

Mikael Myyrä

Reaaliaikaiset nestesimulaatiot

Tietotekniikan kandidaatintutkielma

20. toukokuuta 2019

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Mikael Myyrä

Yhteystiedot: mikael.b.myyra@student.jyu.fi

Työn nimi: Reaaliaikaiset nestesimulaatiot

Title in English: Real-time Fluid Simulations

Työ: Kandidaatintutkielma

Sivumäärä: 21+0

Tiivistelmä: Nesteiden virtausten mallintaminen tietokoneella on haastava ja paljon tutkittu ongelma. Kirjallisuuskatsauksena toteutetussa tutkielmassa tarkastelen ja vertailen erilaisia simulaatiomenetelmiä keskittyen erityisesti tietokonegrafiikan sovelluksiin, joissa reaaliaikainen vuorovaikutus on yleinen tavoite. Menetelmät voidaan luokitella aineen tilan esittämistavan perusteella ruudukko- ja hiukkasmenetelmiin. Molemmilla on omat etunsa ja haasteensa, mutta yleisiä ongelmia liittyy mm. numeeriseen stabiilisuuteen ja laskennan aikavaativuuteen. Lisäksi käyn vaihe vaiheelta läpi Stam'in ruudukkomenetelmän, joka on monen muun ruudukkomenetelmän perusta.

Avainsanat: simulointi, fluidit, nesteet

Abstract: Computer simulation of the flow of liquids is a challenging and extensively researched problem. In this literature review I examine and compare different simulation methods with a focus on computer graphics applications, in which real-time interaction is often desired. Methods can be categorized by their representation of the simulated material into grid-based and particle-based methods. Both categories have their own advantages and challenges, but common problems include e.g. numerical stability and computational complexity. Additionally, I provide a step-by-step overview of Stam's Stable Fluids method, which forms the basis of many other grid-based methods.

Keywords: simulations, fluids, liquids

Sisältö

1	JOHDANTO	1
2	NESTEET, SIMULAATIOT JA REAALIAIKAISUUS	2
3	MENETELMIÄ	5
	3.1 Ruudukkomenetelmät	5
	3.2 Hiukkaskomenetelmät	7
4	STAMIN RUUDUKKOMENETELMÄ	10
	4.1 Malli	10
	4.2 Menetelmän vaiheet	11
5	YHTEENVETO	14
	LÄHTEET	15

1 Johdanto

Nesteiden mallintaminen tietokoneella on monella tapaa hyödyllistä. Müller, Charypar ja Gross (2003) esittävät mallien käyttökohteiksi mm. suunnittelun ja testauksen aerodynamiikan sovelluksissa. He toteavat kuitenkin, että nesteiden monimutkaisen vuorovaikutuksen vuoksi tähän kuluu paljon laskenta-aikaa. Reaaliajassa tapahtuva laskenta on aikarajoitettua, mikä rajoittaa vuorostaan tulosten tarkkuutta ja sulkee pois tarkkuuskriittiset sovellukset. Aikarajoituksen hyötynä on mahdollisuus välittömään vuorovaikutukseen käyttäjän ja simulaation välillä. Tämä on tarpeen esimerkiksi tietokonepeleissä ja lääketieteellisissä koulutussimulaattoreissa, ja nopeampi laskenta nopeuttaa iterointia myös em. teollisuuskäytössä (Müller, Charypar ja Gross 2003).

Nesteiden mallintamisen ongelmaa voidaan lähestyä monella tavalla, joten tulosten koonti ja vertailu on yleiskuvan kannalta arvokasta. Siispä toteutan tämän tutkielman kirjallisuuskatsauksena. On välttämätöntä tuntea olemassaolevia menetelmiä, jotta niitä voi kehittää eteenpäin tai valita niistä omiin tarkoituksiin sopivimman. Kaikki nämä perustuvat kuitenkin pohjimmiltaan samaan fysikaaliseen malliin, joten yksityiskohtainen selostus yhdestä menetelmästä auttaa ymmärtämään myös muita.

Tutkin, minkälaisia menetelmiä on kehitetty, ja vertailen niiden etuja ja haasteita. Keskityn lähinnä grafiikassa ja peleissä käytettäviin menetelmiin, joissa tarkkuutta tärkeämpää on ulkonäkö. Luvussa 2 aloitan yleisellä, kaikkia menetelmiä yhdistävällä teorialla. Luvussa 3 luettelen, luokittelen ja vertailen menetelmiä melko yleisellä tasolla, ja pureudun lopuksi luvussa 4 yksityiskohtaisesti Stamin menetelmään (2003). Valitsin tämän menetelmän tarkasteltavaksi sen yksinkertaisuuden ja aloittelijaystävällisyyden takia.

