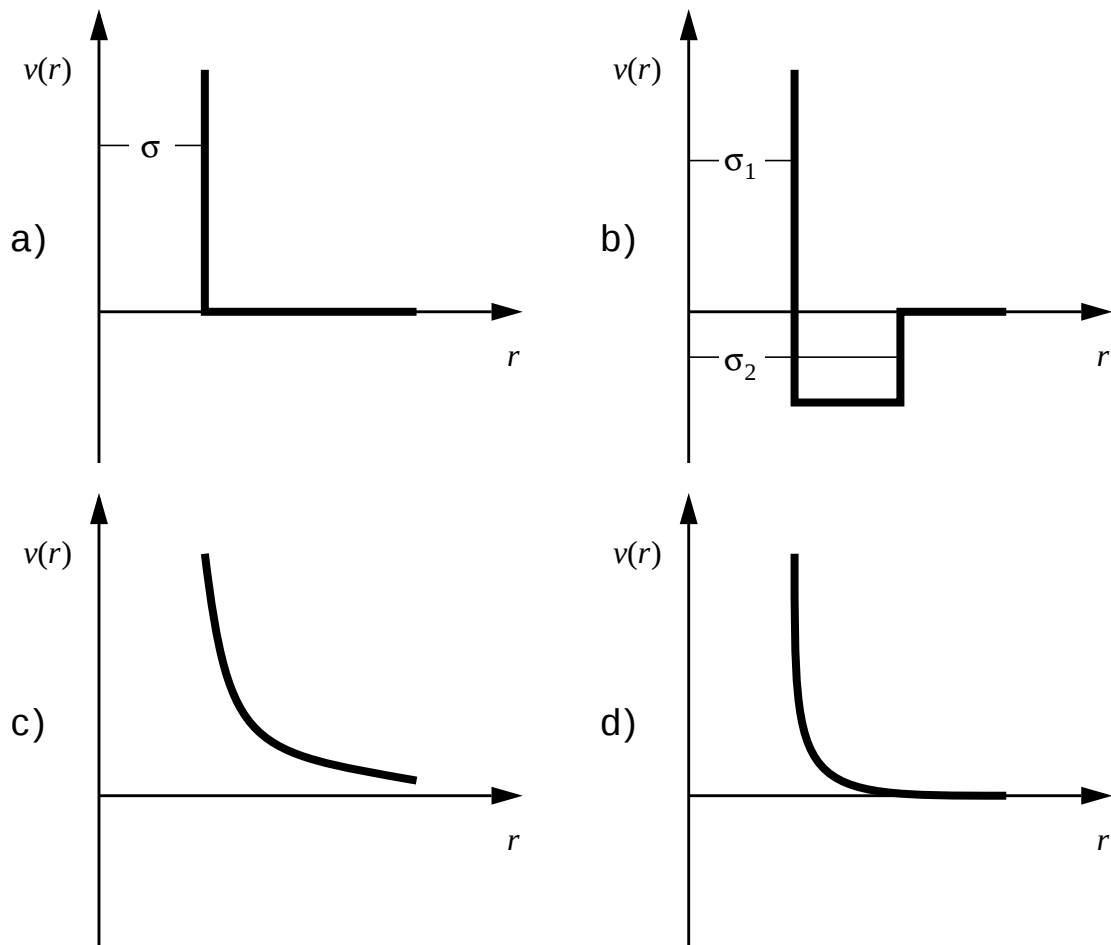
Kuva 6.29: Kemiallisen sidoksen luonne.

Se, että atomisysteemiä voidaan kuvata vuorovaikutuspotentiaalin avulla perustuu Born-Oppenheimer-approksimaatioon. Sen mukaan atomien elektronikuoret reagoivat välittömästi ytimien paikkojen muutokseen.

6.7.2 Idealisoidut potentiaalit

Monesti emme ole kiinnostuneita täsmällisistä luvuista, joita saadaan ulos simulaatioista, vaan ennemminkin systeemin kvalitatiivisesta käyttäytymisestä. Tällöin voimme käyttää idealisoituja potentiaaleja, joiden muodossa on realististen potentiaalien pääpiirteet, mutta jotka ovat laskennallisesti yksinkertaisia.

Kuvassa 6.30 on esitetty muutama yleisesti käytetty potentiaali.



Kuva 6.30: Idealisoituja potentiaaleja: a) kovan pallon potentiaali, b) laatikkopotentiaali, c) pehmeän pallon potentiaali ($v = 1$), d) pehmeän pallon potentiaali ($v = 12$).

Kova pallo -potentiaalissa (*hard sphere*) on vain repulsiivinen osa:

$$V^{\text{HS}}(r) = \begin{cases} \infty, & r < \sigma \\ 0, & r \geq \sigma \end{cases} \quad (6.77)$$

Ensimmäiset MC-simulaatiot¹ tehtiin tällä potentiaalilla. Kuten aikaisemmin mainittiin, kovien pallojen MC-simulaatio redusoituu melko yksinkertaiseen muotoon.

Jos kovaan palloon lisätään vakioattraktio saadaan laatikkopotentiaali (*square well*):

$$V^{\text{SW}}(r) = \begin{cases} \infty, & r < \sigma_1 \\ -\epsilon, & \sigma_1 \leq r < \sigma_2 \\ 0, & r \geq \sigma_2 \end{cases} \quad (6.78)$$

1. Ja myös ensimmäiset MD-simulaatiot.

Jos epäjatkuvuus kovan pallon potentiaalissa tuottaa vaikeuksia voidaan jyrkkyyttä loiventaa: saadaan pehmeä pallo -potentiaali (*soft sphere*):

$$V^{SS}(r) = \varepsilon \left[\frac{\sigma}{r} \right]^v \quad (6.79)$$

Kova pallohan saadaan pehmeästä raja-arvona

$$V^{HS}(r) = \lim_{v \rightarrow \infty} V^{SS}(r) . \quad (6.80)$$

Joskus on käytännöllistä jakaa potentiaali attraktiiviseen ja repulsiiviseen osaan. Esimerkiksi Lennard-Jonesin tapauksessa

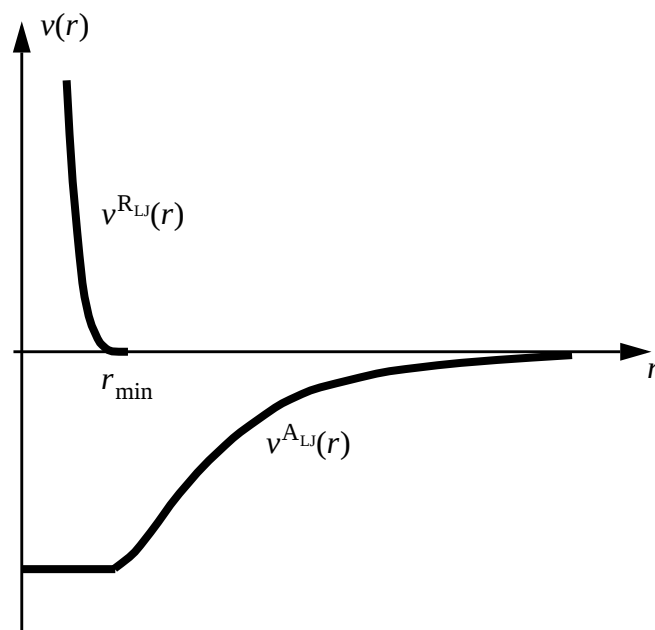
$$V^{LJ}(r) = 4\varepsilon \left[\left(\frac{r}{\sigma} \right)^{12} - \left(\frac{r}{\sigma} \right)^6 \right] = V^{R_{LJ}}(r) + V^{A_{LJ}}(r) , \quad (6.81)$$

missä

$$V^{R_{LJ}}(r) = \begin{cases} V^{LJ}(r) + \varepsilon & r < r_{\min} \\ 0 & r \geq r_{\min} \end{cases} \quad (6.82)$$

ja

$$V^{A_{LJ}}(r) = \begin{cases} -\varepsilon & r < r_{\min} \\ V^{LJ}(r) & r \geq r_{\min} \end{cases} . \quad (6.83)$$



Kuva 6.31: Lennard-Jones-potentiaalin jako repulsiiviseen ja attraktiiviseen osaan.

6.7.3 Redusoidut yksiköt

Jos potentiaalilla on sopivia skaalautumisominaisuuksia, voidaan laskuissa käyttää redusoituja yksiköitä. Tällöin eri systeemien tulokset saadaan skaalaamalla toisistaan. Lisäksi redusoitujen yksikköjen käyttö helpottaa hieman laskentataakkaa: ylimääräisiä vakioita jää pois.

Esimerkiksi otamme tutun Lennard-Jones-potentiaalin:

$$V(r) = 4\varepsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \equiv \varepsilon f(r/\sigma) , \quad (6.84)$$

missä

$$f(x) = 4(x^{-12} - x^{-6}) . \quad (6.85)$$

Tästä voimme helposti päätellä, että

- i. pituuden luonnollinen yksikkö = σ
- ii. energian luonnollinen yksikkö = ε

Eli

$$r^* = r/\sigma , \quad E^* = E/\varepsilon \quad (6.86)$$

Muut yksiköt saamme ylläolevista helposti:

$$\text{aika:} \quad t^* = t/[(m\sigma^2)/\varepsilon]^{1/2} \quad (6.87)$$

$$\text{tiheys:} \quad \rho^* = \rho\sigma^3 \quad (6.88)$$

$$\text{lämpötila:} \quad T^* = k_B T/\varepsilon \quad (6.89)$$

$$\text{paine:} \quad P^* = P\sigma^3/\varepsilon \quad (6.90)$$

$$\text{voima:} \quad \mathbf{f}^* = \mathbf{f}\sigma/\varepsilon \quad (6.91)$$

$$\text{nopeus:} \quad v^* = v/[\varepsilon/m]^{1/2} \quad (6.92)$$

6.7.4 Realistiset potentiaalit

Jos haluamme mallintaa tarkemmin jotain tiettyä materiaalia, eivät edelläesitellyt idealisoidut potentiaalit riitä, vaan on käytettävä realistisempia malleja. Realistiset potentiaalmallit kehitetään yleensä olettamalla jokin tietty funktionaalinen muoto systeemin potentiaalienergialle atomien paikkojen funktiona ja määrittämällä malliparametrit sovittamalla kokeelliseen dataan tai ns. *ab initio* -menetelmillä laskettuihin tuloksiin.

Hyvin yleinen yksinkertaistus on *paripotentiaaliapproksimaatio*. Se kuvaa melko hyvin jalokaasuja, on melko hyvä alkalihalideille ja tietyille ionisille oksideille, kohtalainen metalleille (riippuu metallista ja halutuista ominaisuuksista), mutta huono kovalenttisille yhdisteille.

Ab initio -menetelmistä voisi mainita esimerkiksi Hartree-Fock-menetelmän lähinnä ioniyhdisteiden potentiaalien laskemisessa ja tiheysfunktionaaliteorian (density functional theory, DFT) paikallisen tiheyden approksimaatiolla (local density approximation, LDA) maustettuna¹.

Kokeellisesti potentiaaleja määritettäessä funktionaalinen muoto on usein saatu jostain teoriasta. Potentiaalin parametrit saadaan sovittamalla potentiaalin ennustamat aineen ominaisuudet mittaustuloksiin tai jollain *ab initio* -menetelmällä laskettuihin tuloksiin².

Yleensä käytetään seuraavia suureita

kideparametri a (tai kiderakenne yleensä)

koheesioenergia E_{coh}

elastiset vakiot c_{ij}

kidevaurioiden energiat

pintojen stabiilisuus

hilavärähtelyjen dispersiorelaatiot $\omega(\mathbf{k})$

Jalokaasujen potentiaalienergia koostuu van der Waals -attraktiosta ja repulsiosta. Funktionaalisenä muotona on esimerkiksi Lennard-Jones

$$V(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (6.93)$$

tai Morse

1. R.O. Jones, O. Gunnarsson, *Rev. Mod. Phys.* **61** (1989) 689.

2. C.R.A Catlow, W.C. Mackrodt eds., *Computer Simulation of Solids*, Springer, 1982, luku 1

$$V(r) = De^{-2\alpha(r-r_0)} - 2De^{-\alpha(r-r_0)} . \quad (6.94)$$

Ioniyhdisteillä erityispiirteenä on pitkän kantaman Coulombin vuorovaikutus. Eli potentiaali koostuu periaatteessa kolmesta termistä¹:

i. Coulombin vuorovaikutus:

$$V^{zz}(r_{ij}) = \frac{z_i z_j}{4\pi\epsilon_0 r_{ij}} , \quad (6.95)$$

missä z_i ja z_j ovat ionien i ja j ionivaraukset.

ii. Lyhyen kantaman repulsio, joka syntyy elektronikuorten päällekkäinmenosta (esimerkiksi eksponenttimuoto):

$$V^{SR}(r) = Ae^{-r/\rho} \quad (6.96)$$

iii. van der Waals -attraktio, joka syntyy ionien dipoli-dipolivuorovaikutuksesta (ns. dispersiovuorovaikutus):

$$V^{vdW}(r) = -\frac{C}{r^6} . \quad (6.97)$$

Ääritapaus on alkalihalidit: Monihiukkasvuorovaikutus ei ole kovin suuri, mutta ionien polarisaatio vaikuttaa: optiset fononit. Oksideilla on jo kovalenttisuutta sidoksissa.

Yksinkertaiset metallit (*sp*-metallit, esimerkiksi Na, Mg, Al) ovat kuvattavissa jossain määrin paripotentiaaleilla. Useimmiten käytetään Morsen potentiaalia:

$$V(r) = De^{-2\alpha(r-r_0)} - 2De^{-\alpha(r-r_0)} \quad (6.98)$$

Transitiometalleilla (*d*-kuori vajaa) monihiukkasvuorovaikutus tulee kuvaan mukaan.

Paripotentiaalimallista seuraa ns. Cauchyn relaatio:

$$c_{12} = c_{44} . \quad (6.99)$$

1. C.R.A Catlow, W.C. Mackrodt eds., *Computer Simulation of Solids*, Springer, 1982, luku 10

Tämä pätee kuitenkin vain harvoille metalleille (ks kuva 6.32). Paripotentiaalimallin ennustamat vakanssien muodostumisenergiat ovat liian suuret. Lisäksi paripotentiaalimalli kuvaa huonosti metallipintojen relaksaatiota ja rekonstruktiota.

Viimeisen vuosikymmenen aikana on ns. tiheysfunktioaali- ja tight binding -teorioiden perustalta kehitetty monihiukkaspotentiaaleja, jotka kuvaavat metalleja varsin realistisesti.

Esimerkkinä olkoon Finnisin ja Sinclairin semiempiirinen N-hiukkaspotentiaali¹, joka saa perustelunsa tight binding -approksimaatiosta. Sen mukaan koheesioenergia atomia kohti on verrannollinen koordinaatioluvun² neliöjuureen³:

$$V_{\text{coh}} \propto \sqrt{z}. \quad (6.100)$$

Atomin i energia lausutaan muodossa

$$V_i^{\text{FS}} = \frac{1}{2} \sum_{j \neq i} \Phi(r_{ij}) - \left[\sum_{j \neq i} -\eta(r_{ij}) \right]^{1/2}, \quad (6.101)$$

missä Φ on paripotentiaali ja η on ns. koheesiopotentiaali, joka oli alunperin parabolista muotoa.

Kovalenttisissa aineissa paripotentiaaliapproksimaatio ei päde ollenkaan. Otetaan esimerkiksi pii.

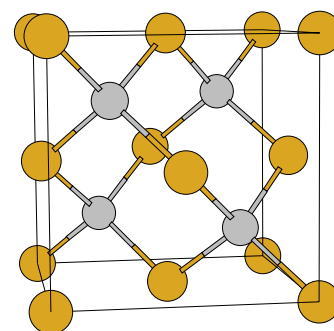
Timanttikide on avoin kiderakenne. Se on kaukana tiivispakkauksesta: jokaisella atomilla on vain 4 lähinaapuria. (Vertaa fcc- ja hcp-rakenteisiin: 12 lähinaapuria.)

Piitä voidaan yrittää mallintaa paripotentiaalilla, mutta siinä on otettava huomioon myös toiseksi lähimmät naapurit⁴. Tämä malli ei ole kovin käyttökelpoinen.

SUBSTANCE	C_{11}	C_{12}	C_{44}
Li (78 K)	0.148	0.125	0.108
Na	0.070	0.061	0.045
Cu	1.68	1.21	0.75
Ag	1.24	0.93	0.46
Au	1.86	1.57	0.42
Al	1.07	0.61	0.28
Pb	0.46	0.39	0.144
Ge	1.29	0.48	0.67
Si	1.66	0.64	0.80
V	2.29	1.19	0.43
Ta	2.67	1.61	0.82
Nb	2.47	1.35	0.287
Fe	2.34	1.36	1.18
Ni	2.45	1.40	1.25
LiCl	0.494	0.228	0.246
NaCl	0.487	0.124	0.126
KF	0.656	0.146	0.125
RbCl	0.361	0.062	0.047

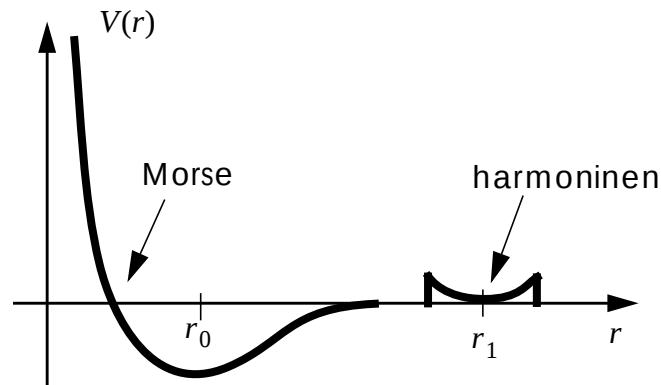
^a Elastic constants in 10^{12} dynes-cm⁻² at 300 K.

Kuva 6.32: Eräiden kuutiollisten kiteiden elastisia vakioita. [N.W.Ashcroft, N.D.Mermin: Solid State Physics]



Kuva 6.33: Timanttirakenne

1. M.W. Finnis, J.E. Sinclair, *Phil. Mag. A* **50** (1984) 45.
2. Atomin koordinaatioluku on sen lähinaapureiden lukumäärä.
3. Huomaa, että paripotentiaalimalli ennustaa lineaarisen riippuvuuden.
4. Yu. Knyzhnikov, *Rad. Eff.* **26** (1975), A. Mazzone *Phil. Mag. Lett.* **60** (1989) 131., *Phys. Stat. Sol.* **165** (1991) 395.



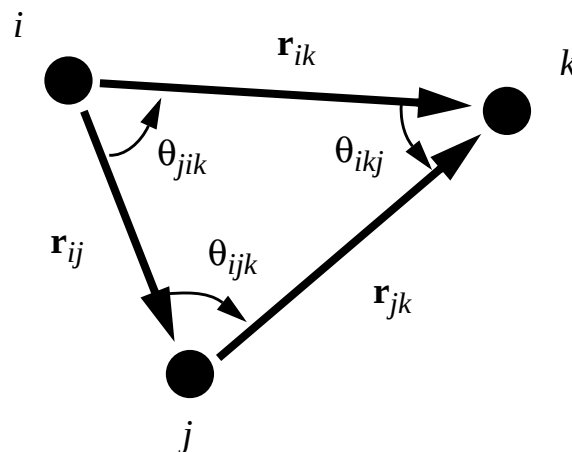
Kuva 6.34: Paripotentiaali piille.

Kovalenttisia aineita mallinnetaan nykyään yleensä kolmi- tai monihiukkaspotentiaalilla.

Kolmihiukkaspotentiaalin tapauksessa energia riippuu atomien etäisyyksien lisäksi kahden sidoksen välisestä kulmasta:

$$V(r) = \varepsilon f_2(r_{ij}) + \varepsilon f_3(r_i, r_j, r_k) \quad (6.102)$$

$$f_3(r_i, r_j, r_k) = h(r_{ij}, r_{ik}, \theta_{jik}) + h(r_{ji}, r_{jk}, \theta_{ijk}) + h(r_{ki}, r_{kj}, \theta_{ikj}) \quad (6.103)$$



Kuva 6.35: Kolmihiukkaspotentiaali.

Monihiukkaspotentiaalimallissa atomin vuorovaikutus naapureidensa kanssa riippuu sen ympäristöstä eli ns. koordinaatioluvusta. Koordinaatioluku on karkeasti ottaen atomin lähinaapureiden lukumäärä.

Esimerkkinä monihiukkaspotentiaalista olkoon Tersoffin¹ kehittämä malli, jota on menestyksekkäästi käytetty esimerkiksi useiden timanttirakenteisten aineiden mallintamiseen. Potenti-

aali on seuraavanlainen. Systeemin kokonaisenergia lausutaan tutussa muodossa

$$E = \frac{1}{2} \sum_{i \neq j} V_{ij} \quad , \quad (6.104)$$

missä

$$V_{ij} = f_C(r_{ij})[a_{ij}f_R(r_{ij}) + b_{ij}f_A(r_{ij})] \quad , \quad (6.105)$$

$$\text{repulsiivinen osa } f_R(r) = Ae^{-\lambda_1 r} \quad , \quad (6.106)$$

$$\text{attraktiivinen osa } f_A(r) = -Be^{-\lambda_2 r} \quad , \quad (6.107)$$

potentiaalin katkaisufunktio

$$f_C(r) = \begin{cases} 1, & r \leq R - D \\ \frac{1}{2} - \frac{1}{2} \sin\left(\frac{\pi(r-R)}{2D}\right), & R - D < r < R + D \\ 0, & r \geq R + D \end{cases} \quad , \quad (6.108)$$

sekä

$$b_{ij} = (1 + \beta^n \zeta_{ij}^n)^{-1/2n} \quad , \quad (6.109)$$

$$\zeta_{ij} = \sum_{k \neq i, j} f_C(r_{ik})g(\theta_{jik})\exp[\lambda_3^3(r_{ij} - r_{ik})^3] \quad , \quad (6.110)$$

$$g(\theta) = 1 + \frac{c^2}{d^2} - \frac{c^2}{d^2 + (h - \cos\theta)^2} \quad , \quad (6.111)$$

$$a_{ij} = (1 + \alpha^n \eta_{ij}^n)^{-1/2n} \quad \text{ja} \quad (6.112)$$

$$\eta_{ij} = \sum_{k \neq i, j} f_C(r_{ik})\exp[\lambda_3^3(r_{ij} - r_{ik})^3] \quad . \quad (6.113)$$

Tässä, kuten edelläkin, atomien i ja j välinen etäisyys on r_{ij} ja sidosten ij ja ik välinen kulma on θ_{jik} .

7 Tasapainosimulaatioiden sovellutuksia

7.1 Tilanyhtälö ja faasitasapaino

Termodynaamisen systeemin tilanyhtälöhän kertoo systeemin tilamuuttujien keskinäisen riippuvuussuhteen tasapainossa. Yleensä tilanyhtälössä on mukana muuttujia, joita on helppo mitata eli paine, lämpötila, tilavuus, tiheys tai — kuten magneettisten systeemien tapauksessa — kenttäsuureita.

Kaasujen, nesteiden ja kiinteiden aineiden tapauksessa tilanyhtälö on useimmiten muotoa

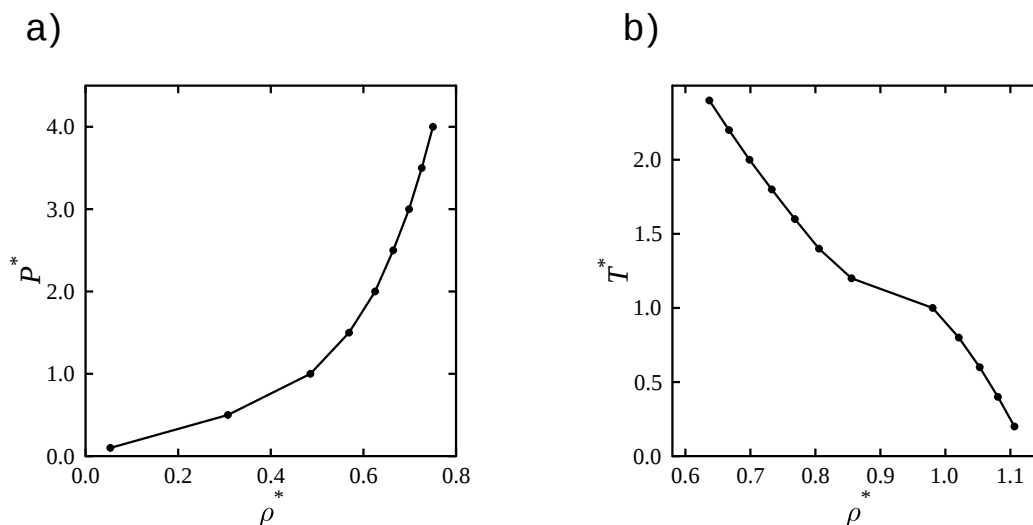
$$P = P(V, T) \quad (7.1)$$

tai

$$P = P(\rho, T), \quad (7.2)$$

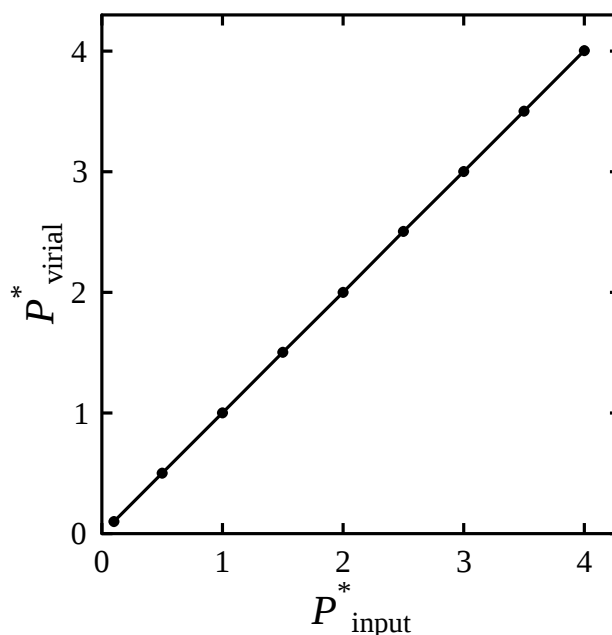
missä ρ on aineen (atomi)tiheys.