2 Nesteet, simulaatiot ja reaaliaikaisuus

Kovien, törmäyksissä muotonsa säilyttävien, kappaleiden keskinäinen vuorovaikutus on verrattain yksinkertaista, koska se tapahtuu vain kappaleiden rajapinnassa. Nesteiden, ja yleisemmin aineen virtaavien olomuotojen, tapauksessa sen sijaan ei voida puhua erillisistä kappaleista, vaan kaikissa aineen pisteissä tapahtuu jatkuvasti muuttuvaa vuorovaikutusta ympäröivien hiukkasten kanssa. Näitä virtaavia olomuotoja ovat nesteet, kaasut ja plasmata, ja niitä kutsutaan fluideiksi. Sama matemaattinen malli kuvaa kaikkien näiden olomuotojen virtauksia.

Fluidin tilan kehitystä ajan suhteen kuvaavat Navier–Stokes-yhtälöt (Zsolnai ja Szirmay-Kalos 2012)

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{F}_{ext}, \quad (2.1)$$

$$\nabla \cdot \mathbf{u} = 0. \quad (2.2)$$

Yhtälö (2.1) kuvaa nopeuden muutosta, ja yhtälö (2.2) määrittelee puristumattomuuden sekä massan säilymisen. Tässä \mathbf{u} on paikallinen nopeus, ν on viskositeetti, ρ on tiheys, p on paine, ja \mathbf{F}_{ext} kuvaa ulkoisia voimia. Näiden merkitystä käsitellään tarkemmin luvussa 4. Malli (2.1) täydennetään sovellustilanteeseen sopivilla alku- ja reunaehdoilla (ks. esim. Moin ja Kim 1980; Wang, Wang ja Xin 2010).

Yhtälöt (2.1)–(2.2) kuvaavat *viskoottisen* (engl. *viscous*) aineen *puristumatonta* (engl. *incompressible*) virtausta. Viskositeetista puhutaan arkikielessä usein paksuutena tai jäykkyytenä. Se johtuu fluidin hiukkasten tarttumiskyvystä viereisiin hiukkasiin ja hidastaa liikenopeuksien muuttumista. Viskositeetin vaikutuksesta tapahtuvaa nopeuksien tasautumista kutsutaan diffuusioksi. Toinen nopeuksiin vaikuttava ilmiö on advektio, jossa fluidin osasten liikkuaessa niiden nopeudet ja muut ominaisuudet liikkuvat niiden mukana. Puristumattomuuden puolestaan voi ajatella tarkoittavan sellaista tilannetta, jossa fluidin tiheys ei muutu. Nesteiden virtauksissa puristuvuus on tavallisesti niin vähäistä, että täysin puristumaton malli approk-

simoi niitä riittävän tarkasti (Monaghan 2005, s. 1746).

Yksinkertaisimmassa mahdollisessa tilanteessa fluidi täyttää koko simulaatioavaruuden, ja Navier–Stokes-yhtälöiden ratkaisu kuvaa tarkasti järjestelmän kehityksen. Tämä malli ei kuitenkaan yksinään voi kuvata kaikkia vuorovaikutuksia, jotka ovat osallisena todellisen fluidin liikkeissä. Muiden aineiden ja aineen olomuotojen läsnäollessa yhtälöt pätevät edelleen, mutta aineiden väliset voimat noudattavat omia lakejaan.

Erityisesti nesteiden mallinnuksessa on yleistä, että fluidi ei täytä koko avaruutta. Tällöin on olemassa eri aineiden välisiä pintoja, joihin kohdistuu omanlaisiaan voimia. Nesteissä hiukasten väliset puoleensavetävät voimat aiheuttavat pintajännityksen, jonka vaikutus pyrkii minimoimaan pinnan alan (Akinci, Akinci ja Teschner 2013). Kiinteiden aineiden ja nesteiden välisillä pinnoilla esiintyy lisäksi tarttumista aiheuttavia voimia.