Esimerkkinä kuvassa 7.1 on 500 atomin Lennard-Jones-systeemin tilanyhtälö vakio paine-MC:llä laskettuna..



Kuva 7.1: Lennard-Jones-systeemin tilanyhtälö a) $P^* = P^*(\rho^*, T^*)$, kun $T^* = 2.0$. ja b) $T^* = T^*(\rho^*, P^*)$, kun $P^* = 3.0$.

Huomaa, että vakiopainesimulaatiossa paine annetaan syöttötietona. Sen lisäksi se voidaan laskea simulaation kuluessa viriaalista (ks. kappale 2.3). On hyvä tarkistaa, että nämä kaksi ovat sopusoinnussa keskenään.



Kuva 7.2: Viriaalista laskettu paine simulaation syöttötietona annetun paineen funktiona vakio-painesimulaatiossa. Systemi koostui 500 Lennard-Jones-atomista.

Tasapainotilassa voi systeemistä löytyä eri olomuodoissa tai faaseissa olevia osia, jos paineella ja lämpötilalla on sopivat arvot. Eri faasit eroavat järjestyksen ja — tässä tapauksessa — tilavuuden suhteen toisistaan.

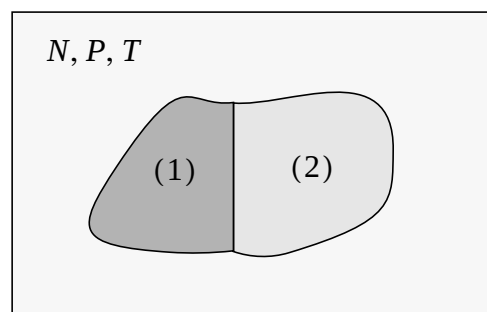
Tarkastellaan seuraavassa yhdelaisista atomeista koostuvan NPT-systeemin faasitasapainoa.

Jos atomien siirtyminen eri faasien välillä on mahdollista, tulee seuraavien ehtojen päteä eri faaseille

$$T^{(1)} = T^{(2)} \quad (7.3)$$

$$P^{(1)} = P^{(2)} \quad (7.4)$$

$$\mu^{(1)} = \mu^{(2)} . \quad (7.5)$$



Kuva 7.3: Faasitasapaino

Jos kirjoitamme kemiallisen potentiaalin lämpötilan ja paineen funktiona, saamme faasitasapainoehdoksi paineessa P ja lämpötilassa T

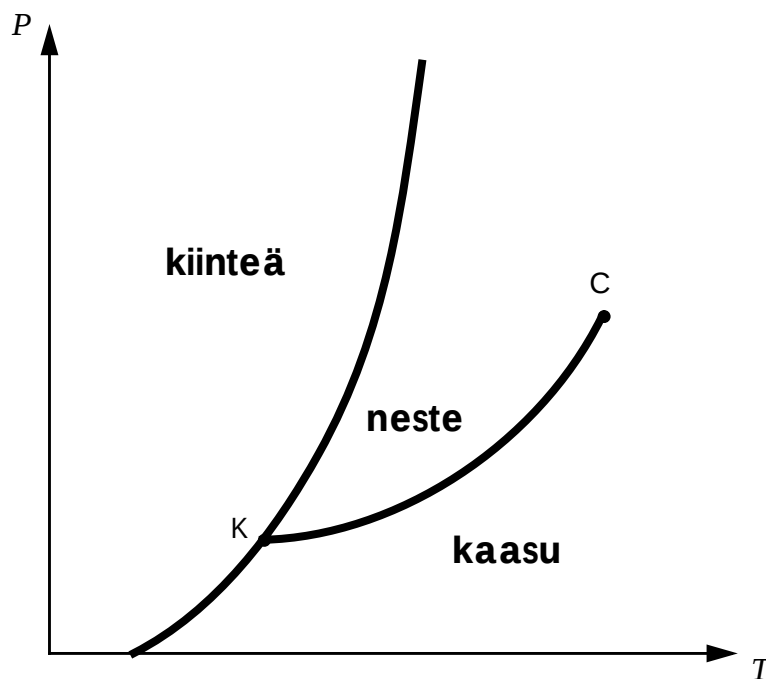
$$\mu^{(1)}(P, T) = \mu^{(2)}(P, T) . \quad (7.6)$$

Tästä näemme, että faasitasapainossa (eli koeksistenssissä) paine on lämpötilan funktio $P_{\text{coex}} = P_{\text{coex}}(T_{\text{coex}})$. Tämä määrää ns. koeksistenssikäyrän PT -tasossa.

Jos faaseja on kolme, on yhtälöä (7.6) vastaava ehto

$$\mu^{(1)}(P, T) = \mu^{(2)}(P, T) = \mu^{(3)}(P, T) . \quad (7.7)$$

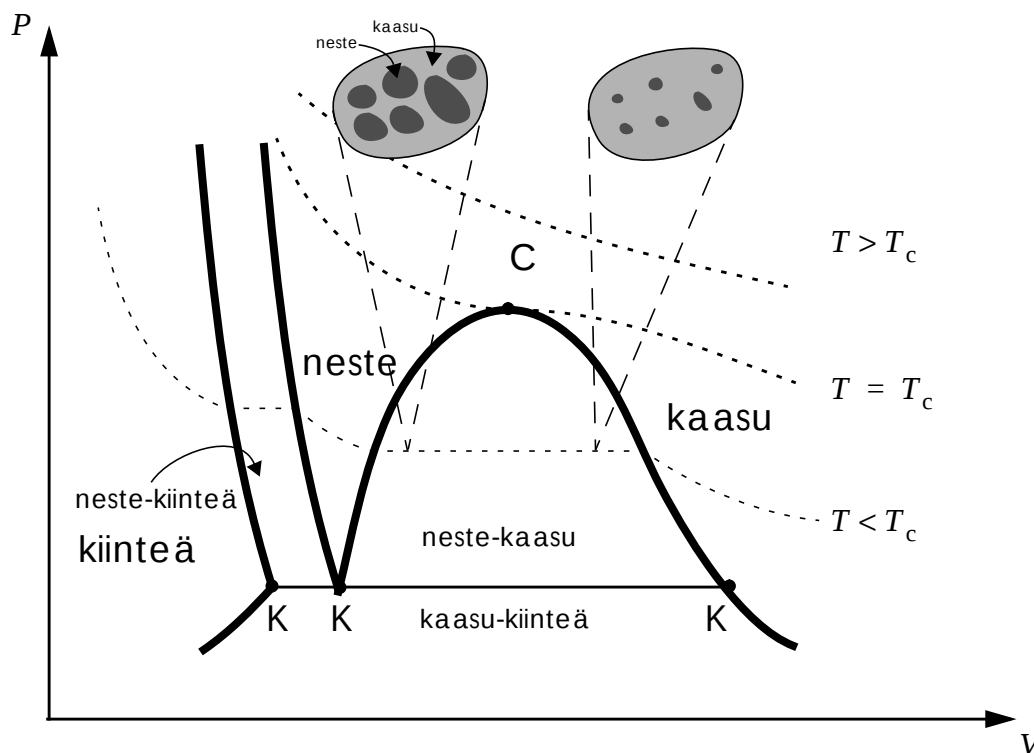
Nyt meillä on kolme yhtälöä ja kolme tuntematonta, eli ratkaisuna voi olla vain piste PT -tasossa. Tämä on ns. kolmoispiste (piste K kuvassa 7.4).



Kuva 7.4: NPT-systeemin faasidiagrammi PT -tasossa.

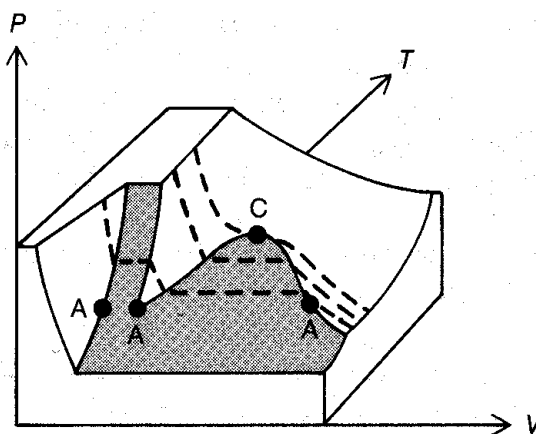
Nesteen ja kaasun välinen koeksistenssikäyrä päättyy ns. kriittiseen pisteeseen C (T_c, P_c). Tämän yläpuolella voimme muuttaa nesteen kaasuksi (tai päinvastoin) jatkuvalla tavalla. Vastaavanlaista pistettä ei ole havaittu kiinteän ja nesteen välillä. Tämä kertoo sen, että kiinteän aineen ja nesteen ero on perustavanlaatuisempi kuin nesteen ja kaasun. Kiinteässä aineessa on järjestystä, jota ei nesteestä tai kaasusta löydy.

PV -tasossa faasien koeksistenssi näkyy käyrien sijasta alueina.



Kuva 7.5: NPT-systeemin faasidiagrammi PV -tasossa.

Katkoviivat ovat isotermejä eri lämpötiloissa. Kuvassa 7.6 on faasidiagrammi hahmoteltuna kolmiulotteisena kuvaajana.



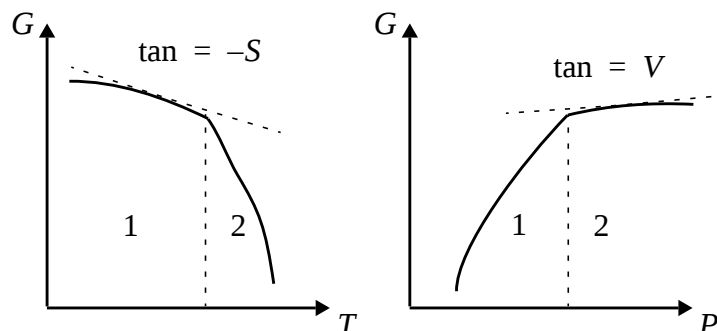
Kuva 7.6: NPT-systeemin faasidiagrammi.

Faasitransitiossa kemiallinen potentiaali eli $\partial G/\partial N$ on jatkuva. Muut Gibbsin energian osittaisderivaatat voivat olla joko jatkuvia tai epäjatkuvia.

Ylläkäsitellyt transitiot ovat ns. ensimmäisen kertaluvun transitoita. Näissä G :n derivaatat T :n ja P :n suhteen ovat epäjatkuvia:

$$\Delta S = -\left(\frac{\partial G}{\partial T}\right)_P^{(2)} + \left(\frac{\partial G}{\partial T}\right)_P^{(1)} \quad (7.8)$$

$$\Delta V = \left(\frac{\partial G}{\partial P}\right)_T^{(2)} - \left(\frac{\partial G}{\partial P}\right)_T^{(1)} \quad (7.9)$$



Kuva 7.7: Gibbsin vapaa energia 1. kertaluvun faasitransitiossa.

Ensimmäisen kertaluvun faasimuutokseen liittyy faasimuutoslämpö tai latentti lämpö, joka on entalpian muutos mentäessä faasista toiseen

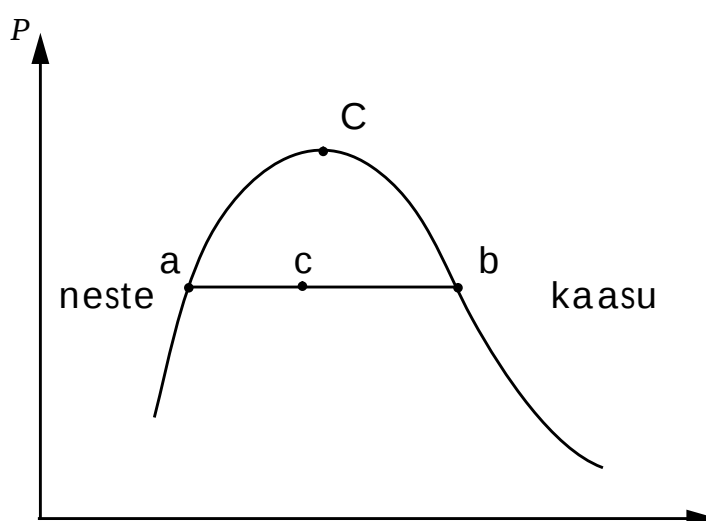
$$\Delta Q = T\Delta S = \Delta E + P\Delta V = \Delta(E + PV) = \Delta H \quad (7.10)$$

Helposti voidaan osoittaa, että mentäessä matalamman lämpötilan faasista korkeamman lämpötilan faasiin on latentti lämpö positiivinen eli lämpöä absorboituu.

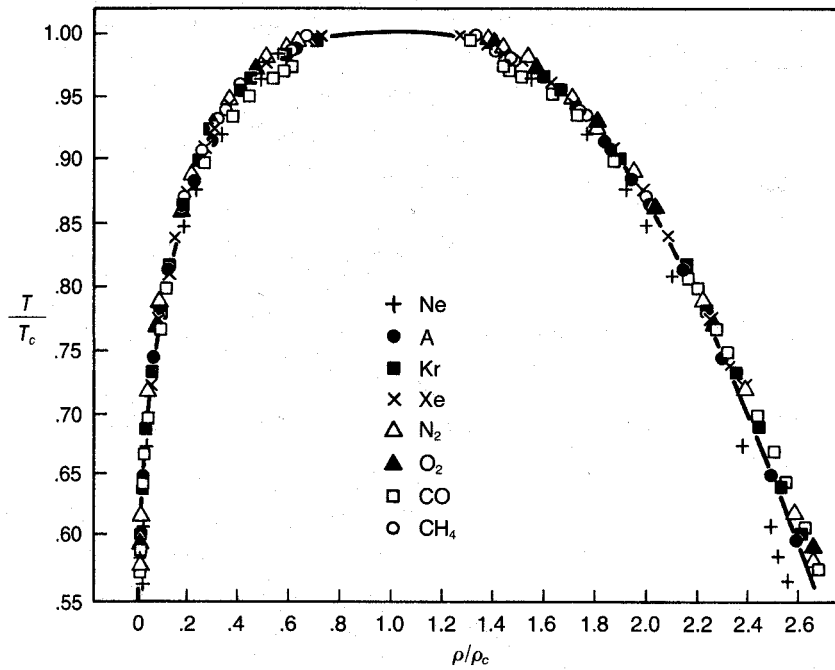
Normaali faasitransitio ei ole termodynaamisten suureiden kannalta mikään patologinen kohta. Transitio pisteessä faasien kemialliset potentiaalit ovat samat: $\mu^{(1)}(P, T) = \mu^{(2)}(P, T)$, mutta piste ei ole potentiaalien $\mu^{(1)}$ ja $\mu^{(2)}$ kannalta millään tavalla erikoinen.

Tiettyssä pisteessä (c kuvassa 7.8) systeemissä olevien faasien osuuksien suhde saadaan yksinkertaisesti janojen ac ja cb suhteesta.

Kuvassa 7.9 on esimerkkinä muuttaman kaasun neste-kaasukoeksistenssikäyrät.



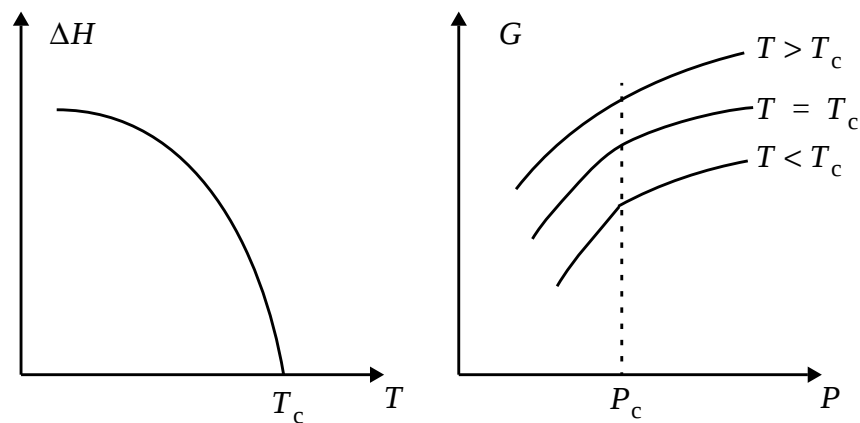
Kuva 7.8: Faasiosuudet koeksistenssialueella.



Kuva 7.9: Joidenkin kaasujen kokeelliset koeksistenssikäyrät. Redusoitujen yksiköiden 'c' viittaa kriittiseen pisteeseen.

Huomaa, että kaikki pisteet osuvat kauniisti samalle käyrälle.

Kuten jo mainittiin, kriittisessä pisteessä C häviää nesteen ja kaasun ero. Lähestyttäessä kriittistä pistettä latentti lämpö menee jatkuvalla tavalla nolnaan.

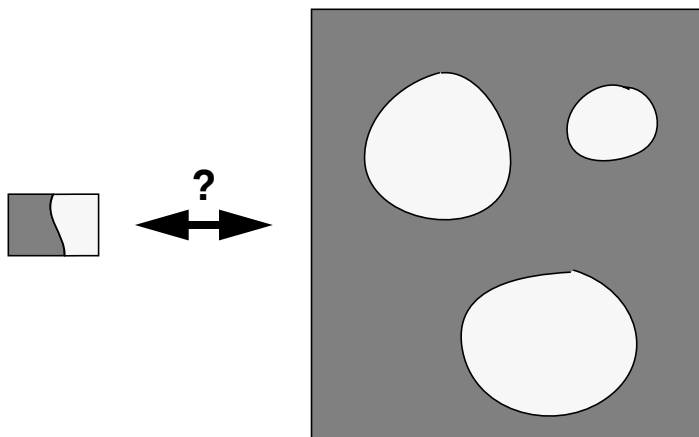


Kuva 7.10: Latentti lämpö ja Gibbsin vapaa energia kriittisen pisteen läheisyydessä.

7.2 Faasitasapainon simulointi

Faasitaspaine on sovellutusten kannalta varsin tärkeä ilmiö. Sen simuloiminen normaalilla MC-menetelmällä on kuitenkin hankalaa.

Jos simulointisysteemiin konstruoidaan eksplisiittinen faasien rajapinta, voivat tulokset olla vääristyneitä, koska tämä rajapinta vie huomattavan osan kopin tilavuudesta. Todellisessa tilanteessa (termodynaamisella rajalla) rajapinnan osuus on paljon pienempi.



Kuva 7.11: Rajapinnat faasitasapainosimulaatioissa.

Esimerkiksi aineen tiheys rajapinnan läheisyydessä voi poiketa todellisesta bulkin tiheydestä.

Eräs tapa on laskea MC:llä aineen tilanyhtälö ja sovittaa se esimerkiksi van der Waalsin (vdW) yhtälöön

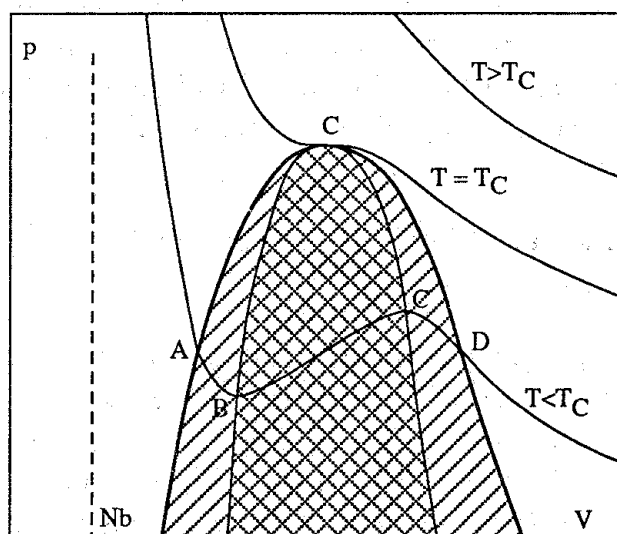
$$\left(P + \frac{a}{v^2}\right)(v - b) = k_B T, \quad (7.11)$$

missä $v = V/N$ ja a , b vakioita.

vdW-yhtälön avulla voimme laskea esimerkiksi kriittisen pisteen

$$v_c = 3b; \quad T_c = \frac{8a}{27b}; \quad P_c = \frac{a}{27b^2} \quad (7.12)$$

ja neste-kaasukoeksistenssikäyrän.

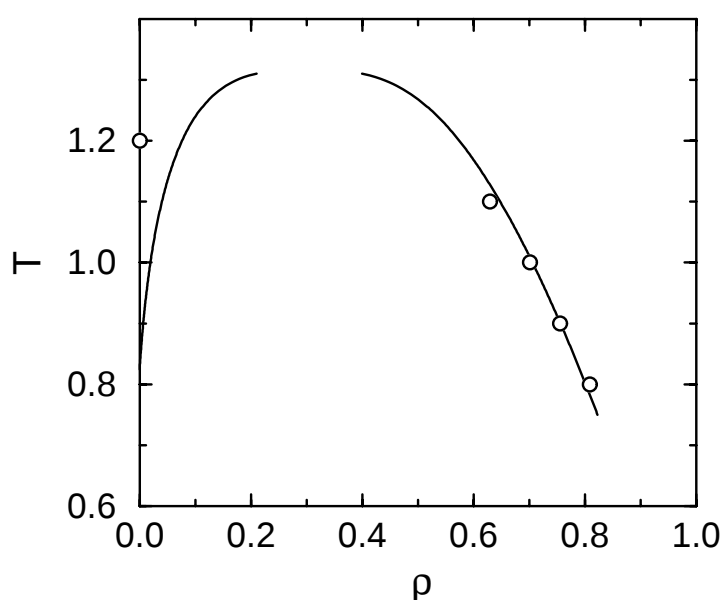


Kuva 7.12: van der Waalsin tilanyhtälö.

Tämä ei kuitenkaan aina toimi. Aine ei välttämättä noudata vdW-yhtälöä kovin hyvin. Ja kiinteälle olomuodolle yhtälö ei tietenkään käy.

Jonkinlainen arvio nesteen tiheydelle neste-kaasukoeksistenssissa voidaan saada vakiopaine-MC:llä nollapaineessa.

Simulointi aloitetaan hieman nesteen tiheyttä suuremmalla tiheydellä. Jos ajo ei ole liian pitkä, saamme jonkinlaisen arvion tiheydelle.

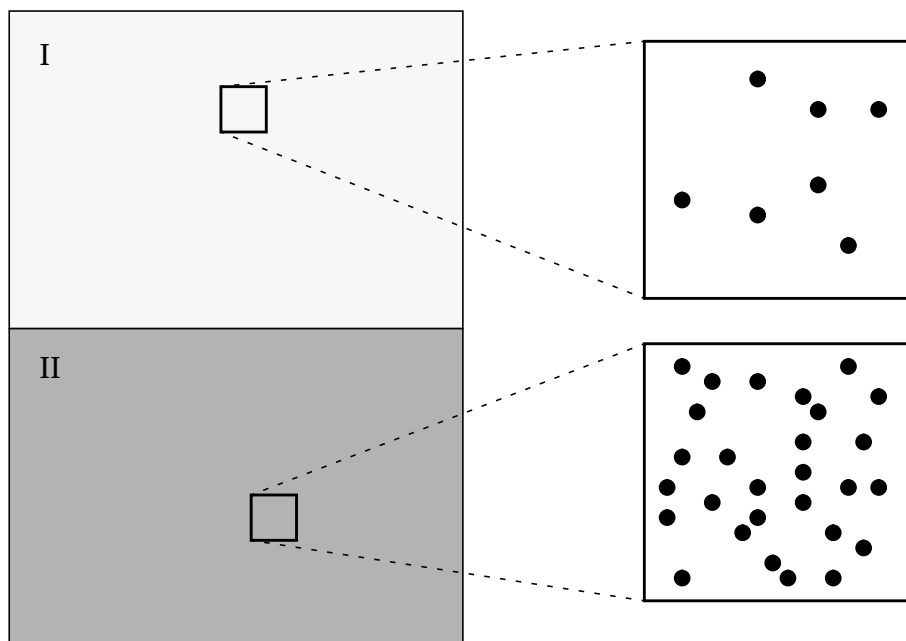


Kuva 7.13: Nestekaasukoeksistenssi vakiopainesimulaatiolla. [Frenkel-Smit]

Tämä on kuitenkin melko karkea menetelmä, sillä tiheydessä voi tapahtua suuriakin fluktuaatioita, ja jos fluktuaatio on pienen tiheyden suuntaan voi systeemin koko kasvaa rajatta.

Varsin yksinkertainen simulointimenetelmä faasitasapainon laskemiseen esitettiin 80-luvun lopulla: Gibbsin ensemble¹(GE). Se on varsin yksinkertainen ja siinä vältetään täysin rajapintojen tuottamat ongelmat: lasketaan vain bulkkiominaisuuksia.