Fluidin liikkeiden mallintaminen algoritmisesti edellyttää mallin numeerista approksimointia äärellisessä joukossa ajan ja avaruuden pisteitä. Jos laskentaan käytettävää aikaa ei ole rajoitettu, voidaan käyttää suurta määrää laskentapisteitä ja lyhyttä aika-askelta ja tuottaa varsin tarkkoja arvioita todellisuudesta. Usein on kuitenkin tarpeen, että simulaatio tapahtuu reaaliajassa, ts. laskenta tapahtuu samaan aikaan kuin sen tulosten esittäminen. Tällöin käyttäjä voi suoraan vaikuttaa simulaation tilaan sen tapahtuessa. Tämä mahdollistaa käytön esimerkiksi peleissä tai simulaattoreissa.

Reaaliaikaisuus on merkittävä rajoite laskenta-ajalle — aika-askelen on oltava pidempi kuin sen laskentaan kuluva aika. Tämän saavuttamiseksi voidaan lisätä aika-askelen pituutta ja saada näin lisää aikaa laskennalle liikkeen visuaalisen sujuvuuden kustannuksella, tai toisaalta vähentää tilallisten laskentapisteiden määrää ja vähentää näin aika-askelen edellyttämää laskentaa yksityiskohtien kustannuksella. Molemmissa tapauksissa approksimaation fysikaalinen tarkkuus heikkenee. Tarkkuus- ja turvallisuuskriittisissä sovelluskohteissa, kuten aerodynamiikassa, näin ei päästä riittävän lähelle todellisuutta. Näissäkin tilanteissa nopeista menetelmistä voi tosin olla hyötyä — Prototyyppien testaus on sitä nopeampaa, mitä nopeammin simulaatiot voidaan laskea. Yleensä reaaliaikaisissa sovelluksissa tarkkuutta tärkeämpää on kuitenkin visuaalinen uskottavuus.

On olennaista, että simulaatio pysyy käyttökelpoisena riippumatta siitä, miten pitkään se on

käynnissä. Mikäli laskennassa tapahtuvat virheet voivat kertyä ajan myötä rajatta suuremmiksi, algoritmi on *numeerisesti epästabiili* (Ames 2014, s. 36-37). Virheen kasvaessa simulaatio muuttuu nopeasti hyödyttömäksi, joten numeerinen stabiilisuus on erittäin tärkeä ominaisuus.

3 Menetelmiä

Tietokoneella laskettaessa avaruus on diskretoitava rajalliseen määrään pisteitä, joissa voidaan approksimoida jatkuvien Navier–Stokes-yhtälöiden ratkaisuja. Pisteiden välisiä arvoja voidaan arvioida interpoloimalla, ja mitä lähempänä pisteet ovat toisiaan, sitä tarkempia tuloksia saadaan. Laskentapisteiden esitystavat voidaan luokitella kahteen ryhmään, jotka perustuvat Eulerin ja Lagrangen määritelmiin.

Eulerin määritelmässä (engl. *Eulerian specification*) fluidin tilaa tarkastellaan paikallaan pysyvissä avaruuden pisteissä. Pisteet voidaan valita monella tavalla, joten esitystapa tietorakenteena vaihtelee. Yksinkertaisimmillaan tämä on suorakulmainen ruudukko, jota kuvataan moniulotteisena taulukkona. Kutsun tähän perustuvia menetelmiä *ruudukkomenetelmiksi*.

Lagrangen määritelmässä (engl. *Langrangian specification*) sen sijaan seurataan yksittäisten fluidin osasten (engl. *fluid parcel*) liikeratoja. Tätä voidaan mallintaa toistensa kanssa vuorovaikuttavien hiukkasten järjestelmänä. Kutsun tähän perustuvia menetelmiä *hiukkasmenetelmiksi*.

Molempia esitystapoja on mahdollista käyttää myös saman järjestelmän osina. Esimerkiksi Cohen, Tariq ja Green (2010) käyttävät ruudukkomenetelmää eniten visuaalista tarkkuutta vaativalla, käyttäjän hallitsemalla alueella, mutta kauempana siirtyvät kevyempään hiukkaspohjaiseen malliin.

3.1 Ruudukkomenetelmät

Ruudukkomenetelmissä laskentapisteet pysyvät paikoillaan, ts. avaruuden rakenne on ennalta määriteltä. Tämä yksinkertaistaa laskentaa siltä osin, että toisin kuin hiukkasmenetelmissä, avaruuden hetkellisen rakenteen selvittämiseen ei tarvita lainkaan laskentaa. Rakenteen määrittelyyn on useita keinoja.