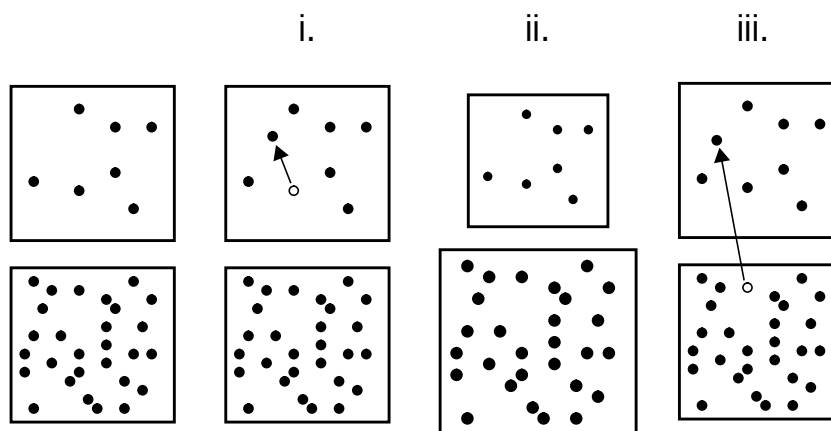
Oletetaan, että tutkimme kahden faasin, I ja II tasapainoa. GE-simuloinnissa kaksi simulointikoppia, yksi molemmista faaseista.



Kuva 7.14: Gibbsin ensemble -simulaatio.

Simuloinnissa on kolmenlaisia tilanmuutoksia:

- i. Normaali atomin siirto jommassa kummassa faasissa.
- ii. Tilavuuden muutos jommassa kummassa faasissa. Tutkimme nyt kanonisesta ensembleä, eli systeemin kokonaistilavuus on vakio. Tästä seuraa, että faasin I tilavuuden muuttuessa määrällä ΔV , on faasin II tilavuuden muutos $-\Delta V$.
- iii. Atomin (tai molekyylin) siirto faasista toiseen.



Kuva 7.15: Tilanmuutoksen Gibbsin ensemble -simulaatiossa.

1. A.Z.Panagiotopoulos ym., *Mol. Phys.* **63** (1988) 527.

Hiukkasen siirto faasin sisällä hyväksytään normaalilla todennäköisyydellä

$$p_{\text{mov}} = \exp[-\delta v_{nm}/k_B T]. \quad (7.13)$$

Tilavuuden muutoksen todennäköisyys faasissa I on

$$p_{\text{vol}}^I = \exp[-\beta P \Delta V - \beta \delta v^I + N^I \ln(V^I + \Delta V) - N^I \ln V^I], \quad (7.14)$$

missä ΔV on tilavuuden muutos, δv^I on potentiaalienergian muutos faasissa I, V^I ja N^I ovat faasin I tilavuus ja hiukkaslukumäärä.

Faasissa II tapahtuu tilavuuden muutos $-\Delta V$, jonka todennäköisyys on

$$p_{\text{vol}}^{II} = \exp[\beta P \Delta V - \beta \delta v^{II} + N^{II} \ln(V^{II} - \Delta V) - N^{II} \ln V^{II}]. \quad (7.15)$$

Todennäköisyys, että nämä tapahtuvat yhtäaikaan on (olettaen, että ne ovat toisistaan riippumattomia)

$$\begin{aligned} p_{\text{vol}} &= p_{\text{vol}}^I \times p_{\text{vol}}^{II} \\ &= \exp\left(-\beta\left[\delta v^I + \delta v^{II} - \frac{N^I}{\beta} \ln \frac{V^I + \Delta V}{V^I} - \frac{N^{II}}{\beta} \ln \frac{V^{II} - \Delta V}{V^{II}}\right]\right) \end{aligned} \quad (7.16)$$

Todennäköisyys atomin vaihdolle faasista II faasiin I koostuu atomin luomisesta I:ssä:

$$p_{\text{int}}^I = \exp\left(-\beta \delta v^I + \ln \frac{z V^I}{N^I + 1}\right), \quad (7.17)$$

missä aktiivisuus on $z = e^{\beta \mu} / \Lambda^3$.

Vastaavasti atomin tuhoaminen faasissa II :

$$p_{\text{int}}^{II} = \exp\left(-\beta \delta v^{II} + \ln \frac{N^{II}}{z V^{II}}\right). \quad (7.18)$$

Atomin vaihdon kokonaistodennäköisyys on näiden tulo

$$p_{\text{int}} = \exp\left(-\beta\left[\delta v^I + \delta v^{II} + \beta^{-1} \ln \frac{V^{II}(N^I + 1)}{V^I N^{II}}\right]\right). \quad (7.19)$$

Huomaa, että yhtälöissä (7.16) ja (7.19) ei esiinny ollenkaan painetta tai kemiallista potentiaalia. Vain systeemin kokonaistilavuus ja lämpötila ovat määrättyt. Paine ja kemiallinen potentiaali asettuvat faasikoeksistentin vaatimiin arvoihin.

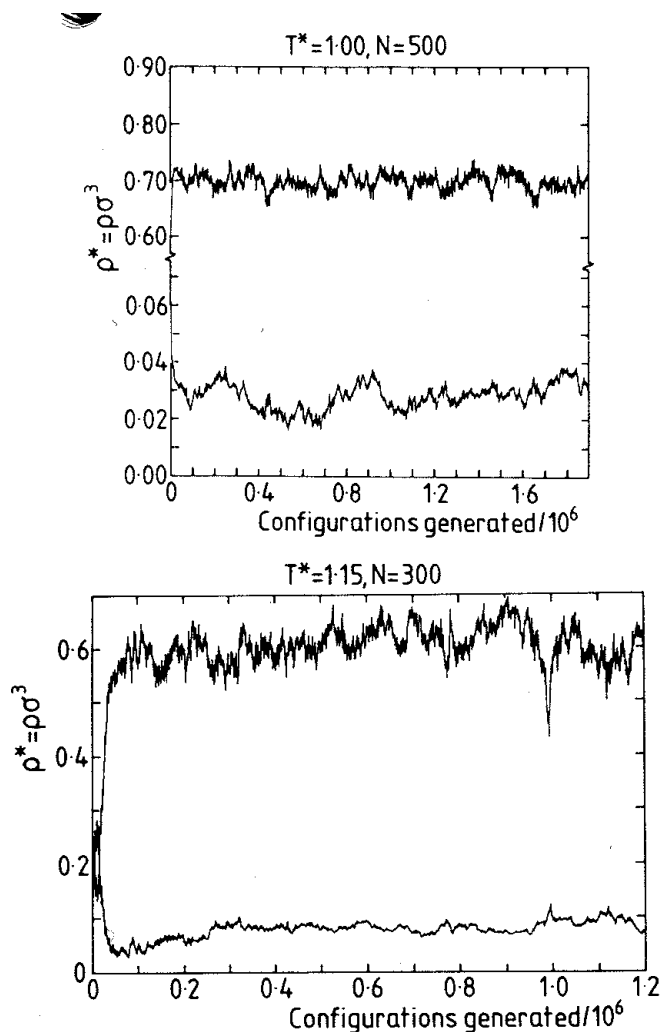
Ne voidaan tietysti laskea simulaation kuluessa. Se onkin hyvä tarkistuskeino, sillä näiden suureiden tulisi olla samat molemmissa faaseissa.

Eri siirtotapojen suhteellisen osuuden sanelee pyrkimys nopeaan konvergenssiin Markovin ketjussa. Käytännössä joka askeleella käydään läpi molempien faasien kaikki atomit ja yritetään normaalia MC-siirtoa. Tämän jälkeen voidaan tehdä tilavuuden muutos. Mikroskooppisen reversiibeliyden säilyttämiseksi valitaan tilavuuden muutoksen merkki (kutistuuko faasi I vai faasi II) satunnaisesti. Lopulta yritetään atomin vaihtoa faasien välillä. Mikroskooppinen reversiibeliys vaatii taas, että vaihdon suunta valitaan satunnaisesti. Näitä vaihtoyrityksiä voidaan tehdä useampia. Tämä on tarpeellista varsinkin suurilla tiheyksillä.

Otetaan esimerkkinä puhtaan LJ-systeemin neste-kaasukoeksistenssi¹. Alkutilana oli kaksi koppia, joihin oli generoitu fcc-hila. Koppeihin laitettiin atomeja halutun tiheyden osoittama määrä. Tiheydet olivat luonnollisesti erilaiset, vaikkakaan eivät täsmälleen nesteen ja kaasun tiheydet ko. lämpötiloissa.

Kuvassa 7.16 on esitetty systeemin tiheyksien käyttäytyminen simulaation kuluessa lämpötiloilla $T^* = 1.00$ ja $T^* = 1.15$.

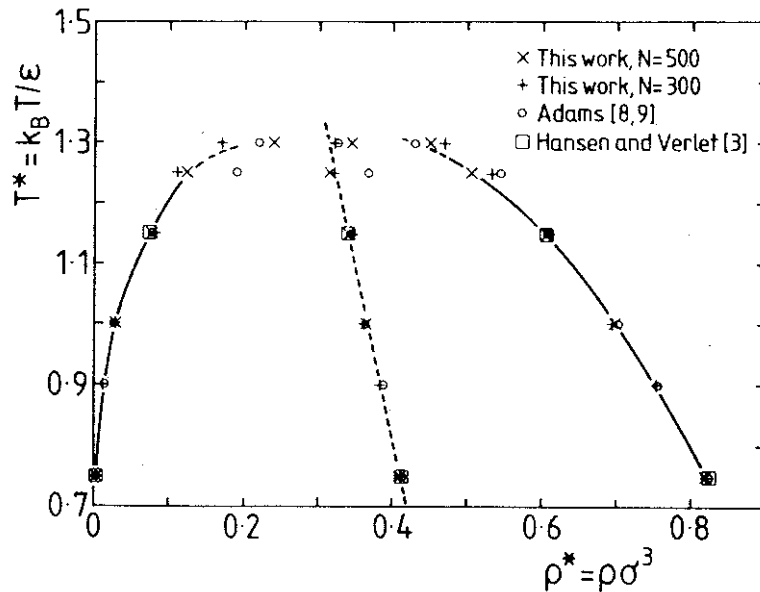
Kuva 7.16: Lennard-Jones-systeemin tiheydet Gibbsin ensemble -simulaatiossa kahdessa eri lämpötilassa.



1. A.Z.Panagiotopoulos, *Mol. Phys.* **61** (1987) 813.

Vaikka alkutilan tiheydet eivät täysin vastaa tasapainotilan tiheyksiä, tasapaino saavutetaan.

Kuvassa on esitetty työn lopullinen tulos, LJ-systeemin neste-kaasukoeksistenssikäyrä.



Kuva 7.17: Lennard-Jones-systeemin neste-kaasu-koeksistenssikäyrä Gibbsin ensemble -menetelmällä laskettuna.

NVT-ensemblen lisäksi Gibbsin ensemble -menetelmä voidaan yleistää myös NPT-ensembleen. Puhtaan aineen tapauksessa NPT-ensemblen käyttö ei ole kuitenkaan kovin järkevää, sillä faasitasapainon ehtojen mukaan vain yhden intensiivisen termodynaamisen muuttujan arvo voidaan vapaasti valita. Tässä tapauksessa se on lämpötila. Paineen arvo määräytyy tasapainoehdoista.

Jos puhtaan aineen GE/NPT-simulaatiossa asetetaan paine siten, että se ei vastaa faasikoeksistenssin vaatimaa arvoa, on tuloksena homogeeninen systeemi.

Käsiteltäessä monikomponenttisia seoksia asia on toisin. Tällöin Termodynaamisia vapausasteita on enemmän (eri komponenttien suhteelliset osuudet), jolloin faasikoeksistenssi voi löytyä kaikille PT -pareille.

Vakiopainesimulaation aiheuttamat muutokset GE-algoritmiin ovat varsin vähäiset. Ainoastaan ehto tilavuuden muutosten täydellisestä korrelaatiosta eri faaseissa poistuu. Tilavuuden muutoksen todennäköisyys, yhtälö (7.16) tulee nyt muotoon

$$\begin{aligned}
 p_{\text{vol}} &= p_{\text{vol}}^{\text{I}} \times p_{\text{vol}}^{\text{II}} \\
 &= \exp\left(-\beta\left[\delta v^{\text{I}} + \delta v^{\text{II}} - \frac{N^{\text{I}}}{\beta} \ln \frac{V^{\text{I}} + \Delta V^{\text{I}}}{V^{\text{I}}} - \frac{N^{\text{II}}}{\beta} \ln \frac{V^{\text{II}} + \Delta V^{\text{II}}}{V^{\text{II}}} \right. \right. \\
 &\quad \left. \left. + P(\Delta V^{\text{I}} + \Delta V^{\text{II}})\right]\right)
 \end{aligned} \tag{7.20}$$

Eli kummallakin faasilla on omat tilavuudenmuutoksensa ΔV^{I} ja ΔV^{II} .

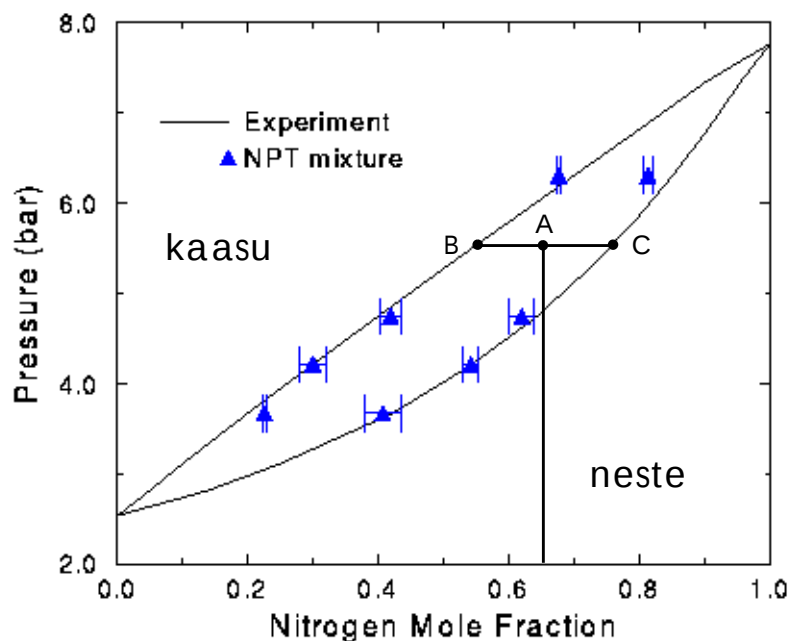
Myös yleistys useampikomponenttisiin aineisiin on suoraviivainen. Jokaisen komponentin kemiallinen potentiaali on sama molemmissa faaseissa. Yhtälössä (7.19) tulee hiukkaslukumääriin mukaan indeksi, joka kertoo atomi- tai molekyyllilajin i :

$$N^I \rightarrow N_i^I, N^{II} \rightarrow N_i^{II}. \quad (7.21)$$

Detaljibaöanssin säilyttämiseksi hiukkasten vaihto suoritetaan seuraavasti:

- i. Valitaan joko faasi I tai II hiukkasen luomiseen yhtä suurella todennäköisyydellä.
- ii. Valitaan vakiotodennäköisyydellä vaihdettavan hiukkasen laji. Todennäköisyys voi olla sama kaikille lajeille tai sitten ei. Kriteerinä tuttuun tapaan konvergenssi tasapainotilaa kohti.

Baranello tutki yllämainituilla menetelmillä typen ja hapen seoksen faasikoeksistenssiä. Lämpötila oli 100 K. Tulokset on esitetty kuvassa 7.18.



Kuva 7.18: Typen ja hapen faasikoeksistenssi GE-menetelmällä laskettuna.

Nähdään, että kaasukäyrän simulaatiot ennustavat melko hyvin, nestekäyrässä taas on hieman heittoa.

8 Ajasta riippuvien ilmiöiden simulointi

8.1 Yleistä

Tähän asti olemme puhuneet tasapainoilmiöiden eli ajasta riippumattomien ilmiöiden MC-simuloinnista.

Voimme sanoa, että tasapaino-ominaisuuksien MC-simuloinnissa ratkaisemme numeerisesti masteryhtälöä¹

$$\frac{\partial P(n, t)}{\partial t} = \sum_m [W(m \rightarrow n)P(m, t) - W(n \rightarrow m)P(n, t)] , \quad (8.1)$$

missä $P(n, t)$ on todennäköisyys, että systeemi on tilassa n ajanhetkellä t . On kuitenkin huomattava, että tuo aika (jonka voi ajatella vastaavan MC-askelta) ei ole fysikaalinen aika. Systeemin dynamiikan määräävät siirtymätodennäköisyydet² $W(m \rightarrow n)$, jotka voidaan valita melko vapaasti. Ainoa vaatimus on, että simuloidussa Markov-ketjussa päädymme lopulta haluttuun tasapainojakaumaan. Eli detaljibalanssin

$$W(m \rightarrow n)P(m, \text{eq}) = W(n \rightarrow m)P(n, \text{eq}) \quad (8.2)$$

tulee olla voimassa. Tässä $P(n, \text{eq})$ on tasapainotodennäköisyystiheys, joka on—kuten muistamme—esimerkiksi kanonisen ensemblen tapauksessa

$$P(n, \text{eq}) = \frac{1}{Z} e^{-H(n)/k_B T} . \quad (8.3)$$

Fysikaalisen systeemin todellisen aikakehityksen määräävät liikeyhtälöt, joita ei staattisessa MC:ssä tarvita.

Jos haluamme simulaatiomme mallintavan myös systeemin todellista aikakehitystä, ei pelkkä detaljibalanssin toteuttaminen riitä. Tähän palaamme myöhemmin. Ensin kuitenkin käymme läpi yksinkertaisempia tapauksia ei-tasapainoilmiöiden mallintamisesta.

Joissain tapauksissa yhtälön (8.1) antama todennäköisyysjakauma $P(n, t)$ on turhan yksityiskohtaista tietoa systeemistä. Esimerkiksi atomisysteemin tapauksessa tuo jakauma on voisi olla todennäköisyystiheys, että atomin 1 paikka ja nopeus ovat $\mathbf{r}_1$, ja $\mathbf{v}_1$, atomin 2 paikka ja nopeus ovat $\mathbf{r}_2$, ja $\mathbf{v}_2$ jne. ajanhetkellä t :

$$P(\mathbf{r}_1, \mathbf{v}_1, \mathbf{r}_2, \mathbf{v}_2, \dots, \mathbf{r}_N, \mathbf{v}_N, t) .$$

Usein kuitenkin riittäisi yksihiukkastodennäköisyysjakauma, eli todennäköisyys, että systeemissä on atomi, jolla on paikassa $\mathbf{r}$ nopeus $\mathbf{v}$ ajanhetkellä t :

1. Tässä esitettynä diskreetissä muodossa.
2. Aikayksikköä kohti.

$$P_1(\mathbf{r}, \mathbf{v}, t) .$$

Masteryhtälöstä voidaan johtaa yhtälö tälle yksihiukkasjakaumalle, mutta sitä ei voida ilman approksimaatioita ratkaista, sillä se sisältää kaksi-, kolmi- ja useampihiukkasjakaumia¹. Eräs paljon käytetty approksimaatio on Boltzmannin kuljetusyhtälö, joka perustuu ns. molekulaarisen kaaoksen oletukseen. Tämän oletuksen mukaan hiukkasten väliset korrelaatiot unohdetaan ja todennäköisyys, että löydämme tietyistä tilavuusalkioista hiukkaset 1 ja 2 on näiden hiukkasten todennäköisyysjakaumien tulo

$$P_2(\mathbf{r}_1, \mathbf{v}_1, \mathbf{r}_2, \mathbf{v}_2, t) = P_1(\mathbf{r}_1, \mathbf{v}_1, t)P_1(\mathbf{r}_2, \mathbf{v}_2, t) . \quad (8.4)$$

Yksinkertaisemmassa muodossa (atomaarinen kaasu) Boltzmannin yhtälö voidaan kirjoittaa seuraavassa muodossa

$$\frac{\partial P_1}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{r}} P_1 + \frac{1}{m} \mathbf{F} \cdot \nabla_{\mathbf{v}} P_1 = \left(\frac{\partial P_1}{\partial t} \right)_{\text{coll}} , \quad (8.5)$$

missä vasemman puolen toinen termi kuvaa atomien nopeuksista aiheutuvaa todennäköisyyden muutosta muutosta, $((\partial P)/\partial t)_{\text{coll}}$ on atomien välisistä törmäyksistä aiheutuva muutos. Lisäksi mukana on ulkoisen voiman $\mathbf{F}$ aiheuttama muutos. Molekulaarisen kaaoksen oletuksen mukaan törmäystodennäköisyys on verrannollinen hiukkastiheyksien tuloon faasiavaruuden tilavuusalkiossa. Tällöin voidaan törmäystermille johtaa muoto

$$\left(\frac{P_1(\mathbf{r}, \mathbf{v}, t)}{\partial t} \right)_{\text{coll}} = \int d\mathbf{v}_2 d\Omega'_1 v_{\text{rel}} \sigma(\theta, v_{\text{rel}}) [P_1(\mathbf{r}, \mathbf{v}_1', t)P_1(\mathbf{r}, \mathbf{v}_2', t) - P_1(\mathbf{r}, \mathbf{v}_1, t)P_1(\mathbf{r}, \mathbf{v}_2, t)] , \quad (8.6)$$

missä v_{rel} on törmäävien atomien suhteellinen nopeus, $\sigma(\theta, v_{\text{rel}})$ on vaikutusala sironnalle, θ on MKP-sirontakulma.

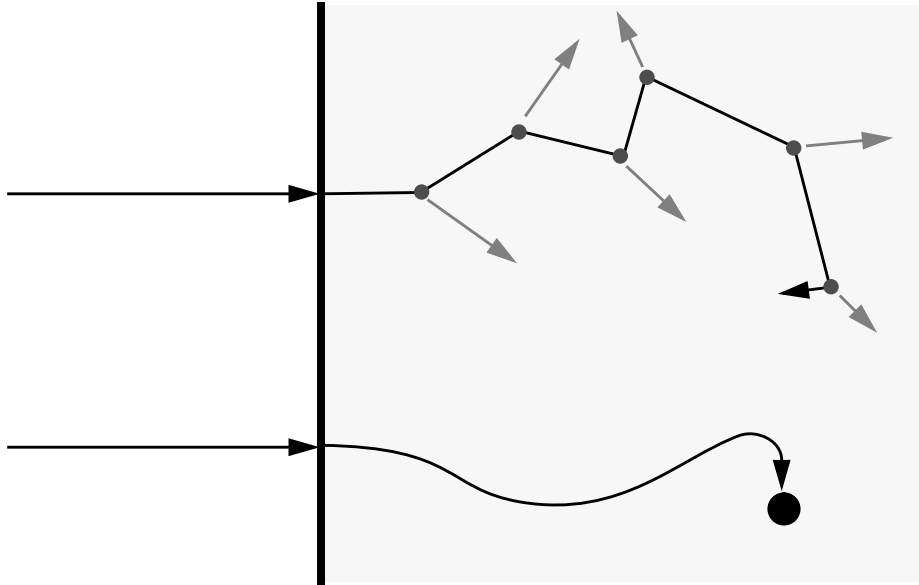
8.2 Säteilyn kulkeutumisen simulointi

Säteilyn² kulkeutuminen väliaineessa on eräs laaja MC-simulaatioiden sovelluskohde.

Kulkeutuminen on luonnollisesti erittäin epätasapainoilmiö, vaikka useimmiten emme ole kiinnostuneita systeemin aikakehityksestä, ainoastaan lopputuloksesta, jona on esimerkiksi säteilyn väliaineeseen lopulta jättämä energia tai vauriot tai säteilyhiukkasten lopullinen paikka väliaineessa.

1. L.E.Reichl, *A Modern Course in Statistical Physics*, luvut 7 ja 13.

2. Hiukkasista koostuva säteily. Mm. ioni-, elektroni-, fotoni-, neutroni- ...



Kuva 8.1: Säteilyn kulkeutuminen väliaineessa.

Energeettisen säteilyhiukkasen saapuessa väliaineeseen se alkaa vuorovaikuttaa aineen elektronien ja atomien kanssa menettäen energiaansa, kunnes se pysähtyy tai annihiloituu.

Vuorovaikutustapahtumat oletetaan diskreeteiksi äärettömän lyhyen aikaa kestäviksi prosesseiksi. Useimmiten tämä pätee, mutta on tapauksia, joita mallinnettaessa on otettava huomioon prosessien äärellinen kesto.

Säteilyn kulkeutumista voidaan kuvata Boltzmannin yhtälön avulla. Säteilyä karakterisoidaan vuolla ϕ_i :

$$\phi_i(\mathbf{r}, E, \mathbf{u}, t) d\mathbf{r} dE d\Omega_{\mathbf{u}} dt \quad (8.7)$$

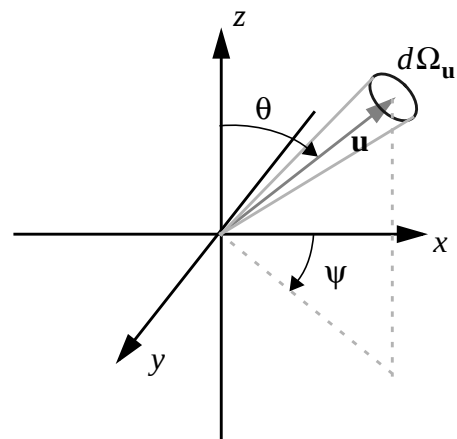
kertoo niiden tyypin i säteilyhiukkasten lukumäärän aikavälillä $[t, t + dt]$, joiden paikka on $\mathbf{r}$:n $d\mathbf{r}$ -ympäristö, energia on välillä $[E, E + dE]$ ja joiden nopeuden suunta on avaruuskulma-alkiossa $d\Omega_{\mathbf{u}}$.

Tarkoituksena on pystyä alkuehdoista lähtien laskemaan säteilyn vuo eri aikoina ja eri paikoissa. Tästä pystymme edeleen laskemaan haluamiamme asioita.

Integroimalla pois yksi tai useampi muuttuja (8.7):sta saadaan eri suureita, esimerkiksi:

$$\text{vuon tiheys } \phi_i(\mathbf{r}, t) = \int d\Omega_{\mathbf{u}} \int dE \phi_i(\mathbf{r}, E, \mathbf{u}, t), \quad (8.8)$$

$$\text{virrantiheys } \mathbf{J}_i^z(\mathbf{r}, t) = 2\pi \int_0^\pi d\theta \sin \theta \int dE \cos \theta \phi_i(\mathbf{r}, E, \mathbf{u}, t). \quad (8.9)$$



Kuva 8.2: Säteilyvuon määrittelmä

Boltzmannin yhtälö kertoo, miten vuo ϕ muuttuu eri prosessien vaikutuksesta. Säteilyn etenemisen ollessa kyseessä ovat yhtälön termit seuraavan näköisiä.

Hiukkasten äärellisestä nopeudesta aiheutuu muutos

$$\frac{1}{v_i} \left[\frac{\partial \phi_i(\mathbf{r}, E, \mathbf{u}, t)}{\partial t} \right]_1 = -\nabla \cdot \mathbf{u} \phi_i(\mathbf{r}, E, \mathbf{u}, t) = -\mathbf{u} \cdot \nabla \phi_i(\mathbf{r}, E, \mathbf{u}, t) . \quad (8.10)$$

Hiukkasten siroonnasta väliaineen hiukkasten kanssa tulee termi

$$\begin{aligned} \frac{1}{v_i} \left[\frac{\partial \phi_i(\mathbf{r}, E, \mathbf{u}, t)}{\partial t} \right]_2 , \quad (8.11) \\ = \sum_j \left[\int (d\Omega'_{\mathbf{u}}) dE' \sigma_{ij}(\mathbf{r}, E', E, \mathbf{u}', \mathbf{u}) \phi_j(\mathbf{r}, E', \mathbf{u}', t) \right. \\ \left. - \int (d\Omega'_{\mathbf{u}}) dE' \sigma_{ij}(\mathbf{r}, E, E', \mathbf{u}, \mathbf{u}') \phi_j(\mathbf{r}, E, \mathbf{u}, t) \right] \\ - \sigma_i(\mathbf{r}, E) \phi_i(\mathbf{r}, E, \mathbf{u}, t) \end{aligned}$$

missä $\sigma_{ij}(\mathbf{r}, E', E, \mathbf{u}', \mathbf{u})$ on vaikutusala siroonnalle tyyppin i hiukkasesta, jolla on koordinaatit $(\mathbf{r}, E', \mathbf{u}', t)$, ja jossa syntyy i -tyyppinen hiukkanen, jolla on koordinaatit $(\mathbf{r}, E, \mathbf{u}, t)$. Vastavasti $\sigma_{ij}(\mathbf{r}, E, E', \mathbf{u}, \mathbf{u}')$ kertoo siroannan kautta faasiavaruuden pisteestä $(\mathbf{r}, E, \mathbf{u}, t)$ poistuvien hiukkasten lukumäärän. Viimeinen termi kuvaa hiukkasen absorptiota väliaineeseen.

Näiden termien lisäksi voidaan osaa vuorovaikutuksista mallintaa jatkuvana kitkanomaisena prosessina. Tästä tulee termi

$$\frac{1}{v_i} \left[\frac{\partial \phi_i}{\partial t} \right]_3 = \frac{(\partial \phi_i S)}{\partial E} , \quad (8.12)$$

missä S on ns. jarrutuskyky (stopping power).

Näiden lisäksi (riippuen säteilyn luonteesta) voidaan yhtälössä olla mukana termi, joka kuvaa säteilyhiukkasten hajoamista:

$$\frac{1}{v_i} \left[\frac{\partial \phi_i}{\partial t} \right]_4 = \left(-\frac{1}{\lambda_i} \right) \phi_i , \quad (8.13)$$

missä λ_i on hajoamistodennäköisyys pituusyksikköä kohti.

Jos systeemeissä on säteilylähde tarvitaan termi kuvaamaan uusien hiukkasten syntymistä:

$$\frac{1}{v_i} \left[\frac{\partial \phi_i}{\partial t} \right]_5 = Y(\mathbf{r}, E, \mathbf{u}, t) . \quad (8.14)$$

Yhdistämällä (8.10)-(8.14) saadaan Boltzmannin yhtälö muotoon

$$\begin{aligned}
 & \frac{1}{v_i} \left(\frac{\partial \phi_i}{\partial t} \right) \\
 & = -\mathbf{u} \cdot \nabla \phi_i(\mathbf{r}, E, \mathbf{u}, t) \\
 & + \sum_j \left[\int (d\Omega'_{\mathbf{u}}) dE' \sigma_{ij}(\mathbf{r}, E', E, \mathbf{u}', \mathbf{u}) \phi_j(\mathbf{r}, E', \mathbf{u}', t) \right. \\
 & \left. - \int (d\Omega'_{\mathbf{u}}) dE' \sigma_{ij}(\mathbf{r}, E, E', \mathbf{u}, \mathbf{u}') \phi_j(\mathbf{r}, E, \mathbf{u}, t) \right] \\
 & - \sigma_i(\mathbf{r}, E) \phi_i(\mathbf{r}, E, \mathbf{u}, t) \\
 & + \frac{\partial [\phi_i(\mathbf{r}, E, \mathbf{u}, t) S]}{\partial E} - \frac{1}{\lambda_i} \phi_i(\mathbf{r}, E, \mathbf{u}, t) + Y(\mathbf{r}, E, \mathbf{u}, t)
 \end{aligned} \tag{8.15}$$

Lisäksi näitä yhtälöitä voi olla useampi ja ne voivat olla kytkettyjä: esimerkiksi gammasäteily synnyttää energettisiä elektroneja väliaineeseen ja päinvastoin.

Boltzmannin yhtälön ratkaisua vaikeuttaa lisäksi se, että realistisissa systeemeissä on aina rajapintoja, joiden käsittely on hankalaa. Lisäksi tarvittavat vaikutusalat usein melko hankalia lausekkeita.

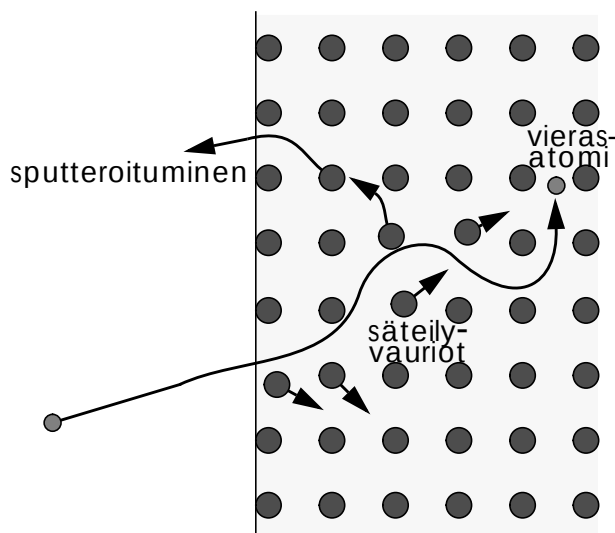
Toisaalta – kuten analyttiset menetelmät yleensäkin – Boltzmannin yhtälöä voi joissain tapauksissa käyttää karkeiden arvioiden laskemiseen ja yleisten trendien tutkimiseen.

Jos tarvitsemme tuloksia realistisille systeemeille ja tuloksia, joita voidaan verrata koetuloksiin on useimmiten turvaututtava simulointiin.

Tarkastellaan lähemmin MC-menetelmän sovellutusta ionisuihkujen ja aineen vuorovaikutukseen. Tämä on ehkä hieman spesifinen esimerkki, mutta valaissee MC-menetelmän käyttöä kuljetusilmiöiden tutkimisessa.

Ionisuihkufysiikassa tutkitaan energettisten ionien (tai atomien) ja aineen vuorovaikutusta. Alkuperäinen ioni saadaan aikaan ampumalla kiihdyttimestä saatu ioni aineeseen. Energeettisellä ymmärretään termisiä energioita huomattavasti suurempia energioita (enemmän tai vähemmän mielivaltaisesti $E > 1\text{eV}$; $1\text{eV} \sim 11600\text{K}$). (Ioni voi tietysti olla myös neutraali, mutta käytäntö on, että alkuperäistä energettistä hiukkasta kutsutaan ioniksi.)

Alue on yleisen mielenkiinnon lisäksi tärkeä myös sovellutusten kannalta. Ionisuihkujen avulla voimme muuttaa aineiden ominaisuuksia tuomalla vierasaineita (ioni-implantointi, kantamajakaumat), sekoittamalla rajapintoja (ionisuihkusekoitus), rikkomalla kidettä (säteilylvauriot), poistamalla atomeja pinnalta (sputteroituminen) jne.

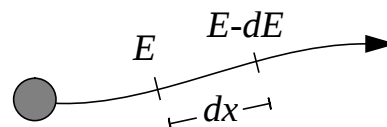


Kuva 8.3: Ionisuihkun ja aineen välinen vuorovaikutus.

Seuraavassa käydään läpi ionin ja aineen vuorovaikutusmekanismit. Aineeseen tunkeutuessaan ioni alkaa jarruuntua. Ioni menettää energiaa törmäyksissä aineen elektronien kanssa: *elektroninen jarruuntuminen*. Ioni siroaa elastisesti aineen atomeista: *nukleaarinen jarruuntuminen*.

Jarruuntumista voidaan kuvata jarrutuskyvyllä S , joka on ionin menettämä kineettinen energia sen kulkemaa matkaa kohti

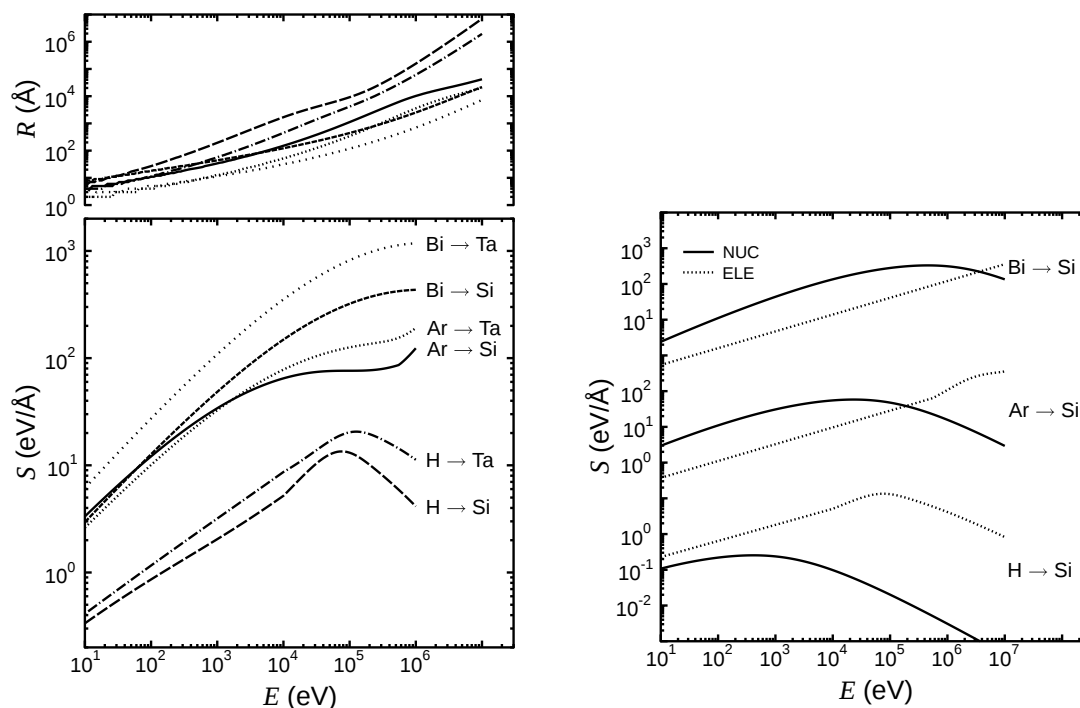
$$S = -\frac{dE}{dx} . \quad (8.16)$$



Kuva 8.4: Jarrutuskyvyn määritelmä.

Jarrutuskyky voidaan jakaa elektroniseen ja nukleariin osaan: $S = S_e + S_n$. Jarrutuskyky riippuu ionin ja aineen ominaisuuksista ja ionin nopeudesta. Sen yksikkönä on yleensä eV/Å, keV/nm, eV/(10^{15} at./cm²) tai keV/(μg/cm²).

Nukleaarinen jarruuntuminen on vallitseva pienillä energioilla, elektroninen taas suurilla. Kuvassa 8.5 on muutama S_n :ää ja S_e :tä vertaileva esimerkki.



Kuva 8.5: Esimerkkejä jarrutuskyvyistä. Vasemmalla puolella on piirretty myös ionien keskikantamat.

Koska ionin törmäyksissä elektronien kanssa ei massaerosta johtuen ionin suunta muutu, voidaan S_e olettaa kitkanomaiseksi voimaksi. Elektroninen jarruuntuminen riippuu periaatteessa ionin kokemasta (paikallisesta) elektronitiheydestä, mutta yleensä approksimoidaan, että S_e riippuu ainoastaan ionin energiasta.

Elektronisen jarruuntumisen laskeminen on varsin vaikea teoreettinen ongelma, kun ottaa huomioon, ne mekanismit, joilla energiaa voi siirtyä ionin ja aineen välillä:

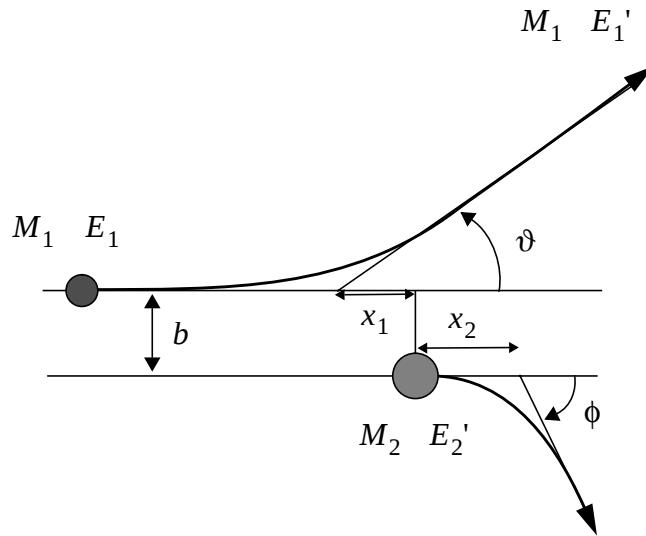
- i. kohtioatomien ionisaatio ja vahvasti sidottujen elektronien viritys
- ii. kohtion johde-elektronien viritys (ei-lokalisoituneet elektronit), kollektiiviset eksitaatiot (plasmonit)
- iii. ionin ionisaatio ja elektronien viritys sekä ionin elektronikaappaus

Tästä syystä elektroninen jarrutuskyky käytännössä lasketaan semiempiirisien mallien avulla tai käytetään kokeellisia tuloksia. Kaikkein eniten ionisuihkufysiikassa käytetty semiempiirinen malli on Zieglerin, Biersackin ja Littmarkin (ns. ZBL) malli¹.

Nukleaarisessa jarruuntumisessa siis ioni siroaa elastisesti atomin varjostetusta Coulombin potentiaalista. Ionisuihkufysiikassa ollaan tekemisissä atomien välisen potentiaalin repulsiivisella osalla.

1. Ziegler, Biersack, Littmark, *The Stopping and Range of Ions in Solids*, vol. 1 (Pergamon, New York, 1985)

Koska ionin ja atomin massat ovat samaa suuruusluokkaa, ionin suunta voi muuttua paljonkin sironnassa, ja ääritapauksessa ioni menettää kaiken kineettisen energiansa yhdessä törmäyksessä. Nukleaarista jarruuntumista ei siis voi realistisesti kuvata pelkkänä kitkavoimana.



Kuva 8.6: Sironna potentiaalista.

Binääritörmäys pystytään helposti käsittelemään klassisena sirontana potentiaalista $V(r)$. Sirontakulma massakeskipistekoordinaatistossa (MKP) θ saadaan sirontaintegraalista

$$\theta = \pi - 2b \int_R^{\infty} [r^2 g(r)]^{-1/2} dr, \quad (8.17)$$

$$g(r) = \left[1 - \frac{V(r)}{E_r} - \left(\frac{b}{r} \right)^2 \right]^{1/2}, \quad (8.18)$$

missä R on ionin ja kohtioatomin minimietäisyys törmäyksen aikana. Se saadaan yhtälön $g(R) = 0$ ratkaisuna. Ioni ja atomi vuorovaikuttavat paripotentiaalin $V(r)$ välityksellä.

Törmäysparametri b on ionin tulevan nopeuden suunnan ja kohtioatomin kohtisuora etäisyys. Mitä pienempi b sitä rajumpi törmäys.

E_r on ionin ja atomin suhteellinen kineettinen energia (eli kineettinen energia MKP-koordinaatistossa), ja saadaan kaavasta

$$E_r = \frac{AE_1}{1+A}, \quad (8.19)$$

missä A on kohtioatomin ja ionin massasuhde

$$A = \frac{M_2}{M_1}. \quad (8.20)$$

Sirontakulmat laboratoriokoordinaatistossa saadaan MKP-kulmasta θ :

$$\tan \vartheta = \frac{Af \sin \theta}{1 + Af \cos \theta} , \quad \tan \phi = \frac{f \sin \theta}{1 - f \cos \theta} , \quad (8.21)$$

missä on otettu huomioon myös törmäyksessä mahdollinen epäelastinen energianmenetys Q :

$$f = \left(1 - \frac{Q}{E_r}\right)^{1/2} . \quad (8.22)$$

Hiukkasten asymptoottisten ratojen määrittämiseksi on myös laskettava suureet x_1 ja x_2 :

$$x_1 = \frac{(1 + f)\tau + (fA - 1)b \tan \frac{\theta}{2}}{f(1 + A)} , \quad (8.23)$$

$$x_2 = b \tan \frac{\theta}{2} - x_1 . \quad (8.24)$$

Tässä τ on ns. aikaintegraali

$$\tau = (R^2 - b^2)^{1/2} - \int_R^\infty \left\{ [g(r)]^{-1} - \left[1 - \left(\frac{b}{r}\right)^2\right]^{-1/2} \right\} dr . \quad (8.25)$$

Kohtioatomille siirtynyt energia T MKP-sirontakulman funktiona on

$$T = \left[f \sin^2\left(\frac{\theta}{2}\right) + \frac{(1 - f)^2}{4} \right] T_m , \quad T_m = \frac{4E_r}{1 + A} . \quad (8.26)$$

Jos epäelastinen energianmenetys jätetään huomiotta, saadaan

$$T = \sin^2\left(\frac{\theta}{2}\right) T_m . \quad (8.27)$$

Kun tiedämme sirontakulman energian E_1 ja törmäysparametrin b funktiona $[\theta(E_1, b)]$ voimme periaattessa laskea nukleaarisen jarrutuskyvyn S_n :

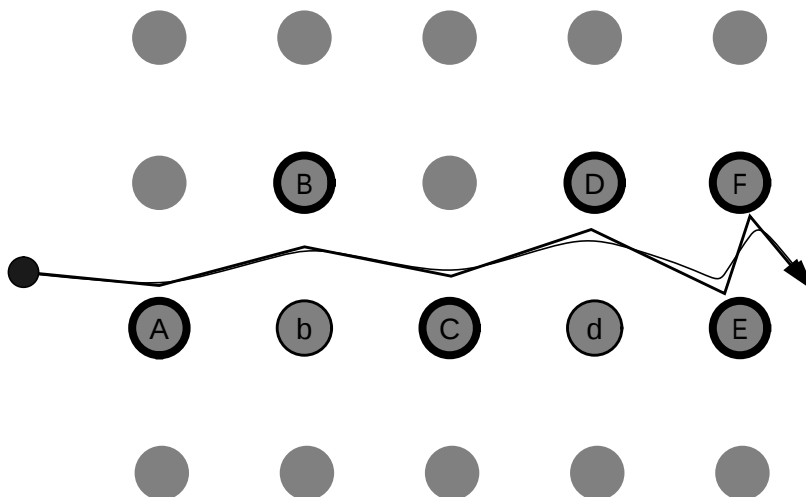
$$S(E) = \int T n d\sigma = n \int_0^\infty T 2\pi b db = 8n\pi \frac{M_1 M_2}{(M_1 + M_2)^2} \int_0^\infty \sin^2\left(\frac{\theta}{2}\right) b db , \quad (8.28)$$

missä n on aineen atomitiheys.

Repulsiivinen potentiaali $V(r)$ voidaan laskea approksimatiivisilla ns. vapaaelektronikaasumalleilla (free electron gas, FEG) tai käyttäen ab initio menetelmiä (tiheysfunktionaaliteoria).

Nukleaarista jarruuntumista on jonkin verran pyritty mallintamaan analyttisesti perustuen lähinnä Boltzmannin yhtälöön. Näissä malleissa joudutaan kuitenkin tekemään melko paljon yksinkertaistuksia ja esimerkiksi rajapintojen kuvaaminen on hankalaa. Tästä on ajauduttu mallintamaan ilmiötä tietokonesimulaatioilla.

Yleensä simulaatiokoodit perustuvat binääritörmäysapproksimaatioon (BCA, binary collision approximation). Siinä ionin radan oletetaan koostuvan suoraviivaisista pätkistä, joiden välillä se siroaa atomista. Ioni vuorovaikuttaa vain yhden atomin kanssa kerrallaan.

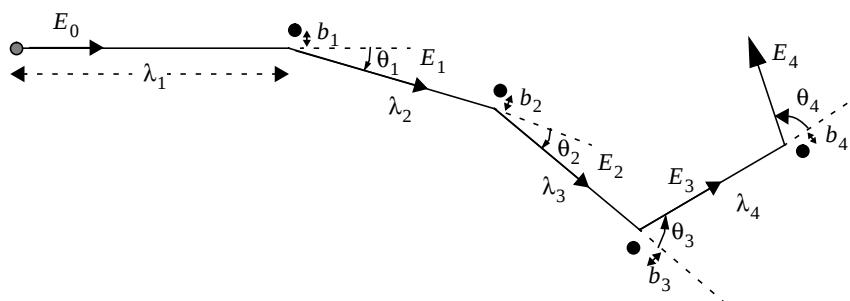


Kuva 8.7: Binääritörmäysapproksimaatio.

Tarpeeksi pienillä energioilla BCA ei enää päde, vaan ioni vuorovaikuttaa yhtä aikaa monen atomin kanssa.

Ionin etenemisen simulointi koostuu siis ionin ja väliaineen vuorovaikutustapahtumien gene-roinnista ja ionin kuljettamisesta näiden välillä.

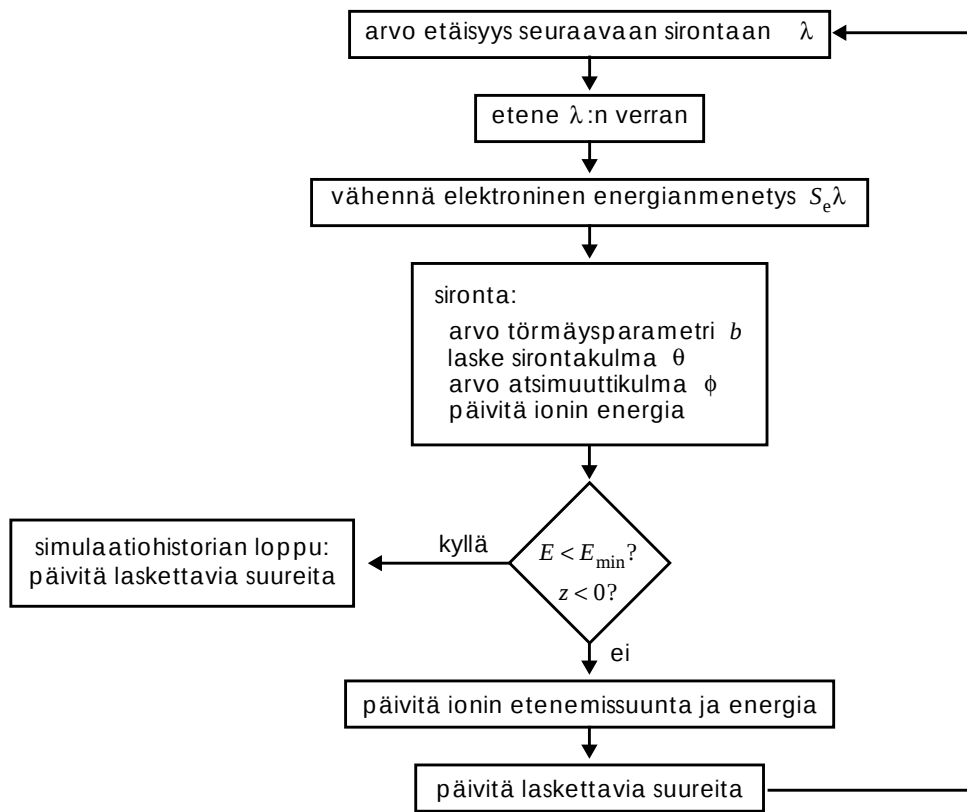
Vuorovaikutustapahtumat ovat tässä tapauksessa elastisia sirontoja väliaineen atomeista. Vuorovaikutus elektronisysteemin kanssa otetaan huomioon kitkanomaisena jatkuvana energianmenetyksenä elastisten sirontojen välillä.



Kuva 8.8: Binääritörmäyssimulaation periaate.

Tietyllä energialla E_0 aineeseen ammutun ionin kulkua simuloidaan kuvan 8.9 mukaisella

algoritmilla.



Kuva 8.9: Binääritörmäyssimulaation algoritmi.