Yksinkertaisin tapa jakaa avaruus osiin ja valita laskentapisteet on rakentaa koordinaattiakselien suuntainen säännöllinen ruudukko. Ei kuitenkaan ole itsestään selvää, missä kohtaa ruudun sisällä piste on. Helppo valinta on suorittaa kaikki laskenta ruutujen keskipisteis-

sä (esim. Stam 2003), mutta porrastamalla ruudukko siten, että nopeudet lasketaan ruutujen rajapinnoissa ja muut arvot niiden keskipisteissä päästään tarkempaan lopputulokseen (Fedkiw, Stam ja Jensen 2001).

Suorakulmaisen ruudukon avulla voidaan kuvata tarkasti vain suorakulmaisesti rajattuja alueita. Viistoja pintoja sisältävien tai muutoin monimutkaisempien kuvioiden kuvaaminen edellyttää avaruuden yksityiskohtaisempaa määrittelyä. Esimerkiksi Elcott ym. (2007) käyttävät simulaatioavaruutena mielivaltaisesti määriteltyä kolmioverkkoa.

Yleisesti käytetty lähtökohta ruudukkomenetelmille on Stamin (1999) stabiilit fluidit. Tähän perustuvat menetelmät ovat stabiileja riippumatta aika-askelen pituudesta, mutta kärsivät numeerisesta diffuusiosta: laskentavirheet aiheuttavat energian katoamista ja hidastavat fluidin liikettä epäluonnollisen nopeasti (Elcott ym. 2007). Luvussa 4 käsiteltävä menetelmä (Stam 2003) on jatkokehitetty versio stabiileista fluideista.

Yksinkertaisimmassa tilanteessa ruudukkona mallinnettua fluidia on kaikissa ruudukon pisteissä, ja puristumattomana sen tilavuus ja tiheys pysyvät vakioina. Tällaisissa tilanteissa ruudukkomenetelmien etuna on toteutuksen yksinkertaisuus. Sen sijaan jos simulaatioon sisältyy muuta ainetta tai tyhjää tilaa, niin täytyy ylläpitää tietoa fluidin pinnan sijainnista, mikä lisää monimutkaisuutta verrattuna hiukkasmenetelmiin. Tähän on olemassa useita menetelmiä, jotka simulaatiomenetelmien tavoin voivat perustua ruudukkoon, hiukkasiin tai molempien yhdistelmään (Enright ym. 2002).

Toinen haastava ongelma on vuorovaikutus liikkuvien kiinteiden kappaleiden kanssa. Joissain tilanteissa riittää yksisuuntainen vaikutus fluidista kappaleeseen tai päin vastoin, mutta usein tarvitaan ongelmallisempaa kaksisuuntaista vuorovaikutusta. Eräs ratkaisu on Carlsonin, Muchan ja Turkin (2004) *kiinteä fluidi* -menetelmä, jossa fluidin kanssa vuorovaikutuksessa oleva kiinteä kappale kuvataan ruudukossa muotonsa säilyttävänä fluidin osa-alueena.

Merkittävä rajoite on, että ruudukon on katettava koko tarkasteltava avaruus, mutta käytettävissä on erityisesti reaaliajassa rajallinen määrä laskentapisteitä. Tästä seuraa, että mallinnettavan avaruuden koko on laskentatehon ja muistin rajoittama. Hiukkasmenetelmät sen sijaan pystyvät ilman erityisiä toimenpiteitä mallintamaan sellaisia fluideja, joita ei ole kaikkialla, joten niitä voidaan käyttää rajattoman suuressa tilassa; laskentateho rajoittaa vain aineen

kokonaismäärää.

Kullekin ruudulle tehtävät laskelmat yhden aika-askelen aikana riippuvat koko järjestelmän tilasta aika-askelen alussa. Aika-askelen laskeminen yhdelle ruudulle ei vaikuta muiden ruutujen lopputuloksiin. Tästä seuraa, että jokainen ruutu voidaan laskea samanaikaisesti. Toisin sanoen ruudukkomenetelmät rinnakkaistuvat hyvin, ja niitä voidaan ratkaista tehokkaasti näytönohjainlaitteiston avulla (esim. Harris 2005). Sama pätee pääsääntöisesti myös hiukkasmenetelmille.