Ionin sirontakulman θ pystymme laskemaan, kun tiedämme törmäysparametrin b [yhtälö (8.17)], ja tästä edelleen törmäyksessä väliaineen atomille jäävän energian [yhtälö (8.27)]. Lisäksi elektroninen jarrutuskyky on tiedossa.

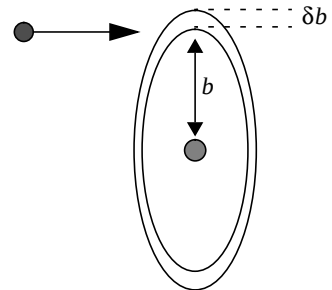
Jäljelle jää sirontojen välimatka λ , törmäysparametri b sekä sironnan atsimuuttikulma ϕ .

Usein väliaine oletetaan amorfiseksi, jolloin törmäykset ovat korreloimattomia. Lisäksi otetaan huomioon vain törmäykset, jotka sirottavat ionia enemmän kuin tietyn minimikulman $\theta_{\min}$ (tyypillinen arvo $\theta_{\min} = 0.5^\circ$). Eli sirontoja, joiden törmäysparametri $b > b_{\max}$ [$b_{\max} = b(\theta_{\min})$], ei oteta huomioon. Parametri $b_{\max}$ riippuu ionin energiasta.

Törmäysparametri arvotaan satunnaisesti väliltä $[0, b(\theta_{\min})]$. Geometrisella tarkastelulla on helppo huomata, että todennäköisyys, että törmäysparametri on välillä $[b, b + \delta b]$ on

$$P \propto b \delta b .$$

(8.29) Kuva 8.10: Törmäysparametrin b arvonta.



Tällöin (muistele kappaletta 6.6.2) törmäysparametri saadaan seuraavasti:

$$b = b(\theta_{\min})\sqrt{\xi}, \quad (8.30)$$

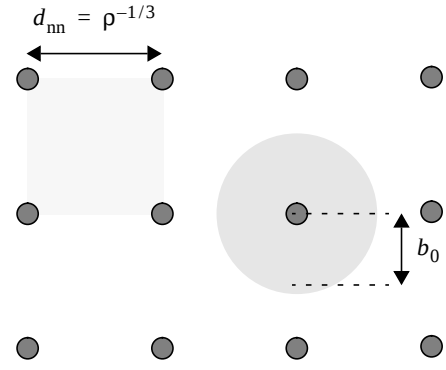
missä ξ on tasaisesti välille $[0, 1]$ jakautunut satunnaisluku.

Tarpeeksi pienillä energioilla maksimitörmäysparametri kasvaa suuremmaksi kuin aineen atomien *keskimääräinen* lähinaapurien etäisyyden puolikas $d_{nn}/2$.

Tämä ei ole realistinen tilanne, sillä simulaatiossa tulee ottaa huomioon aina mahdollisista törmäyksistä rajuimmat.

Tämä korjataan siten, että maksimitörmäysparametria ei aseteta suuremmaksi kuin arvo b_0 , joka vastaa yhden atomin *poikkipinta-alaa* aineessa, jonka atomitiheys on ρ :

$$\begin{aligned} \pi b_0^2 &= (\rho^{-1/3})^2 \\ \Rightarrow b_0 &= \pi^{-1/2} \rho^{-1/3} \end{aligned} \quad (8.31)$$



Kuva 8.11: Atomin poikkipinta-ala.

Eli törmäysparametrin arvonta on lopulta seuraava:

$$b_{\max} = \min(b_0, b(\theta_{\min})) ; b = b_{\max} \sqrt{\xi} . \quad (8.32)$$

Törmäysten välimatka arvonta perustuu siihen, että todennäköisyys sille, että tilavuudessa V on k atomia on Poisson-jakautunut:

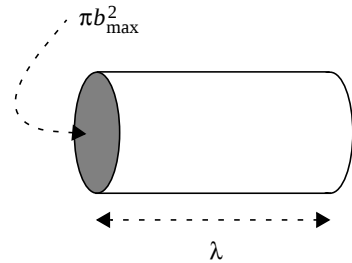
$$P(k) = \frac{(\rho V)^k}{k!} e^{-\rho V} . \quad (8.33)$$

Tästä saamme todennäköisyyden, ettei tilavuudessa ole yhtään atomia

$$P(0) = e^{-\rho V} . \quad (8.34)$$

Todennäköisyys ettei atomi lainkaan siroa kulkiessaan matkan λ on todennäköisyys, ettei tilavuudessa $\pi b_{\max}^2 \lambda$ ole yhtään atomia:

$$P(0) = e^{-\rho \pi b_{\max}^2 \lambda} . \quad (8.35)$$



Kuva 8.12: Keskimääräinen vapaa matka.

Tämä on siis ionin vapaan matkan λ todennäköisyysjakauma. Sen arpominen tapahtuu seuraavasti:

$$\lambda = -\bar{\lambda} \ln \xi , \quad (8.36)$$

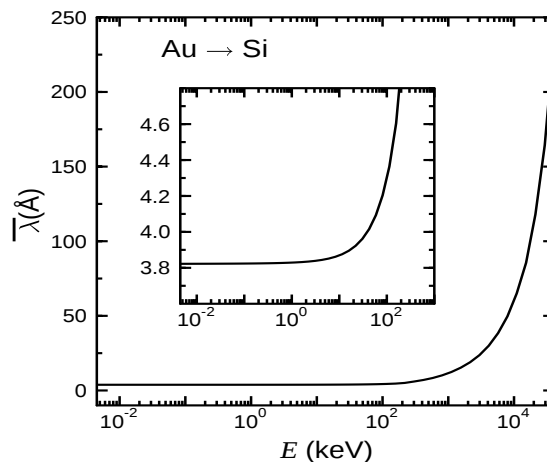
missä ξ on kuten edellä ja $\bar{\lambda}$ on ionin keskimääräinen vapaa matka

$$\bar{\lambda} = \frac{1}{\pi \rho b_{\max}^2} . \quad (8.37)$$

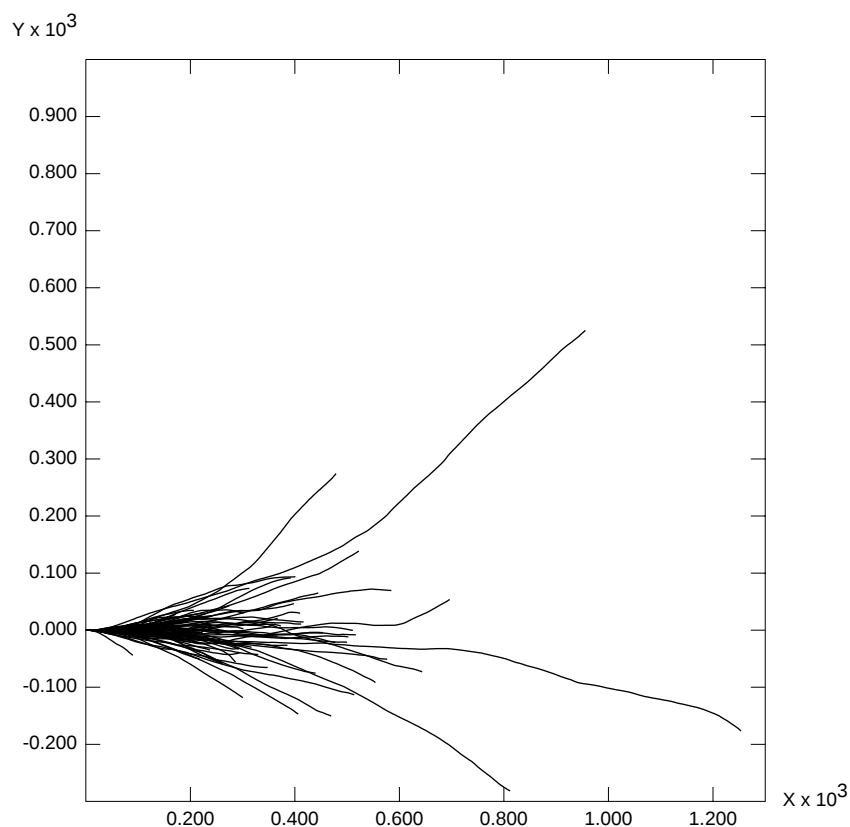
Kuvassa 8.13 esimerkkinä kultaionin keskimääräinen vapaa matka piissä.

Sirontaintegraali (8.17) lasketaan numeerisesti joko simulaation alussa kaksiulotteiseen taulukkoon $[\theta(E, b)]$ tai simulaation aikana käyttäen approksimatiivisia kaavoja. Minimisirontakulmaa vastaavat törmäysparametrit lasketaan yleensä simulaation alussa taulukkoon.

Simulaation raakatuloksena saadaan siis ionien ratoja. Alla esimerkkinä 100 keV:n kultaionien ratoja piissä [pituusyksikkö Å].

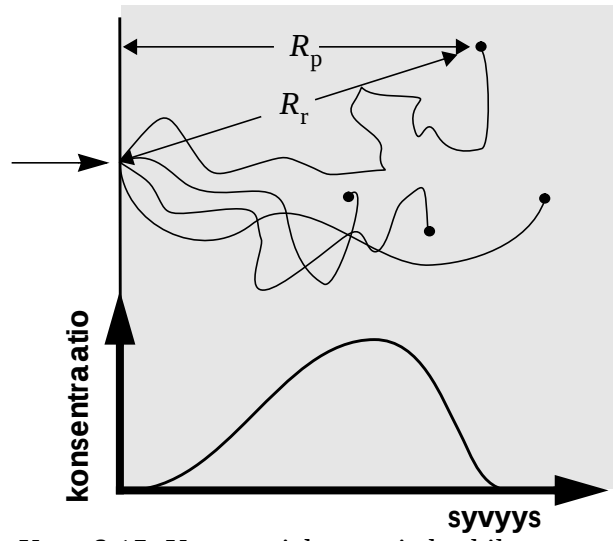


Kuva 8.13: Kultaionin keskimääräinen vapaa matka piissä.



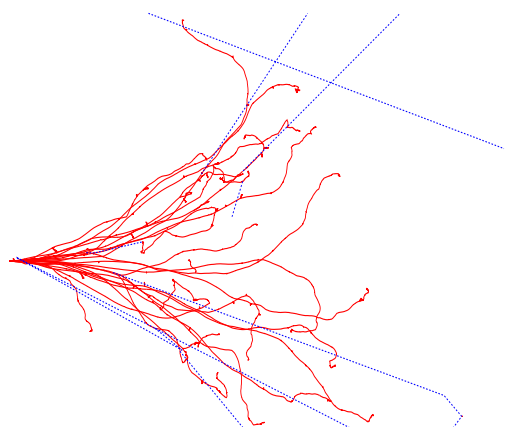
Kuva 8.14: 100 keV:n kultaionien ratoja piissä.

Ioni-istutuksessa ensisijainen mielenkiinnon kohde on se, miten syvälle aineeseen ammuttavat ionit pysähtyvät eli kantamajakauma.

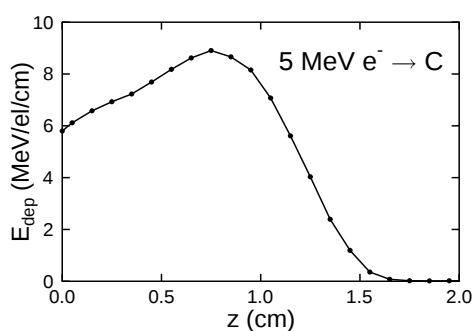


Kuva 8.15: Kantamajakauma ja keskikantamat.

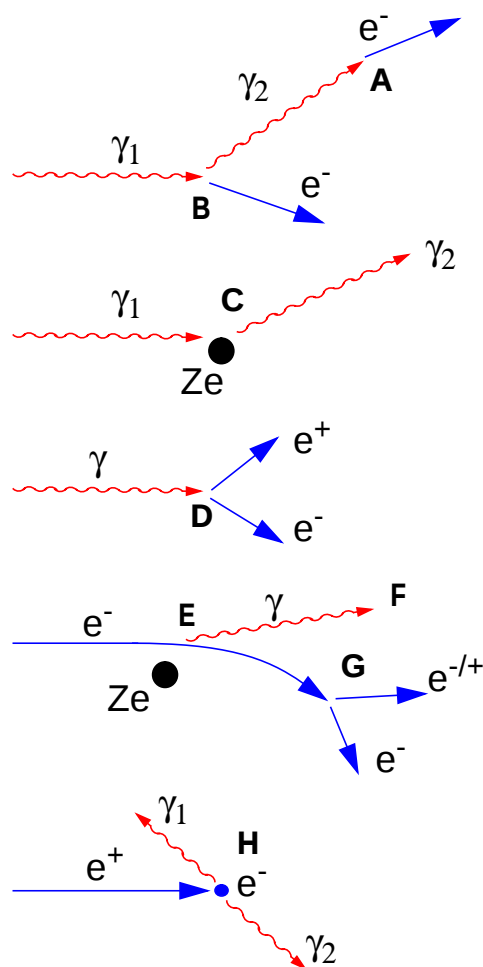
Elektroni- ja gammasäteilyn kulkeutumisen simuloinnissa tilanne on edellistä monimutkaisempi, koska vuorovaikutusmekanismeja (ks. kuva 8.16) on useampia ja hiukkasia voi syntyä ja hävitä vuorovaikutusprosesseissa. Eri mekanismien todennäköisyydet eli vaikutusalat tunnetaan kuitenkin kohtalaisen hyvin, joten MC-menetelmällä saadaan varsin hyviä tuloksia.



Kuva 8.17: 5 MeV:n elektronien ratoja grafiitissa. Simulointi on tehty EGS4-ohjelmistolla.



Kuva 8.18: 5 MeV:n elektronisuihkun grafiittiin jättämä energia suihkun tunkeutumisvyyden funktiona.

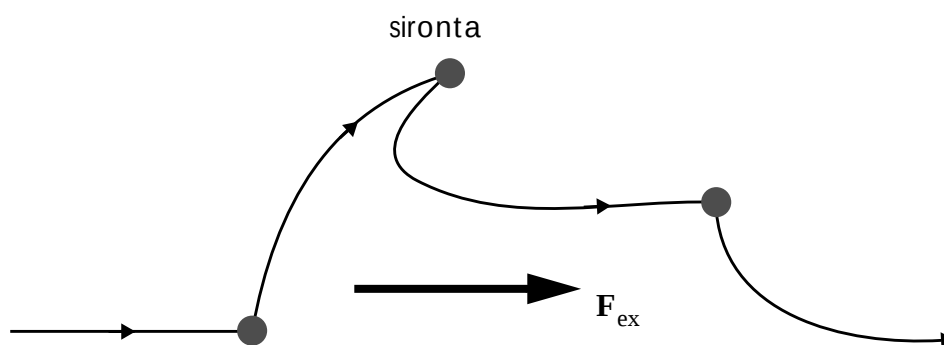


Kuva 8.16: Gamma- ja elektronisäteilyn vuorovaikutusmekanismit väliaineen kanssa. A: valosähköinen ilmiö, B: Compton-sironta, C: Rayleigh-sironta, D: parinmuodostus, E: elastinen sironta (ytimestä), F: jarrutussäteily, G: epäelastinen sironta (elektronista), H: positronin annihilaatio.

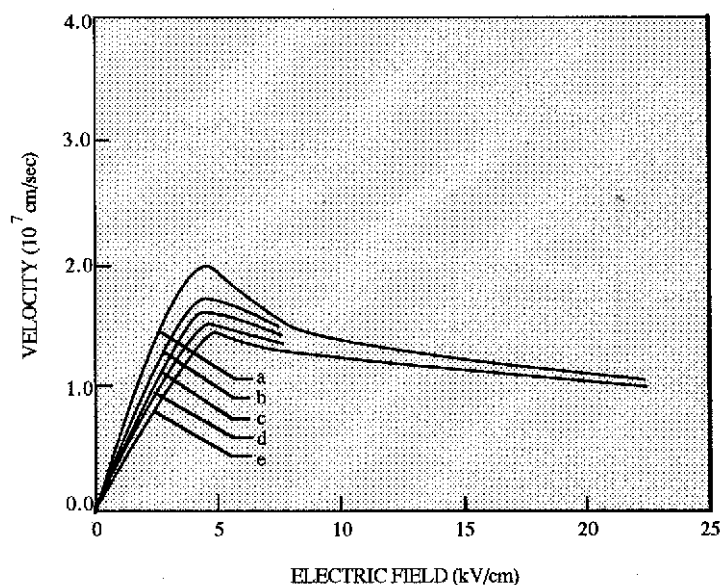
8.3 Varaustenkuljettajien kulkeutuminen puolijohdemateriaaleissa

Elektronien ja aukkojen kulkeutuminen ulkoisen sähkökentän ajamana erilaisissa puolijohdemateriaaleissa on komponenttien nopeuden kannalta tärkeä tutkimuskohde. Tässäkin tapauksessa Monte Carlo -menetelmä on yksi käyttökelpoisimmista menetelmistä.

Simuloinnin periaatteet ovat samat kuin edellisessä kappaleessa: elektroni (tai aukko) vuorovaikuttaa väliaineen kanssa diskreeteissä siirtoissa, joiden välisen ajan se kulkee vapaasti ulkoisten voimien vaikutuksen alaisena.



Kuva 8.19: Varauksenkuljettajan kulkeutumisen MC-simulointi.

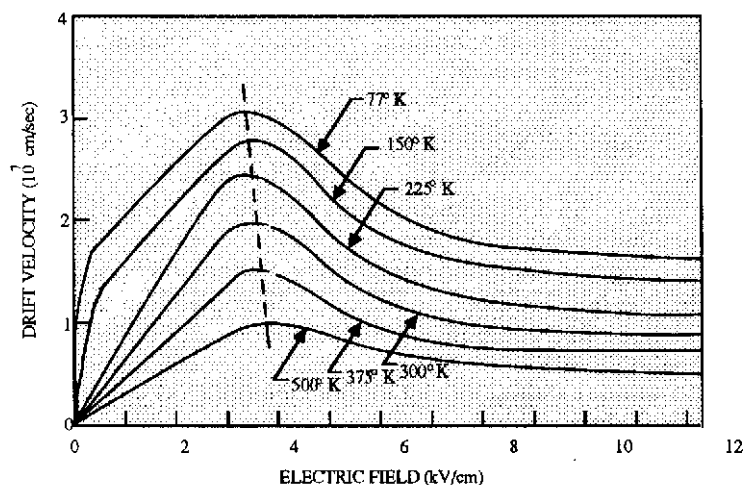


Kuva 8.20: Elektronien nopeus galliumarsenidissa ulkoisen sähkökentän funktiona MC-menetelmällä laskettuna¹. Eri käyrien epäpuhtauksien konsentraatiot ovat (a) $N_D = 0$, (b)

$N_D = 1.0 \times 10^{17} \text{ cm}^{-3}$, (c) $N_D = 2.0 \times 10^{17} \text{ cm}^{-3}$, (d) $N_D = 4.0 \times 10^{17} \text{ cm}^{-3}$, (e)

$N_D = 8.0 \times 10^{17} \text{ cm}^{-3}$.

1. J. Singh: *Physics of semiconductors and their heterostructures*, McGraw-Hill, 1993, kappale 13.3.



Kuva 8.21: Elektronien nopeus galliumarsenidissa ulkoisen sähkökentän funktiona eri läpötiloissa MC-menetelmällä laskettuna¹.

8.4 Kineettinen Monte Carlo

8.4.1 Yleistä

Erilaisia kuljetus- ja kasvuilmiöitä voidaan mallintaa kineettisellä Monte Carlolla (KMC). Tärkeitä sovellutuksia ovat mm. diffuusio kiinteässä aineessa, kiteen kasvu, säteilyvaurioiden kulkeutuminen.

Systeemiä kuvataan masteryhtälöllä

$$\frac{\partial P(C, t)}{\partial t} = \sum_{C'} [W(C' \rightarrow C)P(C', t) - W(C \rightarrow C')P(C, t)] , \quad (8.38)$$

missä C indeksoi systeemin tilaa.

Kuten staattisessa MC:ssä (SMC), KMC:ssäkin masteryhtälö ratkaistaan generoimalla jono systeemin tiloja. SMC:ssä ainoa kriteeri generoiduille jonoille oli, että mikroskooppinen reversiibeliys pätee. Tämä takasi konvergenssin tasapainojakaumaa kohti. KMC:ssä tämän lisäksi siirtymien tilasta toiseen tulee vastata todellisia systeemin siirtymiä.

KMC perustuu siihen, että kuvaamme systeemiä sellaisella tasolla, että voimme erottaa tietyn joukon diskreettejä tapahtumia $E = \{e_1, e_2, \dots, e_N\}$, joissa systeemi siirtyy tilasta toiseen. Lisäksi aikaskaala on sellainen, että mitkään tapahtumat eivät tapahdu samanaikaisesti. Esimerkiksi kiteen kasvussa kaasufaasista tapahtumia ovat: adsorptio, desorptio ja atomin hyppy pinnalla erilaisissa geometrioissa.

Olkoon tapahtuman a todennäköisyys aikayksikköä kohti (rate) R_a . Yleensä todennäköisyydet $\{R_a\}_{a=1}^N$ ja N riippuvat konfiguraatiosta C .

Määritellään kokonaistodennäköisyys siirtymälle (*total rate*) tietyssä konfiguraatiossa

$$Q = Q(C) = \sum_{a=1}^N R_a . \quad (8.39)$$

Muodollisesti siirtymätodennäköisyys W voidaan kirjoittaa muodossa

$$W(C \rightarrow C') = \sum_{a=1}^N R_a V^a(C \rightarrow C') , \quad (8.40)$$

missä V^a on tavallaan stokastinen yritematriisi. Se kertoo, onko siirtymä $C \rightarrow C'$ mahdollinen tapahtuman a kautta.

Simulaatiossa tapahtuma a pitäisi valita todennäköisyydellä $R_a/Q(C)$ eli verrannollisesti todennäköisyyteen aikayksikkö kohti.

Suoraviivainen algoritmi tämän toteuttamiseksi olisi seuraava. Simulaation alussa etsimme tapahtuman, jolla on suurin mahdollinen todennäköisyys aikayksikköä kohti $R_{\max}$. Laskemme kaikille mahdollisille tapahtumille suhteelliset todennäköisyydet

$$P_a = \frac{R_a}{R_{\max}} , \quad (8.41)$$

ja luomme listan alkukonfiguraatiossa mahdollisista tapahtumista.

Algoritmin sisin silmukka askeleella k voisi olla seuraavanlainen:

- i. Valitse satunnaisesti yksi konfiguraatiossa C_k mahdollinen tapahtuma. olkoon sen suhteellinen todennäköisyys P_e .
- ii. Generoi tasaisesti jakautunut satunnaisluku $\xi \in [0, 1)$.
- iii. Jos $\xi \leq P_e$, päivitä systeemin tila $C_{k+1} = C'$ ja mahdollisten tapahtumien lista. Muuten pysy entisessä tilassa.

Tämä algoritmi ei ole kovin käyttökelpoinen, koska kaikkia tapahtumia yritetään samalla todennäköisyydellä. Usein systeemissä on tapahtumia, joiden todennäköisyydet aikayksikköä kohti eroavat jopa kertaluvuilla. Tällöin epätodennäköisiä tapahtumia yritetään ja hylätään usein. Esimerkiksi diffuusiosimulaatiossa atomeja, joilla on suuri tai pieni koordinaatio¹ yritetään siirtää yhtä usein, mutta atomi, jolla on pieni koordinaatio liikkuu paljon enemmän.

Usein mallista voidaan unohtaa tapahtumat, joiden todennäköisyys on kovin pieni. Aina tämä ei kuitenkaan ole mahdollista. Esimerkkinä voisi mainita nukleaation kiteen kasvun simulaatiossa. Usein adatomien² desorption todennäköisyys on paljon suurempi kuin useamman adato-

1. Todennäköisyyksien suhde on $e^{-\beta\Delta E}$, missä ΔE on atomien sidosenergioiden ero.
 2. Materiaalin pintaan tarttunut (vieras)atomi.

min liittyminen klusteriksi. Kuitenkin, kun klusteri on saavuttanut tietyn kriittisen koon, se alkaa kasvaa.

Nopeampi algoritmi, jossa vältetään tuloksettomat yiritykset on ns. BKL-algoritmi¹. Siinä tapahtuma alunperin valitaan oikealla todennäköisyydellä ja toteutetaan.

- i. Generoi tasaisesti jakautunut satunnaisluku $\xi \in [0, Q(C_k))$.
- ii. Valitse tätä vastaava tapahtuma: valitse ensimmäinen indeksi s , jolle pätee

$$\sum_{a=1}^s R_a(C) \geq \xi.$$

- iii. Etene uuteen konfiguraatioon C_{k+1} toteuttamalla tapahtuma s .
- iv. Päivitä niitä todennäköisyyksiä R_a , jotka ovat muuttuneet tapahtuman s seurauksena. Päivitä Q ja muut tarvittavat tietorakenteet.

Tarkastellaan algoritmin CPU-ajankäytön riippuvuutta systeemin koosta N . Askeleiden i. ja iii. ajankäyttö ei riipu N :stä. Askel ii. vaatii ajan $O(N)$. Useimmiten askeleen iv. suoritus-aika ei riipu systeemin koosta, vaikkakin se vaatii huolellista ohjelmointia.

Vielä nopeammaksi algoritmin saadaan, jos tapahtumat ryhmitellään². Niputetaan tapahtumat n :ään ryhmään ja indeksoidaan ryhmiä α :lla: $\alpha = 1 \dots n$. Ryhmittely voidaan tehdä ottamalla jokaiseen ryhmään sama määrä tapahtumia (tehokkainta) tai luokittelemalla fysikaalisesti samantyyppiset *prosessit* samoihin ryhmiin (adatomin diffuusio tietyn energiavallin yli, tietyn sidosennergiaisen adatomien desorptio, jne.).

Oletetaan, että ryhmittely on tehty jälkimmäisellä tavalla siten, että kaikilla ryhmän tapahtumilla on sama todennäköisyys ρ_α .

Nyt jokaisessa konfiguraatiossa C on mahdollisia prosesseja, jotka voidaan realisoida monella eri tavalla. Prosessi α voidaan realisoida $n_\alpha(C)$ eri tavalla. Suuretta $n_\alpha(C)$ kutsutaan multiplisiteetiksi. Esimerkiksi systeemissä voi olla $n_{\text{adatom}}(C)$ diffundoituvaa adatomia, joilla on samanlainen ympäristö.

Joka prosessille määritellään osittaistodennäköisyydet aikayksikköä kohti (*partial rate*)

$$q_\alpha(C) = n_\alpha(C) \rho_\alpha \quad (8.42)$$

ja suhteelliset osittaistodennäköisyydet

$$p_\alpha(C) = \frac{q_\alpha(C)}{Q(C)}. \quad (8.43)$$

Konfiguraation C kokonaissiirtymätodennäköisyys aikayksikköä kohti on nyt

1. A. B. Bortz, M. H. Kalos, J. L. Lebowitz, *J. Comput. Phys.* **17** (1975) 10.
 2. P. A. Maksym, *Semicond. Sci. Tech.* **3** (1988) 594.

$$Q(C) = \sum_{\alpha=1}^n n_{\alpha}(C) p_{\alpha} . \quad (8.44)$$

Algoritmi näyttää seuraavalta:

- i. Generoi tasaisesti jakautunut satunnaisluku $\xi_1 \in [0, Q(C_k))$.
- ii. Valitse toteutettava prosessi: valitse ensimmäinen indeksi σ , jolle pätee
$$\sum_{\alpha=1}^{\sigma} q_{\alpha}(C) \geq \xi_1 .$$
- iii. Valitse prosessin σ toteutustapa. Esimerkiksi käytössä voi olla lista atomeista $(1 \dots n_{\sigma}(C))$, joille prosessi voidaan toteuttaa. Generoidaan tasaisesti jakautunut satunnaisluku (kokonaisluku) väliltä $[1, n_{\sigma}(C))$ ja valitaan luvun osoittama atomi.
- iv. Toteutetaan prosessi (siirretään atomia).
- v. Päivitetään multiplisiteetteja n_{α} ja todennäköisyyksiä q_{α} ja Q .

Arvioidaan algoritmin ajankäyttöä. Jos oletamme, että multiplisiteetit ovat samansuuruisia $n_{\alpha} \approx N/n$. Tapahtuman etsintä on kaksivaiheinen. Ryhmän etsintä vie aikaa $O(N/n)$ ja etsintä ryhmän sisällä $O(n)$. Optimaalinen tilanne on, jos ryhmien määrä $n \propto N^{1/2}$. Tällöin laskenta-ajan riippuvuus systeemin koosta on $O(N^{1/2})$. Askel v. voi edelleenkin viedä aikaa enimmillään $O(N)$.

Algoritmin tehokkuutta voidaan parantaa edelleen tekemällä ryhmittely useammalla tasolla¹. Periaatteessa optimaalinen tilanne saavutetaan, kun alimman tason ryhmissä on jokaisessa vain kaksi alkiota. Tällöin aikaa kuluu $O(\ln_2 N)$.

Ylläolevissa algoritmeissa ei ollut aikaa eksplisiittisesti mukana. Joskus kuitenkin haluamme kiinnittää systeemin kehityksen aikaskaalan.

Emme voi suoraan olettaa, että MC-askel (MCS) kertoo ajan: $t \propto \text{MCS}$. Jos voimme olettaa, että vain yksi tapahtuma voi tapahtua kerrallaan (eli aikaskaala, jolla systeemiä tutkitaan on tarpeeksi pieni) ja että tapahtumat ovat Poisson-jakautuneita saamme laskuihimme aikaskaalan.

Todennäköisyys, että aikavälillä $[0, t]$ on p kappaletta tapahtumia on

$$P(p) = \frac{(Qt)^p}{p!} e^{-Qt} , \quad (8.45)$$

missä edelleen $Q = Q(C) = \sum_{a=1}^N R_a$. Todennäköisyys, että ko. välillä ei ole yhtään tapahtumaa on siten

$$P(0) = e^{-Qt} . \quad (8.46)$$

1. J. L. Blue, I. Beichl, F. Sullivan, *Phys. Rev. E* **51** (1995) R867.

Tapahtumien välisen ajan τ (odotusaika, waiting time) (oikein normitettu) todennäköisyysjakauma on siten

$$P(\tau) = Qe^{-Q\tau} , \quad (8.47)$$

ja sen odotusarvo on

$$\langle \tau \rangle = \frac{1}{Q} . \quad (8.48)$$

Simulaatioissa aika saadaan siis generoimalla ylimääräinen satunnaisluku ξ_3 ja laskemalla siitä aikaväli seuraavaan tapahtumaan:

$$\Delta t_k = \frac{\ln \xi_3}{Q(C_k)} . \quad (8.49)$$

Erilaiset aikakeskiarvot lasketaan simulaatioissa seuraavasti:

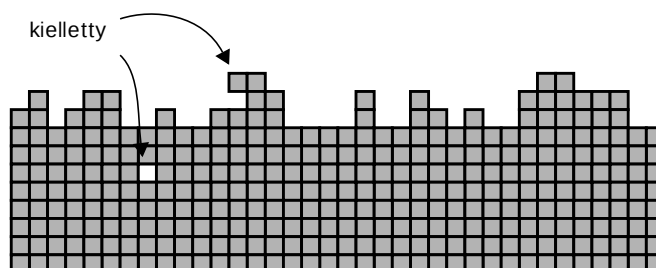
$$\langle A \rangle = \frac{1}{t} \sum_{k=1}^M \Delta t_k A(C_k) , \quad (8.50)$$

missä kokonaisaika on

$$t = \sum_{k=1}^M \Delta t_k . \quad (8.51)$$

8.4.2 Pinnan kasvu ja pintadiffuusio¹

Pinnan kasvun simuloinnissa käytetään usein ns. solid-on-solid-mallia (SOS-malli). Siinä pinta voidaan kuvata funktiolla $h(x, y)$, joka kertoo pinnan korkeuden.

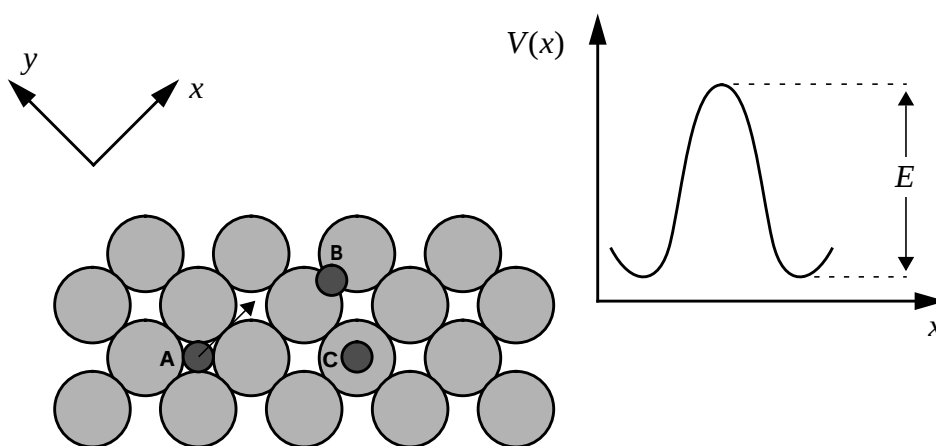


Kuva 8.22: Pinnan kasvun solid-on-solid -malli.

Vakansseja ja ns. overhang-atomeja ei mallissa sallita. Hyvin usein mallitetaan kidettä yksinkertaisella kuutiollisella hilalla.

Yksinkertaisimmillaan mallissa kaasufaasin atomi yrittää säännöllisin väliajoin kiinnittyä satunnaiseen hilapaikkaan pinnalla. Tarttumistodennäköisyys riippuu hilapaikan naapureista; esimerkiksi syntyvien sidosten lukumäärästä. Mallin toteuttaminen MC-simulointina on varsin yksinkertainen tehtävä. Koska mallista puuttuu adatomien diffuusio pinnalla, se ei ole kovin realistinen.

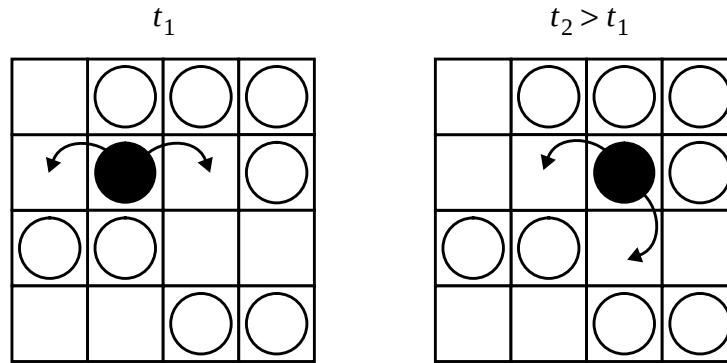
Tartuttuaan pintaan atomi² voi desorptoitua eli palata kaasufaasiin. Tämän tapahtuman todennäköisyys on yleensä pieni kemisorption tapauksessa, koska atomin sidosenergia pinnalla on kemiallisen sidosenergian suuruusluokkaa eli muutama eV. Atomi voi kuitenkin kulkeutua pinnalla termisen virityksen seurauksena: on kyse pintadiffusiosta.



Kuva 8.23: Adatomin mahdollisia paikkoja tilakeskisen kuution (100)-pinnalla. Potentiaalienergiavalli E on adatomin A kuvaan piirretyn hyppäyksen aktivaatioenergia.

1. Lisätietoa pintafysiikasta saat esimerkiksi kirjoista: M.Manninen, R.Nieminen, *Pintafysiikka*, A.Zangwill, *Physics at Surfaces*.
2. Adsorboitunut atomi eli adatomi.

Yksinkertaisin pintadiffuusion malli lienee hilakaasu, jossa pinta koostuu hilapaikoista, jotka ovat tai eivät ole diffundoituvan atomin miehittämiä.



Kuva 8.24: Pintadiffuusion hilakaasumalli.

Diffuusio koostuu adatomien hypäyksistä naapuripaikkoihinsa. Hyppyjä rajoittavana ehtona on se, että yhdessä hilapaikassa voi olla vain yksi atomi kerrallaan. Puhutaan korreloituneesta satunnaiskävelystä. Jos adatomien konsentraatio pinnalla on hyvin pieni, on todennäköisyys, että naapuripakka on miehitetty, pieni. Tällöin on kyseessä normaali satunnaiskävely, jolle pätee¹

$$\langle [\mathbf{r}(t) - \mathbf{r}(0)]^2 \rangle \propto t . \quad (8.52)$$

Diffuusioprosessia voidaan kuvata yhtälöllä

$$\frac{\partial}{\partial t} P(\mathbf{r}, t) = D \nabla^2 P(\mathbf{r}, t) , \quad (8.53)$$

missä $P(\mathbf{r}, t)$ on todennäköisyys, että adatomi on paikassa $\mathbf{r}$ ajanhetkellä t . Diffuusiovakio D kuvaa adatomien liikkuvuutta. Se voidaan määrittää esimerkiksi Einsteinin relaation avulla:

$$D = \lim_{t \rightarrow \infty} \frac{1}{2dt} \langle [\mathbf{r}(t) - \mathbf{r}(0)]^2 \rangle . \quad (8.54)$$

Hilakaasumallissa diffuusiovakio riippuu adatomien konsentraatiosta. On odotettavissa, että mitä suurempi tämä konsentraatio on, sitä vaikeampi atomien on tehdä hyppyjä ja diffuusiovakio on pieni.

Tämän systeemin simulointi MC-menetelmällä on suoraviivainen tehtävä. Koska systeemissä on vain yhdenlaisia siirtymiä, voimme tehdä rinnastuksen $t \propto n_{\text{MC}}$. Systeemi koostuu $L \times L$:n suuruisesta neliöhilasta, johon asetetaan haluttu määrä adatomeja, jotka yrittävät hyppyjä naapuripaikkoihinsa. Algoritmi on seuraavanlainen:

- i. Arvo satunnaisesti adatomien paikat pinnalla.
- ii. Arvo siirrettävän adatomien indeksi.
- iii. Arvo uusi hilapaikka adatomien naapuripaikoista (ylös, alas, vasemmalle, oikealle).
- iv. Jos uusi hilapaikka on tyhjä, siirrä adatomi sinne. Jos se on miehitetty, älä tee mitään.

1. Ks. kappale 3.3.

v. Päivitä haluttuja laskureita.

vi. Mene askeleeseen ii.

C-ohjelmana algoritmi näyttää seuraavalta:

```
/*
Käännös ja linkkaus: cc -o diffusion diffusion.c getseed.c lcg.c
*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define REAL double
#define INT long int
#define INFOOUT 20
#define I(x,y) (x + L*y)

long INT getseed();
REAL lcg(long int *seed);
REAL badlcg(long int *seed);

main (int argc, char **argv)
{
long int seed;
INT Natom,Ntime,Nsample,L,itime,i,j,k,n,step,next,Nodump,Nodist;
INT lsize,*lattice,x,y,xp,yp,dx,dy;
INT *atomx,*atomy,*atomx0,*atomy0,*atomx1,*atomy1;
INT sample,infoint;
INT left,right,up,down;
REAL *norm;
REAL *R2,r,d;
REAL trials,accepts;
FILE *fd;
int s,pp[5]={0,0,0,0,0};

if (argc!=8) {
    fprintf
(stderr,"Usage: %s L Natom Ntime seed Nodist Nodump Nsample\n",argv[0]);
    return (1);
}

L=atoi(++argv);
Natom=atoi(++argv);
Ntime=atoi(++argv);
seed=atol(++argv);
Nodist=atoi(++argv);
Nodump=atoi(++argv);
Nsample=atoi(++argv);

fprintf(stderr,"\nInput parameters\n");
fprintf(stderr,"  System size:      %d\n",L);
fprintf(stderr,"  Number of atoms:  %d\n",Natom);
fprintf(stderr,"  Number of time steps: %d\n",Ntime);
fprintf(stderr,"  RNG seed number:   %d\n",seed);
fprintf(stderr,"  R2 output interval: %d\n",Nodist);
fprintf(stderr,"  Atom dump interval: %d\n",Nodump);
fprintf(stderr,"  Number of samples: %d\n",Nsample);
fprintf(stderr,"\n");
if (seed<=0) {
    seed=getseed();
```

```

    fprintf(stderr,"Seed calculated from time %d.\n",seed);
}
lsize=L*L;
lattice=(INT *) malloc((size_t) (lsize*sizeof(INT)));
atomx=(INT *) malloc((size_t) (Natom*sizeof(INT)) );
atomy=(INT *) malloc((size_t) (Natom*sizeof(INT)) );
atomx0=(INT *) malloc((size_t) (Natom*sizeof(INT)) );
atomy0=(INT *) malloc((size_t) (Natom*sizeof(INT)) );
atomx1=(INT *) malloc((size_t) (Natom*sizeof(INT)) );
atomy1=(INT *) malloc((size_t) (Natom*sizeof(INT)) );
R2=(REAL *) malloc((size_t) (Ntime*sizeof(REAL)) );
norm=(REAL *) malloc((size_t) (Ntime*sizeof(REAL)) );
for (i=0;i<Ntime;i++) { R2[i]=0.0;norm[i]=0.0;};
infoint=Nsample/INFOOUT; if (infoint==0) infoint=1;
trials=accepts=0.0;

if (Nodump) fd=fopen("diffusion.dump","w");

for (sample=0;sample<Nsample;sample++) {

    if (sample%infoint==0) fprintf(stderr,"%d ",sample);

    /* set up the lattice */
    for (i=0;i<lsize;i++) lattice[i]=0;
    for (i=0;i<Natom;i++) {
        x=lcg(&seed)*L;
        y=lcg(&seed)*L;
        while (lattice[I(x,y)]>0) {
            x=lcg(&seed)*L;
            y=lcg(&seed)*L;
        }
        lattice[I(x,y)]=1;
        atomx[i]=atomx0[i]=atomx1[i]=x;
        atomy[i]=atomy0[i]=atomy1[i]=y;
    }

    /* main loop */
    for (itime=0;itime<Ntime;itime++) {

        for (k=0;k<Natom;k++) {

            trials+=1.0;
            n=lcg(&seed)*Natom;
            x=xp=atomx[n];
            y=yp=atomy[n];
            dx=dy=0;
            s=4.0*lcg(&seed); if (s>3) s=3;
            pp[s]++;
            switch (s) {
            case 0:
                x--; dx=-1; if (x<0) x=L-1;
                break;
            case 1:
                x++; dx=1; if (x>=L) x=0;
                break;
            case 2:
                y--; dy=-1; if (y<0) y=L-1;

```

```

        break;
    case 3:
        y++; dy=1; if (y>=L) y=0;
        break;
    }
    if (lattice[I(x,y)]==0) {
        accepts+=1.0;
        atomx[n]=x; atomy[n]=y;
        atomx1[n]+=dx; atomy1[n]+=dy;
        lattice[I(xp,yp)]=0;
        lattice[I(x,y)]=1;
    }
    if (itime%Nodist==0) {
        R2[itime]+=(pow((double)(atomx1[n]-atomx0[n]),(double)2)+
                    pow((double)(atomy1[n]-atomy0[n]),(double)2) );
        norm[itime]+=1.0;
    }
}

if (itime%Nodump==0 && sample==0)
    fprintf(fd,"%d %d %d\n",atomx1[0],atomy1[0],itime);
}

}

fprintf(stderr,"\nR2 normalization %g\n",norm[0]);
fprintf(stderr,"Acceptance ratio %g\n\n",accepts/trials);

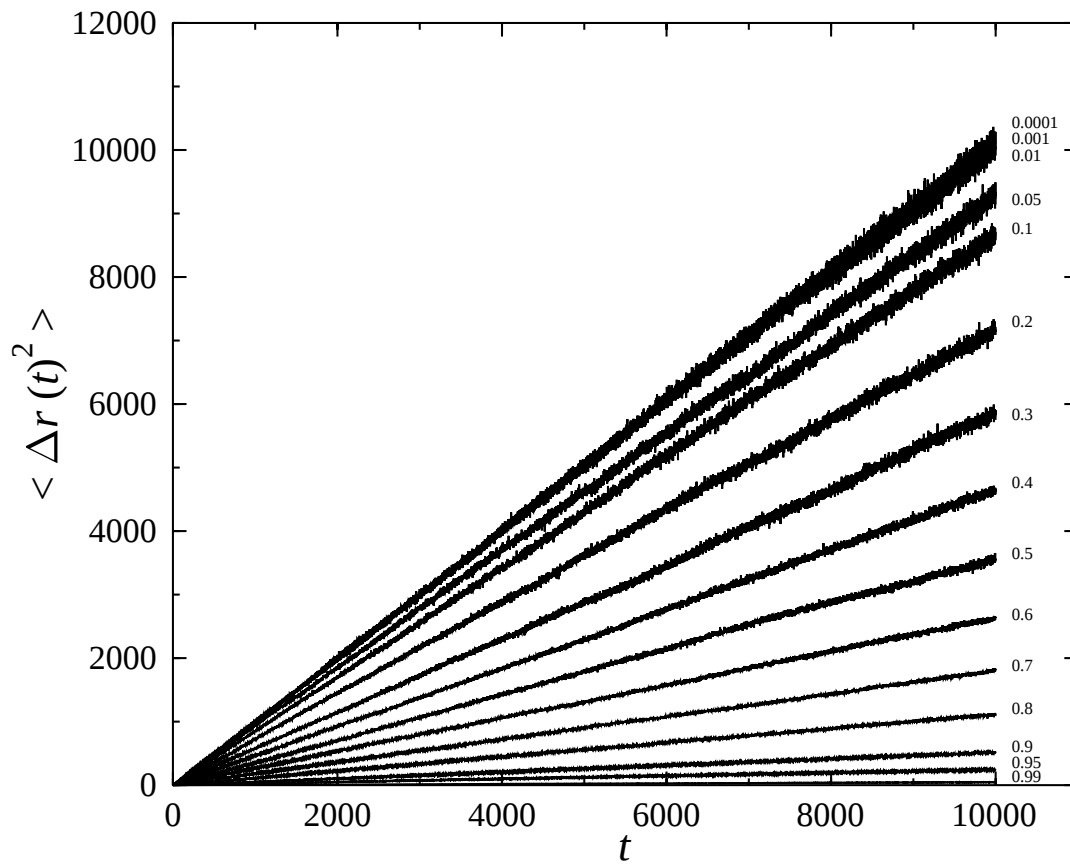
for (j=0;j<Ntime;j++)
    if (j%Nodist==0) fprintf(stdout,"%d %g\n",j,R2[j]/norm[j]);
fprintf(stderr,"pp: %d %d %d %d %d\n",pp[0],pp[1],pp[2],pp[3],pp[4]);

return(0);
}

```

Tällä ohjelmalla määritettiin diffuusiovakio D adatomien konsentraation¹ θ funktiona käyttäen kaavaa (8.54). Systemin koko oli 100×100 .

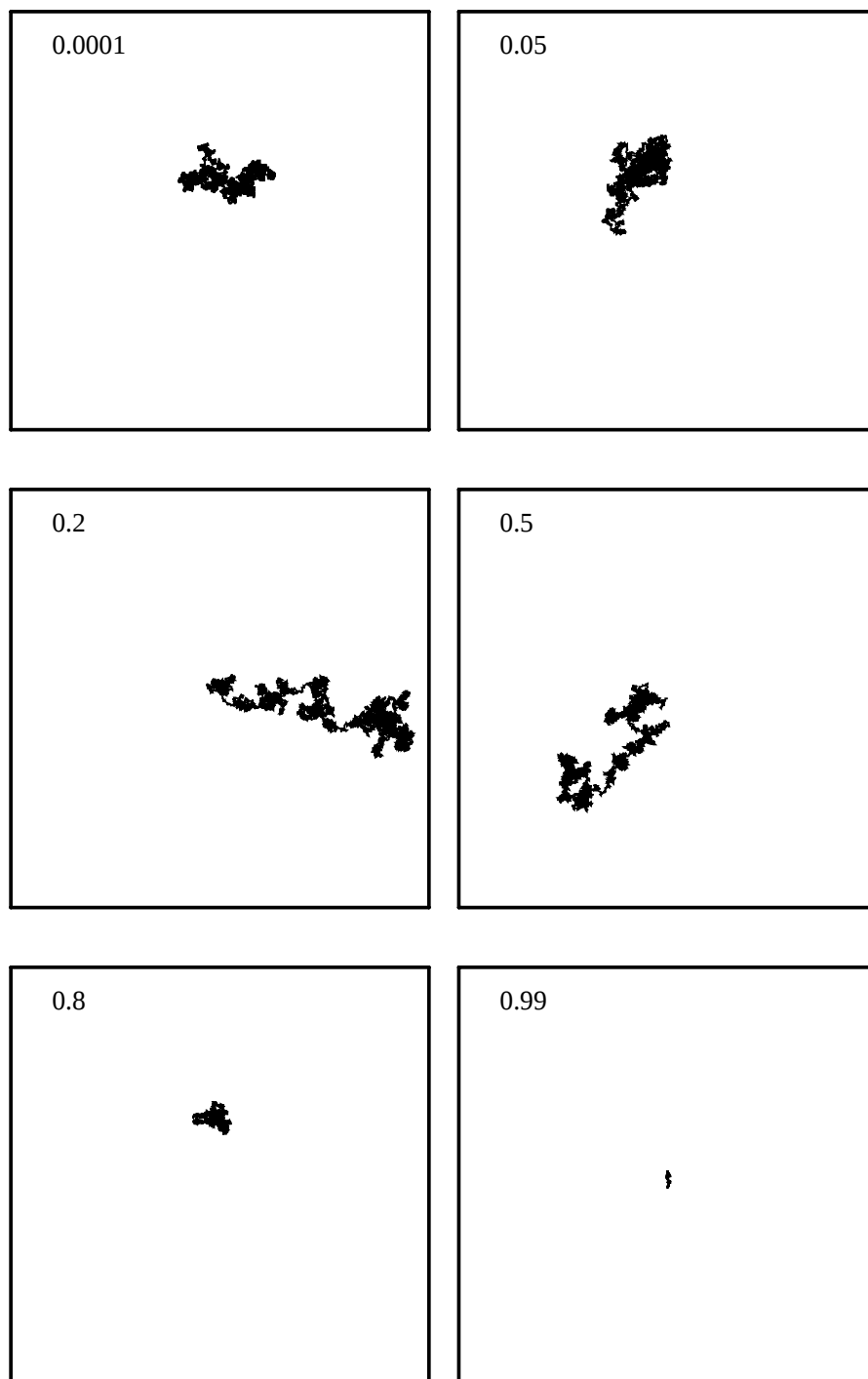
1. $\theta = \frac{n_{\text{adatom}}}{n_{\text{lattice sites}}}$



Kuva 8.25: Adatomien paikkojen neliöllinen keskiarvo ajan funktiona laskettuna erilaisille adatomikonsentraatioille.

Kuvassa 8.25 on esitetty adatomien paikkojen neliöllinen poikkeama ajan funktiona. Huomataan, että kaikki käyrät noudattavat relaatiota (8.52).

Kuvassa 8.26 on esitetty muutamia adatomien ratoja eri konsentraatiolle laskettuna.



Kuva 8.26: Tyypillisiä adatomien ratoja muutamalle eri konsentraation arvolle.