3.2 Hiukkasmenetelmät

Hiukkasmenetelmissä fluidi koostuu vapaasti liikkuvista hiukkasista, joilla on omat nopeutensa ja massansa. Fluidimainen liike syntyy lähekkäisten hiukkasten keskinäisestä vuorovaikutuksesta. Tämä ratkaisee joitakin ruudukkomenetelmissä esiintyviä haasteita yksinkertaisella tavalla.

Koska kullakin hiukkasella on oma massa ja hiukkasten määrä ei yleisesti ottaen muutu, niin massan säilyminen on taattu ilman laskentaa (Müller, Charypar ja Gross 2003). Lisäksi hiukkasten nopeus liikkuu luonnostaan niiden mukana, joten Navier–Stokes-liikemääräyhtälössä (2.1) nopeuskentän gradienttia tarvitsevan advektiotermin $-(\mathbf{u} \cdot \nabla)\mathbf{u}$ (ks. luku 4) vaikutus toteutuu yksinkertaisesti hiukkasia liikuttamalla (Müller, Charypar ja Gross 2003). Myös vapaiden pintojen läsnäolo toteutuu yksinkertaisesti, kun hiukkaset täyttävät vain osan simulaatioavaruudesta. Pinnalla esiintyvät voimat edellyttävät edelleen omia laskelmiaan (Akinci, Akinci ja Teschner 2013), mutta ruudukkomenetelmissä välttämätöntä pinnan sijainnin seuraamista ei tarvita.

Hiukkasmenetelmien perustana on Gingoldin ja Monaghanin (1977) tasoitettu hiukkashydrodynamiikka (engl. *Smoothed Particle Hydrodynamics*, SPH). SPH on tapa muodostaa interpoloimalla erillisistä hiukkasista kenttä, jonka arvo voidaan laskea kaikissa avaruuden pisteissä (Müller, Charypar ja Gross 2003).

SPH:n mukaan skalaarikentän A arvo pisteessä \mathbf{r} on

$$A_S(\mathbf{r}) = \sum_b m_b \frac{A_b}{\rho_b} W(\mathbf{r} - \mathbf{r}_b, h), \quad (3.1)$$

missä b on hiukkasen indeksi, m_b on hiukkasen b massa, ρ_b hiukkasten tiheys b :n kohdalla, A_b A :n arvo b :n kohdalla, \mathbf{r}_b hiukkasen b sijainti, ja $W(\mathbf{r}, h)$ on *tasoitussydin* (engl. *smoothing kernel*), joka määrää kunkin hiukkasen painotuksen (Monaghan 2005). Muuttuja h on mitataavatermi, joka vaikuttaa tasoitusytimen muotoon. Yleensä muuttujan h arvo on lähellä hiukkasten etäisyyttä normaalitilanteessa ja W funktio, jonka arvo on nolla, jos etäisyys on suurempi kuin $2h$ (Monaghan 2005, s. 1710).

Yhtälön (3.1) mukaan kaikki olemassaolevat hiukkaset vaikuttavat kaikkiin muihin hiukkasiin. Laskenta-algoritmin aikavaativuus on tällöin $O(n^2)$, missä n on hiukkasten lukumäärä. Koska käytetään tasoitussydinfunktioita, joiden arvo on aina nolla jonkin h :sta riippuvan alueen ulkopuolella, niin käytännössä voidaan rajoittaa vaikuttavien hiukkasten määrä niihin, joiden etäisyys tarkasteltavasta hiukkasesta on tätä pienempi. Näin ollen aputietorakenteen avulla voidaan sulkea pois suurin osa hiukkaspareista ilman tasoitussydinfunktion laskentaa. Näin optimoituina menetelmän aikavaativuus on $O(nm)$, missä m on vaikutusetäisyydellä olevien hiukkasten keskimääräinen lukumäärä (Müller, Charypar ja Gross 2003).

SPH on konseptina intuitiivinen — todellinen aine koostuu hiukkasista — mutta käytännössä siihen liittyy haastavia rajoitteita, joiden kiertämiseen tai lieventämiseen useimmat siihen perustuvat menetelmät tähtäävät. Ensimmäinen ongelma, joka kaikkien menetelmien on ratkaistava on, että SPH-malli ei suoraan takaa fysiikan lakien toteutumista; esimerkiksi hiukkasten välisten voimien symmetrisyyden saavuttamiseksi täytyy interpolointiyhtälöihin lisätä tekijöitä (Müller, Charypar ja Gross 2003).

SPH:n naapurihiukkasten summaan perustuva tiheysapproksimaatio voi aineiden rajapintojen yhteydessä saada lepotiheyttä matalampia arvoja, mikä aiheuttaa negatiivisen paineen ja vetää hiukkasia toisiaan kohti muodostaen epäuskottavia kasautumia (Macklin ja Müller 2013). Tämä voidaan ratkaista esimerkiksi rajoittamalla paine ei-negatiiviseksi tai lisäämällä kaikkialle keinotekoista painetta, mutta näin ei saada täysin todellisuutta vastaavia tuloksia; pintajännityksen realistinen mallintaminen edellyttää kehittyneempiä menetelmiä (Akinci, Akinci ja Teschner 2013).

Fysikaalisen tarkkuuden lisäksi ongelmia liittyy laskennan aikavaativuuteen ja stabiilisuu-teen. Puristumattomalle virtaukselle ominaisen tasaisen tiheyden ylläpitäminen on raskas

operaatio, joka voidaan suorittaa esimerkiksi ruudukkomenetelmissä tyypillisen projektion (ks. luku 4) avulla. *Heikosti puristuva SPH* (engl. *Weakly Compressible SPH*, WCSPH) (Becker ja Teschner 2007) sen sijaan sisällyttää puristumattomuusehdon järjestelmän liikeyhtälöön. Tämän menetelmän stabiilisuus edellyttää lyhyitä aika-askelia (Becker ja Teschner 2007), mikä rajoittaa reaaliaikaista käyttöä.

Projektiot ja WCSPH ovat jokseenkin laskennallisesti vaativia menetelmiä. Nopeampia, ja siten reaaliajassa hyödyllisempiä, menetelmiä ovat esimerkiksi *ennakoiva-korjaava puristumaton SPH* (engl. *Predictive-Corrective Incompressible SPH*, PCISPH) (Solenthaler ja Pajarola 2009) ja *sijaintipohjaiset fluidit* (engl. *Position-Based Fluids*, PBF) (Macklin ja Müller 2013). PCISPH-menetelmässä hiukkasten liikkeissä syntyvät tiheyden muutokset johdetaan painevoimiksi, ja näitä iteroidaan, kunnes vakiotiheys saavutetaan. PBF-menetelmässä liikkeen eteneminen ja rajoitteet määritellään soveltaen Verlet-integrointia, jossa nopeuden sijasta tallennetaan kaksi viimeisintä sijaintia (Hairer, Lubich ja Wanner 2003). Koska menetelmän liikerajoitteet kohdistuvat suoraan hiukkasten sijainteihin nopeuksien sijasta, niin nopeuksien hallitsematonta kasvua ei voi tapahtua riippumatta aika-askelen pituudesta. Lisäämällä simulaatioon keinotekoisia painetta saadaan aikaan kasautumista ehkäiseviä, joskin fysikaalisesti perusteettomia pintajännitysvoimia. Nämä ominaisuudet parantavat menetelmän stabiilisuutta ja vähentävät tarvittavaa hiukkastiheyttä verrattuna SPH-menetelmiin (Macklin ja Müller 2013).

Myös muita aineen olomuotoja kuin fluideja voidaan mallintaa hiukkasten avulla. Kovat ja pehmeät kiinteät kappaleet voidaan sopivin liikerajoittein kuvata toisiinsa kiinnittyneiden hiukkasten rykelminä. Tästä seuraa, että hiukkasmenetelmiä käyttävä järjestelmä voi kuvata useiden erityyppisten kappaleiden ja fluidien vuorovaikutuksia saman ratkaisinohjelman puitteissa (esim. Macklin ym. 2014).

4 Stamin ruudukkomenetelmä

Stamin (2003) ruudukkomenetelmä on erittäin yksinkertainen ja siten hyvä lähtökohta fluidi-simulaatioihin tutustumiseen. Se kuvaa homogeenistä fluidia äärellisen kokoisessa laatikossa ja fluidin mukana liikkuvan muun aineen tiheyttä. Tässä luvussa esitellään tämän menetelmän pääpiirteet mukaillen Stamin (2003) artikkelia.