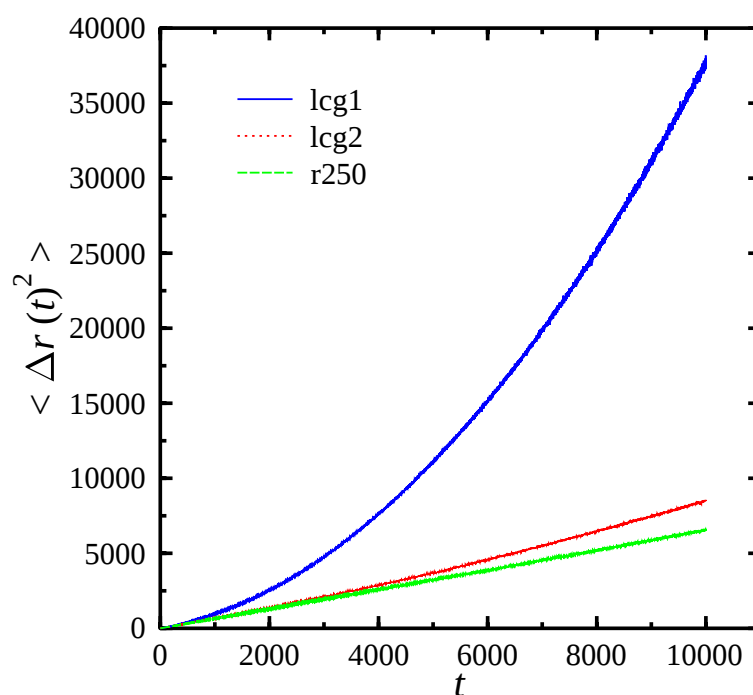
Näissä simulaatioissa käytettiin **R250**-astunnaislukugeneraattoria.¹ Jos laskut tehdään käyttäen allaesitettyjä LC-generaattoreita **lcg1** ja **lcg2**, saadaan väärä tuloksia.

1. Ks. kappale 6.6.1.

```
double lcg1(long int *ix)
{
double ret;
static long int k1,a=127773,b=16807,c=2836,d=2147483647;
double xx=4.656612875e-10;
k1 = *ix / a;
*ix = (*ix - k1 * a) * b - k1 * c;
if (*ix < 0) *ix += d;
ret = *ix * xx;
return ret ;
}
```

```
double lcg2(long int *seed)
{
static long int a=69069,c=1,m=4294967296;
static double rm=4294967296.0;

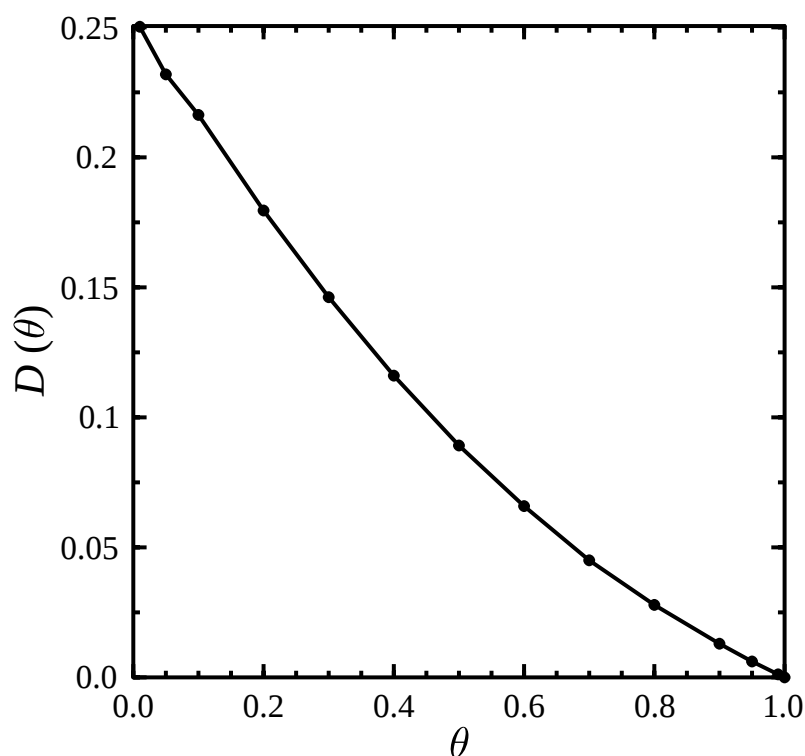
*seed=(*seed*a+c)%m;
return (double)*seed/rm;
}
```



Kuva 8.27: Adatomin paikan neliöllinen poikkema konsentraatiolla $\theta = 0.25$ laskettuna eri satunnaislukugeneraattoreilla.

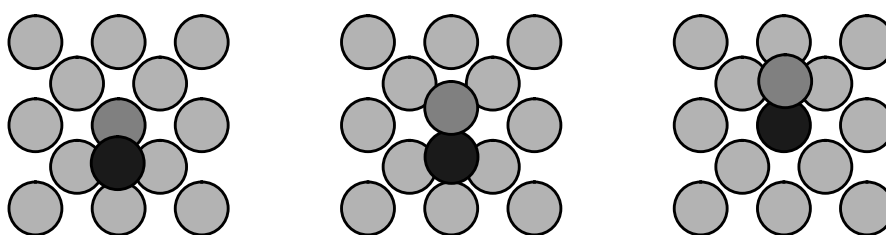
Huomaa, että LC-generaattoreiden tulokset eivät edes noudata kaavaa (8.52).

Diffuusiovakio laskettiin kaavalla (8.54) laskemalla keskiarvo 2000 viimeisestä simulaatioaskeleesta. Tulokset on esitetty kuvassa 8.28.



Kuva 8.28: Diffuusiovakio adatomien konsentraation funktiona.

Tällainen yksinkertainen hilakaasumalli voi kuvata pintadiffuusiota matalissa lämpötiloissa melko hyvinkin. On kuitenkin mahdollista, että adatomit esimerkiksi hyppävät kauemmas kuin naapuripaikkaan. Lisäksi diffuusiomekanismeja voi olla erilaisia, ja johonkin voi osallistua useampia atomeja. Yksi esimerkki on kuvassa 8.29. Lisäksi adatomien konsentraatio voi vaikuttaa substraatin pinnan rakenteeseen. Tällöin oletus täysin staattisesta pinnasta ei päde.



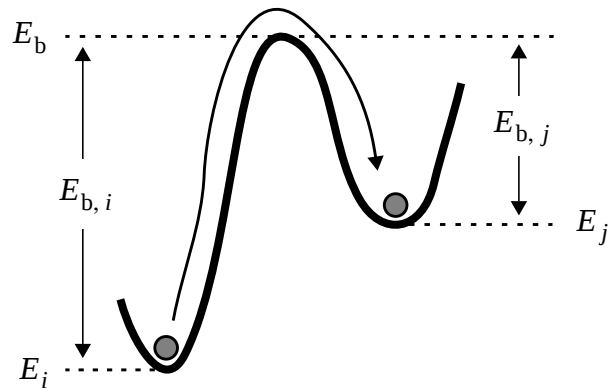
Kuva 8.29: Vaihtomekanismiin perustuva diffuusio tilakeskisen hilan (100)-pinnalla.

Huomaa, että edelläesitettyssä mallissa ei ollut ollenkaan mukana lämpötilaa.

Useimmiten diffuusiohyppyn todennäköisyys aikayksikköä kohti voidaan kuvata ns. siirtymätilateorian (transition state theory, TST) antamalla Arrhenius-tyyppisellä muodolla

$$R_{ij} = \nu e^{-E_{b,i}/k_B T}, \quad (8.55)$$

missä ν on oleellisesti diffundoituvan atomin keskimääräinen värähtelytaajuus ($\nu \sim 10^{13} \text{ s}^{-1}$) ja $E_{b,i}$ on diffuusiovallin korkeus katsottuna tilasta i . TST perustuu oletukseen, että kunhan atomi pääsee potentiaalienergiavallin huipulle (E_b), se todennäköisyydellä yksi päättyy tilaan j . Huomaa, että yhtälön (8.55) mukainen siirtymätodennäköisyys toteuttaa detaljibalanssin (4.18) olettaen, että tekijä ν on sama kaikille tiloille:



Kuva 8.30: Diffuusiovalli

$$\begin{aligned} R_{ij}P_i &= \nu e^{-E_{b,i}/k_B T} \frac{e^{-E_i/k_B T}}{Z}, \\ &= \frac{\nu}{Z} e^{-(E_b - E_i)/k_B T} e^{-E_i/k_B T} = \frac{\nu}{Z} e^{-E_b/k_B T} \end{aligned}$$

$$R_{ji}P_j = \nu e^{-E_{b,j}/k_B T} \frac{e^{-E_j/k_B T}}{Z} = \frac{\nu}{Z} e^{-E_b/k_B T}.$$

Huomaa, että siirtymätodennäköisyys ei riipu lopputilasta j !

Edelläesitettyä yksinkertaista mallia voidaan parantaa myös ottamalla adatomien vuorovaikutukset¹ mukaan. Tämä voidaan toteuttaa olettamalla adatomien i diffuusiovallin korkeudeksi²

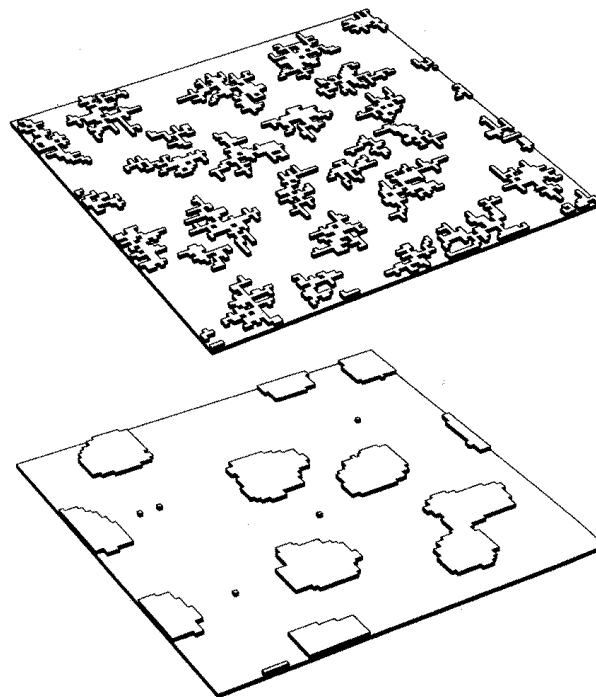
$$E_{b,i} = E_S + n_i E_N, \quad (8.56)$$

missä E_S on substraatin aiheuttama potentiaalienergia, n_i on atomin i naapureiden lukumäärä (pinnan suunnassa) ja E_N on vuorovaikutusenergia naapurin kanssa. Tällöin hyppytodennäköisyys atomille i on

$$R_i = \nu \exp[-(E_S + n_i E_N)/k_B T]. \quad (8.57)$$

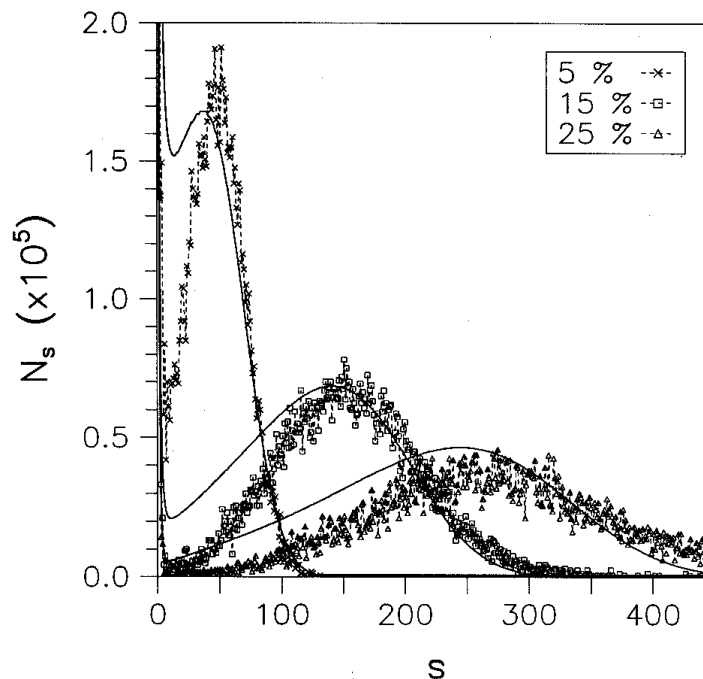
1. Muutenkin kuin ehtona, että samassa hilapaikassa voi olla vain yksi adatomi.

2. D.D.Vvedensky ym., *Fluctuations and Order*, toim. M.Millonas, Springer 1996.



Kuva 8.31: Pinnan kasvun MC-simulaatio. Adatomikonsentraatio on 0.2, depositionopeus 0.1 atomikerrosta/s, $T = 800$ K ja energiaparametrit $E_S = 1.3$ eV, $E_N = 1.0$ eV (yläkuva), $E_S = 1.3$ eV, $E_N = 0.3$ eV (alakuva).

Kuvassa 8.31 on esimerkkejä erilaisilla simulaatioäparametreilla syntyvistä pintarakenteista. Kuvassa 8.32 esitetty saarekkeiden kokojakaumia erilaisille adatomikonsentraatioille.

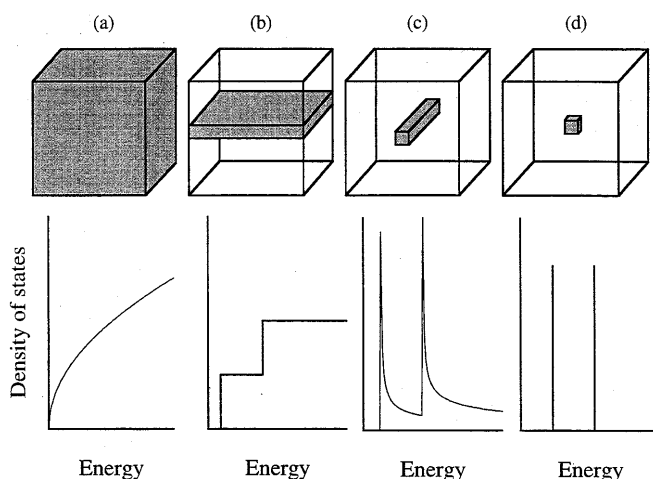


Kuva 8.32: Simulaatiossa syntyvien saarekkeiden kokojakautumat adatomikonsentraatioille 5, 15 ja 25%. Energiaparametrit olivat $E_S = 1.3$ eV ja $E_N = 1.0$ eV. Lämpötila oli 700 K ja depositionopeus 0.1 atomikerrosta/s. Yhtenäiset viivat ovat analyttisen mallin tuloksia.

Esitellään lopuksi MC-simulaatioiden sovellutusta eri kiteenkasvumoodien mallinnuksessa.

Pinnan kasvatus on tärkeä osa esimerkiksi uusien puolijohdekomponenttien ja varsinkin nano-rakenteiden valmistuksessa.

Alla esimerkkinä bulk-puolijohteen, kvanttikaivon, kvanttilangan ja kvanttipisteen elektronitiheydet.



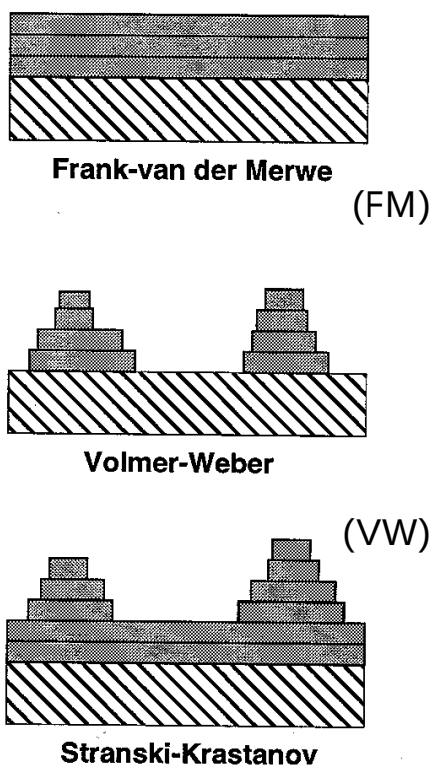
Kuva 8.33: Erilaisten puolijohderakenteiden energian tilatiheyksiä.

Esimerkiksi puolijohdelasereiden tapauksessa on kvanttipisteeseen perustuvilla komponenteilla hyviä ominaisuuksia. Näiden rakenteiden valmistaminen on jossain määrin hankalaa ja vaatii kasvatusprosessin hyvän tuntemisen.

Ohutkalvojen kasvatuksessa on useita menetelmiä joilla päästään varsin puhtaisiin materiaaleihin: LPE, VPE, MBE, ALE...¹

Kiteen kasvussa voidaan erottaa kolme erilaista moodia:

1. Markku Sopanen, *Self-organized growth and optical spectroscopy of semiconductor nanostructures*, väitöskirja, TKK, 1997.

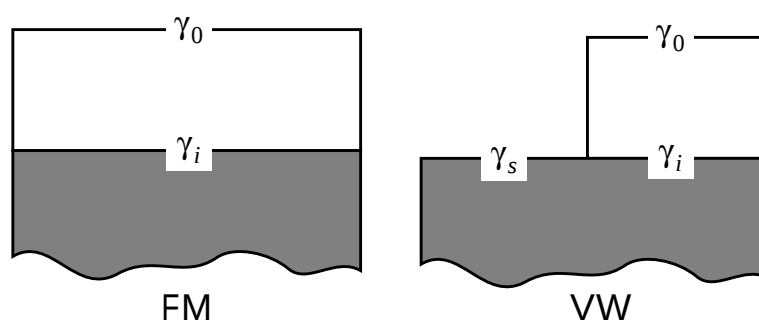


Kuva 8.34: Kiteen kasvumoodit.

FM-moodissa kide kasvaa tasaisesti kerros kerrokselta. VW-moodissa substraatin pinnalle syntyy saarekkeita. SK-kasvu on näiden väliltä. Yleensä muutaman tasaisesti kasvaneen kerroksen päälle alkaa syntyä saarekkeita. Saarekkeiden koko tietysti vaihtelee systeemistä toiseen, mutta yleensä niiden koko on nanometrialueella.

Saarekekasvua hyväksikäyttäen pystytään valmistamaan edellämainittuja kvanttipisteitä.

Eri moodit voidaan karkeasti selittää energetiikalla.



Kuva 8.35: Eri kasvumoodien energetiikkaa.

Olkoon meillä ylläkuvatut tilanteet. Pintajännitykset (tai vapaat energiat/pinta-alayksikkö) ovat γ_0 : kasvukerros-vakuumi; γ_i : kasvukerros substraatti; γ_s : substraatti-vakuumi. Oletetaan, että VW-kasvussa saarekkeet täyttävät puolet substraatin pinnasta. voimme laskea eri konfiguraatioiden energiaeron:

$$\Delta E = E_{\text{FM}} - E_{\text{VW}} = (\gamma_0 + \gamma_i)A - \frac{1}{2}(\gamma_0 + \gamma_i + \gamma_s)A . \quad (8.58)$$

Täydellinen kastuminen eli FM-kasvu toteutuu, kun

$$\Delta\gamma = \gamma_0 + \gamma_i - \gamma_s < 0 , \quad (8.59)$$

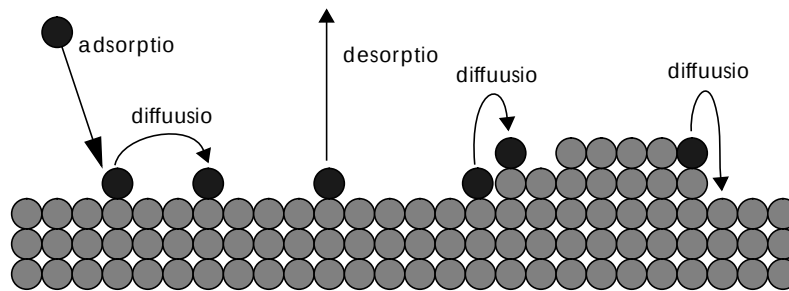
VW-kasvu, kun

$$\Delta\gamma > 0 , \quad (8.60)$$

js SK-kasvu, kun

$$\Delta\gamma \approx 0 . \quad (8.61)$$

Kuten jo aikaisemmin oli puhetta, kiteen kasvun simuloinnissa tärkeimmät tapahtumat ovat kuvan 8.36 mukaiset.



Kuva 8.36: Pinnan kasvun tärkeimmät tapahtumat.

Kuvassa 8.37 on esitetty kaavamaisesti kasvavan pinnan erilaisia atomitason rakenteita.

Adsorption nopeuteen vaikuttaa epätasapaino kemiallisessa potentiaalissa:

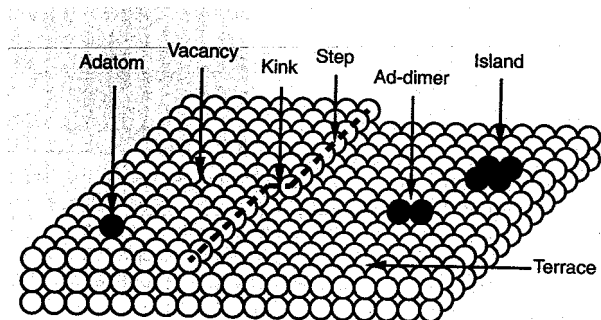
$$R_{\text{ads}} = R_{\text{ads}}^{\text{eq}} e^{\beta\Delta\mu}; \Delta\mu = \mu - \mu_{\text{eq}}. \quad (8.62)$$

Desorptionnopeus riippuu irtoavan atomin sidoksista:

$$R_{\text{des}} = \nu e^{\beta E_b} ;$$

$$E_b = m\varepsilon , \quad (8.63)$$

missä ε on energia sidosta kohti ja m on atomin lähinaapureiden lukumäärä.



Kuva 8.37: Materiaalin pinnan erilaisia atomikonfiguraatioita.

Atomin diffuusion nopeutta kuvataan usein Arrheniuksen dynamiikalla eli

$$R_{\text{diff}} = k_0 e^{-\beta E}, \quad (8.64)$$

missä E on diffuusiovallin energia, ja k_0 pinta-atomin värähtelytaajuus.

Simulaatiossa on varsin paljon parametreja, sillä esimerkiksi erilaisia diffuusiohyppyjä voi pinnan rakenteesta riippuen olla hyvinkin monta. Diffuusiovallien korkeudet voidaan laskea esimerkiksi molekyyldynaamisilla simulaatioilla tai jollain *ab initio* -menetelmällä.

Kuvassa 8.38 on esitetty muutama esimerkki eri kasvumoodien aikaansaamisesta MC:llä¹. Simulaatioissa atomien välinen vuorovaikutus kuvattiin Morsen potentiaalilla

$$V(r_{ij}) = D \left[e^{-2\alpha \left(\frac{r_{ij}}{r_0} - 1 \right)} - 2e^{-\alpha \left(\frac{r_{ij}}{r_0} - 1 \right)} \right], \quad (8.65)$$

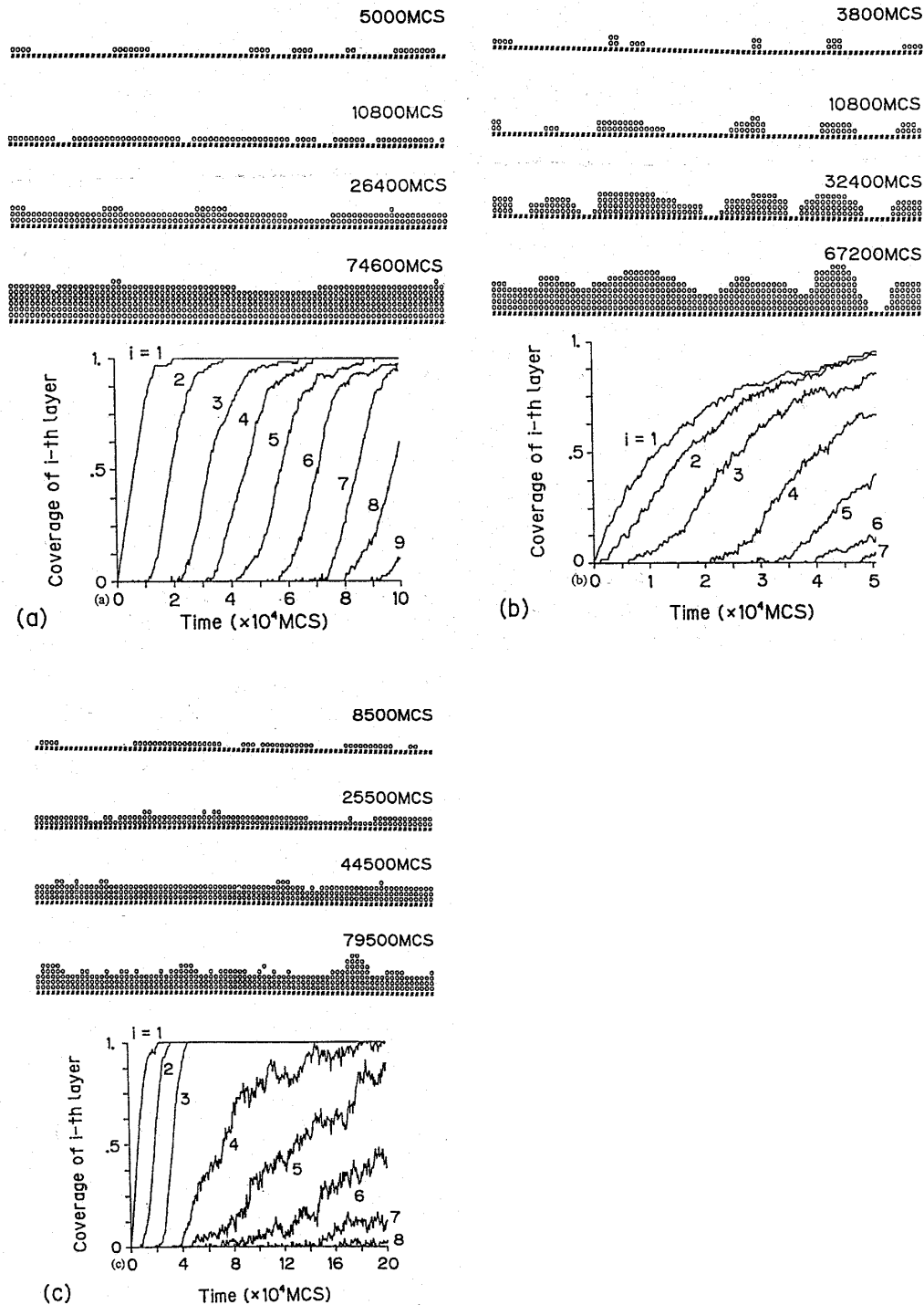
missä α kuvaa vuorovaikutuksen kantamaa, r_0 atomien välistä tasapainoetäisyyttä ja D vuorovaikutuksen vahvuutta. Vakiolle D käytettiin eri arvoja, kun kyseessä oli kalvon atomin (adatomien) vuorovaikutus substraatin kanssa ($D = E_{\text{fs}}$) ja kalvon atomien keskinäinen vuorovaikutus ($D = E_{\text{ff}}$). Näiden suhdetta varioimalla saatiin aikaan erilaiset kasvumoodit.