4.1 Malli

Navier–Stokes-liikemääräyhtälön (2.1) mukaan fluidin nopeutta muuttavia tekijöitä on neljä. Advektio $-(\mathbf{u} \cdot \nabla)\mathbf{u}$ kuvaa nopeuden liikkumista fluidin liikkeen mukana. Fluidin sisäinen paine \mathbf{p} aiheuttaa liikettä suuremman paineen alueilta pienemmän suuntaan kääntäen verrannollisesti fluidin tiheyteen: $\frac{1}{\rho}\nabla\mathbf{p}$. Diffuusiosta $\nu\nabla^2\mathbf{u}$ nopeus tasautuu viskositeetin vaikutuksesta kohti lähiympäristön keskiarvoa. Lisäksi fluidiin voi vaikuttaa ulkopuolisia voimia \mathbf{F}_{ext} , kuten painovoima. Zsolnai ja Szirmay-Kalos (2012, s. 17-21) selittävät termien fysikaaliset ja matemaattiset perusteet yksityiskohtaisesti.

Stamin menetelmässä yhtälön (2.1) arvoja approksimoidaan ensin välittämättä painetermistä ja puristumattomuusehdosta, ja lopuksi puristumattomuus palautetaan Helmholtz–Hodge-hajotelmaan perustuvan projektio-operaation avulla. Projektiossa paine nollautuu (Stam 1999), ja ratkaistavaksi jää

$$\frac{\partial\mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} + \nu\nabla^2\mathbf{u} + \mathbf{F}_{ext}. \quad (4.1)$$

Fluidin mukana liikkuvan aineen tiheyttä kuvaa yhtälö

$$\frac{\partial\rho}{\partial t} = -(\mathbf{u} \cdot \nabla)\rho + \kappa\nabla^2\rho + \mathbf{S}, \quad (4.2)$$

missä ρ on kuljetettavan aineen tiheys, κ on diffuusionopeus, \mathbf{u} on ainetta kuljettavan fluidin paikallinen nopeus, ja \mathbf{S} on ainetta lisäävä tai poistava lähde-termi. Sekä yhtälön (4.1) että yhtälön (4.2) oikea puoli muodostuu advektio-, diffuusio- ja lähde-termien summana.

Laskenta-alueen diskretointi tapahtuu yksinkertaisesti säännöllisenä ruudukkona, ja kaikki laskenta suoritetaan ruutujen keskipisteiden suhteen. Toisin sanoen laskentapisteet asetellaan koordinaattiakselien suuntaisille suorille tasaisin välimatkoin.

Fluidin tilan kuvaamiseen n -ulotteisessa avaruudessa riittää yksittäinen nopeutta kuvaava vektorikenttä, joka on tietorakenteena n -ulotteinen taulukko n -ulotteisia reaalilukuvektoreita. Mukana liikkuvaa tiheyttä kuvaa samankokoinen n -ulotteinen skalaarikenttä. Molempien kenttien reunoilla on ylimääräinen kerros ruutuja, joissa lasketaan reunaehdot. Jos esimerkiksi fluidin poistuminen laskenta-alueelta halutaan estää, voidaan reunaehto asettaa siten, että nopeuden ulospäin suuntautunut komponentti on nolla laskenta-alueen reunalla.

4.2 Menetelmän vaiheet

Yhtälöiden (4.1) ja (4.2) diskretointi paikan suhteen tapahtuu laskenta-alueen diskretointia vastaavasti säännöllisenä ruudukkona. Ajan suhteen diskretointi tapahtuu valitsemalla sopiva aika-askellus. Esimerkiksi Turek 1999, s. 9–10 on luetellut puristumattoman virtauksen simuloimisessa käytettyjä aika- ja paikkadiskretointi- sekä ratkaisumenetelmien yhdistelmiä.

Paikka- ja aikadiskretoinnin jälkeen voidaan muodostaa ratkaistava yhtälöryhmä. Ratkaistavina muuttujina ovat nopeus ja tiheys, joiden muutosta ajan suhteen yhtälöt (4.1) ja (4.2) kuvaavat. Menetelmä etenee vaiheittain siten, että advektion, diffuusion ja lähteiden vaikutukset lasketaan erikseen.

Ratkaistavista muuttujista ensimmäisenä lasketaan tiheys, johon vaikuttavat yhtälön (4.2) mukaisesti fluidin nopeus, diffuusio ja ulkoiset ainetta tuottavat lähteet. Näistä ensimmäisenä käsitellään ulkoiset tiheyden lähteet. Menetelmä ei ota kantaa lähteiden sijainteihin tai voimakkuuksiin, vaan lähteet otetaan syötteenä käyttäjältä.