Kuvassa 8.38 esitettyjen simulaatioiden parametrit on esitetty taulukossa 8.1.

Taulukko 8.1: Kuvan 8.38 simulaatioiden parametrit.

	(a)	(b)	(c)
E_{ff} (eV)	0.2	0.4	0.1
E_{fs} (eV)	0.4	0.2	0.4
T (K)	300	300	390
α	6.0	6.0	2.0
r_0 (Å)	3.0	3.0	3.0
N_{MC}	100	100	100

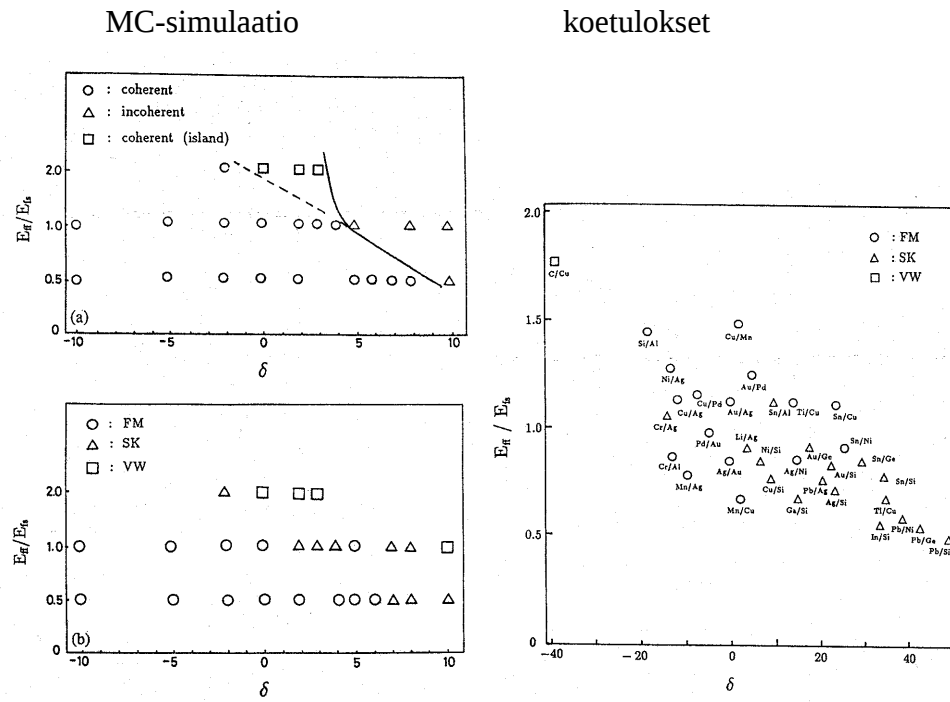
1. S. Ozawa, Y. Sasajima, D. W. Heermann, *Thin Solid Films*, **272** (1996) 172.



Kuva 8.38: Eri kasvumoodien aikaansaaminen KMC-simulaatiolla.

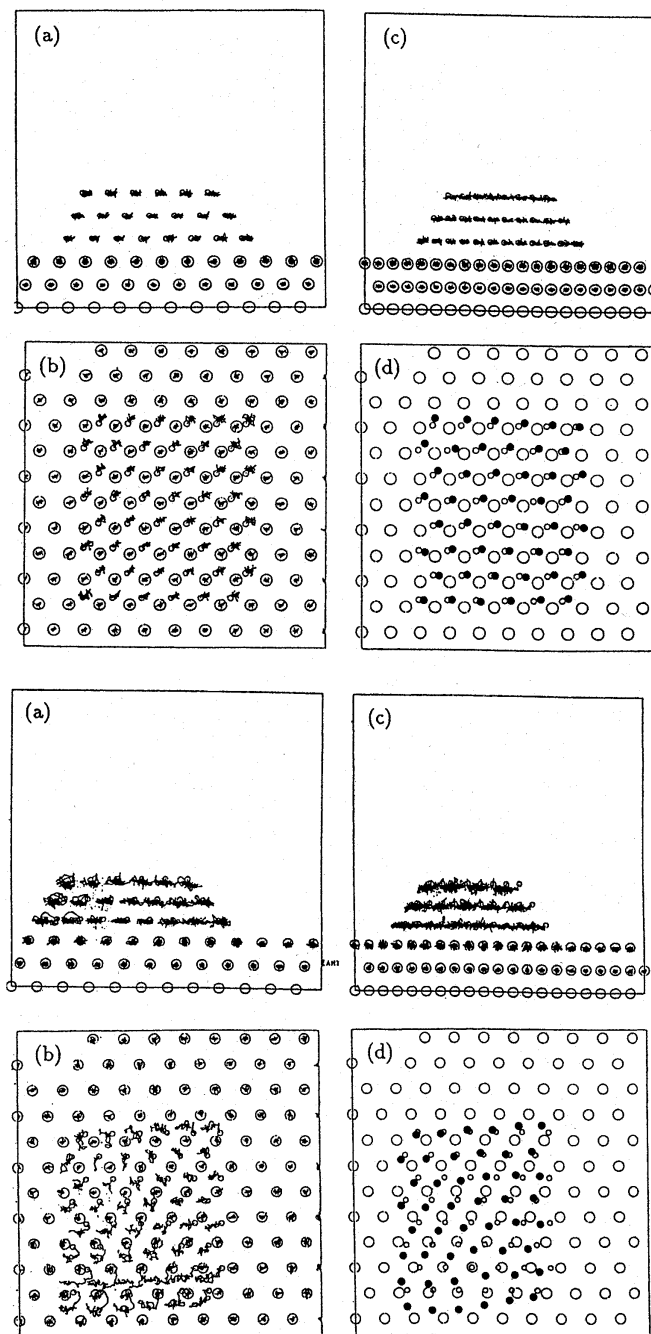
Eri moodit on saatu siis aikaan varioimalla suhdetta E_{ff}/E_{fs} .

Lisäksi tekijät kartoittivat, millä suhteen E_{ff}/E_{fs} ja kasvatettavan aineen ja substraattimateriaalin hilaparametrien eron δ arvoilla syntyy mikäkin kasvumekanismi.



Kuva 8.39: Kasvumoodin riippuvuus energiasuhteesta E_{ff}/E_{fs} ja hilaparametrierosta δ .

Lisäksi voisi vielä mainita, että MD-simulaatioita (joita ei voi käyttää koko kasvuprosessin kuvaamiseen) voidaan soveltaa tiettyihin yksityiskohtiin.



Kuva 8.40: Saarekekasvun mallintaminen MD-simulaatioilla¹. Tietynmuotoisen saarekkeen tasapainokonfiguraatiot on laskettu simuloidulla jäähdytyksellä. Neljä ylintä kuvaa: Ni/Cu(111), neljä alinta kuvaa: Cu/Au(111). Kuvissa (d) saarekkeiden atomien alkuperäiset paikat on esitetty pienillä avoimilla ympyröillä ja jäähdytyksen jälkeiset paikat pienillä ympyröillä.

1. S. Ozawa, Y. Sasajima, D. W. Heermann, *Thin Solid Films*, **272** (1996) 172.

A Van der Waalsin tilanyhtälö

Ideaalikaasumallissa oletetaan, että kaasumolekyylit ovat massapistettä, jotka eivät vuorovaikuta toistensa kanssa. Ainoastaan molekyyliden ja kaasun sisältävän astian seinien välillä on vuorovaikutus. Ideaalikaasun tilanyhtälöhän on

$$P = \frac{Nk_B T}{V} = nk_B T, \quad (\text{A.1})$$

missä $n = N/V$ on molekyylitiheys.

Mallia voidaan yleistää ottamalla mukaan molekyyliden välinen repulsio ja attraktio. Repulsio voidaan ottaa huomioon olettamalla molekyylit koviksi palloiksi, joiden halkaisija on d . Tällöin kunkin molekyylin tilavuus on

$$v_d = \frac{4}{3} \left(\frac{d}{2} \right)^3. \quad (\text{A.2})$$

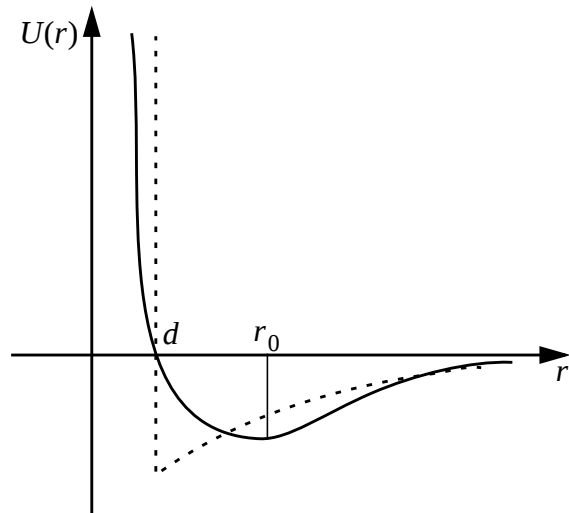
Mitä tästä seuraa? Ideaalikaasun tapauksessa molekyyleillä oli koko tilavuus V käytettävissä. Nyt molekyyliden¹ välinen etäisyys tulee olla vähintään d . Voimme hahmottaa molekyyliden väliset törmäykset siten, että joka törmäyksessä toinen molekyyleistä on pistemäinen ja toinen on kova pallo, jonka halkaisija on $2d$ (kuva A.2). Näinollen molekyyleille kielletty tilavuus on keskimäärin neljä kertaa molekyyliden kokonaistilavuus. Eli

$$V_f = 4v_d N. \quad (\text{A.3})$$

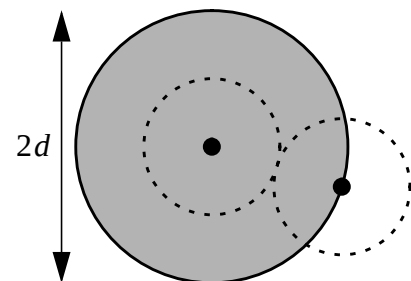
Tilanyhtälö (A.1) tulee nyt muotoon

$$P = \frac{Nk_B T}{V - 4v_d N}. \quad (\text{A.4})$$

Attraktion vaikutus kaukana astian reunoista on mitätön, koska molekyyliden välisten vetovoimien aikakeskiarvot häviävät. Astian seinien lähellä asia on toisin. Jos oletamme, että molekyyliden ja seinän välillä ei ole attraktiota, on molekyyliden välisten voimien aikakeskiarvo seinän lähellä seinästä poispäin.

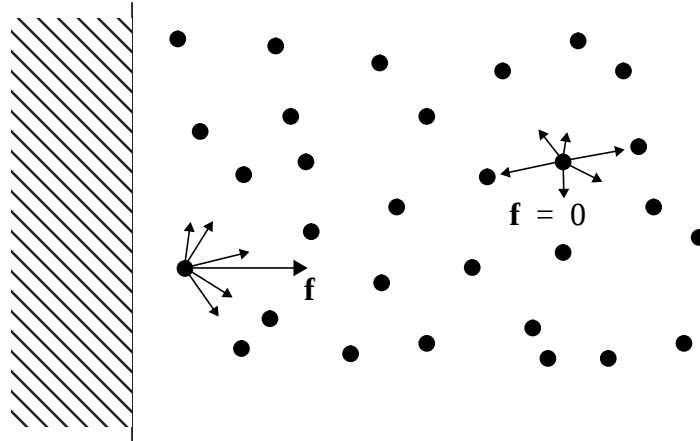


Kuva A.1: Todellinen molekyyliden välinen potentiaali (yhtenäinen viiva) ja sen approksimaationa käytetty vuorovaikutus.



Kuva A.2: Kielletty tilavuus.

1. Tai niiden massakeskipisteiden.



Kuva A.3: Molekyylien välisen attraktion vaikutus.

Tämä tiettyyn molekyyliin vaikuttava voima on suoraan verrannollinen sen kanssa vuorovaikuttavien molekyylien lukumäärään eli tiheyteen. Toisaalta voimasta aiheutuva paine on suoraan verrannollinen seinään törmäävien molekyylien lukumäärään. Näinollen attraktiosta aiheutuva paineen alennus on kääntäen verrannollinen tilavuuden neliöön.

Tilanyhtälö tulee siis lopulta muotoon

$$P = \frac{Nk_B T}{V - B} - \frac{AN^2}{V^2}, \quad (\text{A.5})$$

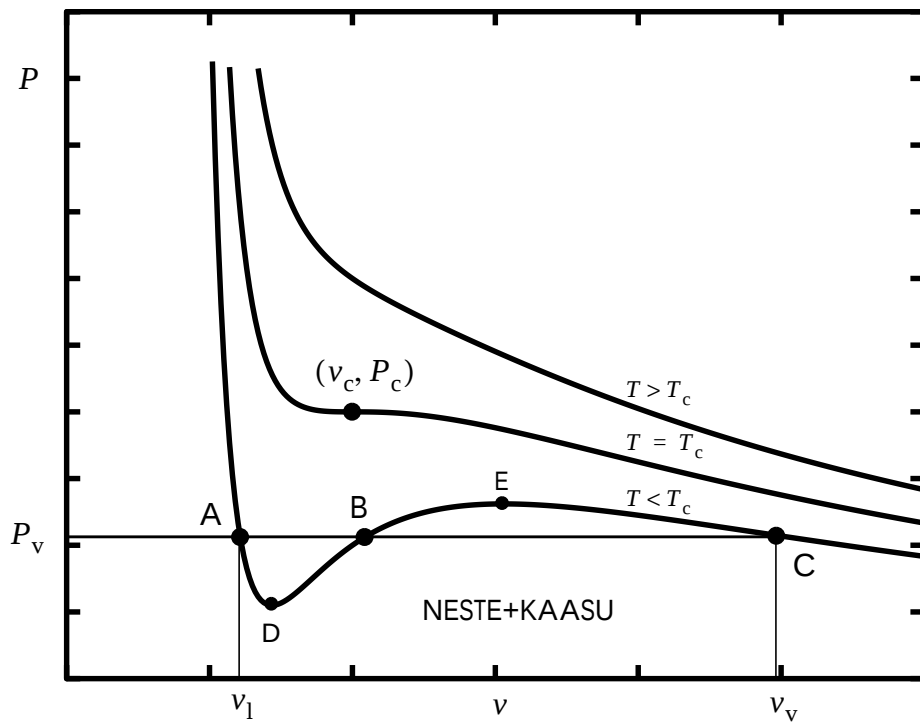
missä vakiot A ja B määritetään useimmiten kokeellisesti.

Usein tilavuuden sijasta käytetään tilavuutta molekyyliä kohti (tai moolitilavuutta) $v = V/N$, jolloin kaikki suureet tilanyhtälössä ovat intensiivisiä:

$$P = \frac{k_B T}{v - b} - \frac{a}{v^2}. \quad (\text{A.6})$$

Kuvassa A.4 on esitetty van der Waalsin tilanyhtälön isotermejä $P = P(v, T = \text{vakio})$ kolmella eri lämpötilalla¹.

1. Suhteessa kriittiseen lämpötilaan T_c , josta lisää hieman myöhemmin.



Kuva A.4: Van der Waalsin tilanyhtälön ennustamia isotermejä.

Kun $T \geq T_c$, isotermi on monotonisesti laskeva käyrä, eli tilavuutta pienennettäessä systeemin paine kasvaa. Kun lämpötila on tarpeeksi korkea ja tilavuus suuri, saadaan ideaalikaasulain ennustama hyperbeli.

Kun $T < T_c$, tilavuutta pienennettäessä paine ensin kasvaa, mutta tietyssä vaiheessa tilanyhtälö ennustaa, että se alkaa pienenemään. Pisteiden D ja E välisellä alueella isotermillä on positiivinen derivaatta eli $(\partial P / \partial V)_T > 0$. Tällainen materiaali on kuitenkin epästabiili, joten nämä alueet eivät vastaa aineen todellista tilaa.

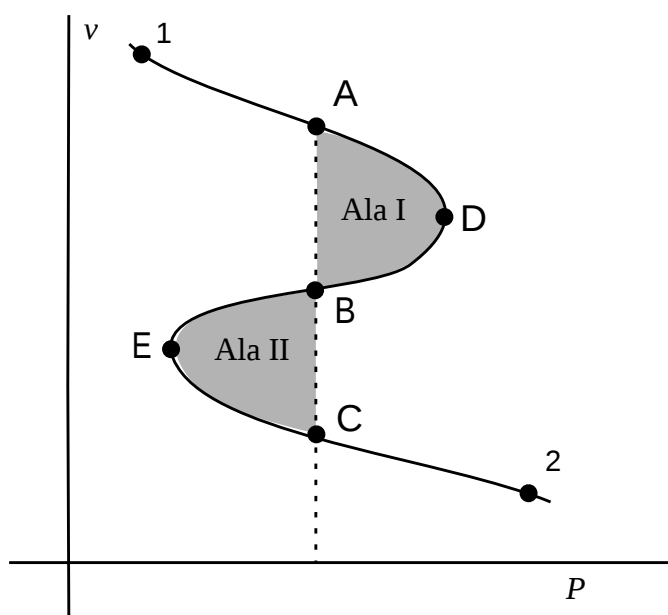
Todellisuudessa pisteessä C systeemin paine asettuu vakioarvoonsa, ja kaasun sekaan alkaa muodostua nestepisaroihin. Tultaessa pisteeseen A on kaikki kaasu tiivistynyt nesteeksi. Tästä tilavuutta edelleen pienennettäessä paine noudattaa nesteen tilanyhtälöä. Tätä voimme perustella käyttäen ns. Gibbsin ja Duhemin yhtälöä

$$d\mu = -s dT + v dp, \quad s = \frac{S}{N} \quad (\text{A.7})$$

ja sitä, että aineen stabiilisuus vaatii, että Gibbsin vapaa energia on kovera paineen funktiona

$$\left(\frac{\partial^2 G}{\partial P^2} \right)_{T, N} < 0. \quad (\text{A.8})$$

Tarkastellaan kuvan A.4 isotermiä $T < T_c$.

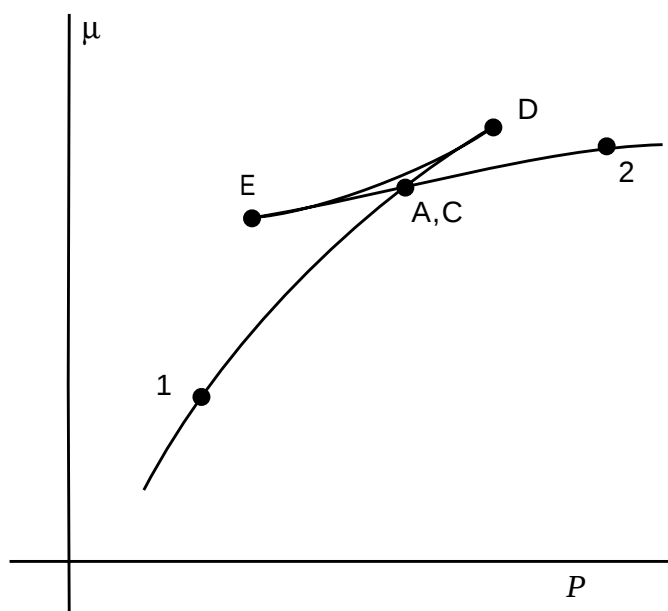


Kuva A.5: Van der Waals -isotermi, kun $T < T_c$.

Yhtälö (A.7) tulee isotermillä muotoon

$$d\mu = vdp . \quad (\text{A.9})$$

Tätä käyttäen voimme tutkia, miten kemiallinen potentiaali¹ käyttäytyy kuljettaessa isotermillä reittiä 1ADBEC2 pitkin.



Kuva A.6: Kemiallisen potentiaalin käyttäytyminen kuvan A.5 mukaisella isotermillä.

1. Tai Gibbsin vapaa energia hiukkasta kohti.

Välillä 1AD μ on nouseva ja kovera, välillä DE se on laskeva ja kupera ja välillä EC2 se on jälleen nouseva ja kovera. Väli DE ei siten vastaa mitään reaalista tilaa. Välit 1AD ja EC2 ovat mekaanisesti stabiileja, mutta vain reitti 1A2 vastaa tasapainotilaa, koska sillä kemiallinen potentiaali on minimissään. Ainoa tapa mennä pisteestä A pisteeseen C siten, että kemiallinen potentiaali pysyy vakiona on suora viiva eli paine pysyy vakiona.

Se mihin pisteet A ja C sitten asettuvat, saadaan ehdosta, että kemiallinen potentiaali näissä pisteissä on sama. Eli integroimalla (A.9) A:sta C:hen isotermiä pitkin saamme

$$\mu_C - \mu_A = \int_A^C v dp . \quad (\text{A.10})$$

Eli ehto on

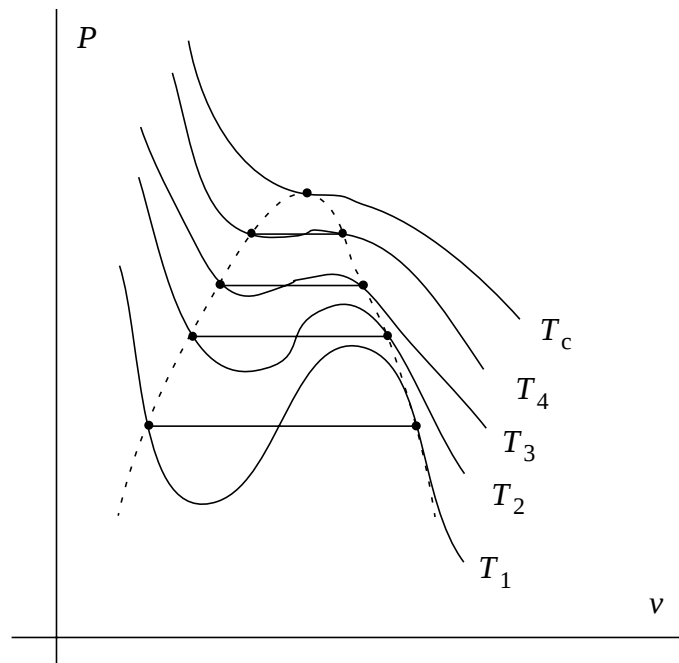
$$\int_A^C v dp = 0 . \quad (\text{A.11})$$

Kuvan A.5 mukaisen geometrisen tarkastelun mukaan pätee

$$\int_A^C v dp = \text{Ala I} - \text{Ala II} . \quad (\text{A.12})$$

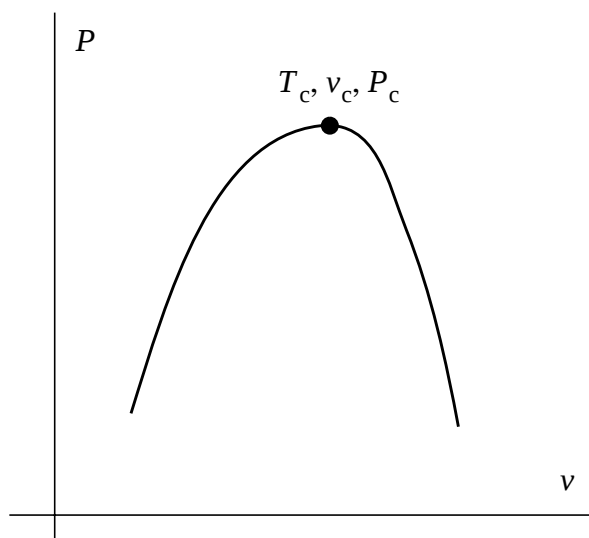
Eli jana AC on asetettava isotermille siten että pinta-alat I ja II ovat identtiset. Tätä kutsutaan Maxwellin konstruktioksi.

Lämpötilan lähestyessä kriittistä lämpötilaa T_c isotermin 'silmukka' kutistuu, ja kriittisessä lämpötilassa se häviää. Tällöin ei nestettä ja kaasua pysty erottamaan toisistaan.



Kuva A.7: Isotermien käyttäytyminen lähestyttäessä kriittistä lämpötilaa.

Pisteet A ja C eri lämpötiloissa piirtävät ns. koeksistenssikäyrän. Se kertoo kussakin paineessa, mitkä ovat nesteen ja kaasun tilavuudet (tai tiheydet).



Kuva A.8: Nesteen ja kaasun koeksistenssikäyrä.

Kriittisessä pisteessä ero nesteen ja kaasun välillä häviää.