Diffuusiossa ainetta siirtyy ruutujen välillä suuremmista tiheyksistä pienempien suuntaan. Ainetta siirtyy yksittäisen ruudun ja sen välittömien naapurien välillä näiden tiheyksien erotuksen suhteessa. Helppo tapa laskea nämä muutokset olisi eksplisiittisesti laskea tiheyksien erotukset ja lisätä ne suoraan ruutujen nykyisiin arvoihin. Tämä menetelmä on kuitenkin epästabiili ja tuottaa hallitsemattomasti kasvavia värähtelyitä, kun diffuusionopeus on

suuri. Eräs tapa tuottaa kaikissa tilanteissa stabiili ratkaisu on implisiittinen aika-askellus: etsitään tiheys, josta diffuusio aika-askelen verran taaksepäin tuottaa nykyiset tiheyden arvot. Saadaan lineaarinen yhtälöryhmä, joka voidaan ratkaista iteratiivisesti Gauss–Seidelmenetelmällä. Haittapuolena implisiittisessä aika-askelluksessa on yhtälöryhmän ratkaisuun kuuluva laskenta-aika.

Advektio fluidin nopeuskenttää pitkin tapahtuu myös implisiittisesti. Jokaisen ruudun keskipisteeseen voidaan kuvitella hiukkanen, joka liikkuu aika-askelen aikana siihen nopeuskenttää seuraten jostain muusta pisteestä. Tämän pisteen tiheys aika-askelen alussa on ruudun uusi tiheys aika-askelen lopussa. Piste päättyy todennäköisesti jonnekin ruudukon laskentapisteiden välimaastoon, jolloin sen tiheysarvo saadaan lineaarisesti interpoloimalla lähimpien laskentapisteiden välillä. Tämä on helppo, nopea ja numeerisesti stabiili tapa laskea advektio, mutta paremman tarkkuuden ja energian säilymisen voisi laskenta-ajan kustannuksella saavuttaa jollain korkeamman asteen menetelmällä (esim. Molemakerin ym. (2008) käyttämä QUICK).

Yhtälö (4.1) kuvaa nopeuskentän muuttumista ajan suhteen ja muistuttaa merkittävästi tiheyden yhtälöä (4.2). Nopeuskenttä liikkuu itseään pitkin, tasoittuu viskositeetin vaikutuksesta ja ottaa vastaan voimia ympäristöstä. Yhtäläisyyksien vuoksi tiheyden ohjelmakoodia voidaan käyttää uudelleen nopeuden laskennassa.

Nopeuden laskennassa on tiheyden laskennasta poiketen huomioitava massan säilyminen. Puristumattomalle fluidille tämä tarkoittaa, että jokaiseen pisteeseen tulee yhtä paljon massaa kuin siitä lähtee, ts. nopeuskentän divergenssi on nolla kaikissa sen pisteissä. Tämä saavutetaan Helmholtz–Hodge-hajotelmaan perustuvan projektio-operaation avulla (ks. esim. Harris 2005).

Projektiossa ja advektiossa tapahtuu energiahukkaa, joka vaikuttaa erityisen voimakkaasti fluidin pyörteisyyteen (Elcott ym. 2007; Molemaker ym. 2008). Pyörteiden läsnäolo vaikuttaa merkittävästi simulaation visuaaliseen uskottavuuteen, joten pyörteisyyden säilyttämiseen tähtäävät toimenpiteet, kuten usein käytetty *pyörteisyyden eristäminen* (engl. *vorticity confinement*) (esim. Fedkiw, Stam ja Jensen 2001), voisivat parantaa tämän menetelmän lopputulosten laatua.

Stamin mukaan (2002) jaksollisessa avaruudessa menetelmää voidaan nopeuttaa Fourier-muunnoksen avulla. Diffuusio- ja projektioaskelten suorittaminen Fourier-avaruudessa yksinkertaistaa operaatioita merkittävästi, mutta menetelmän käyttökohteita rajoittaa jaksollisuuden vaatimus. Tämä tarkoittaa käytännössä, että liikkeet laskenta-alueen reunan ylitse siirtyvät vastakkaiselle reunalle, joten alueella ei voi olla kiinteitä rajoja.

5 Yhteenveto

Nesteiden ja muiden fluidien virtausten mallintaminen tietokoneella on haastava tehtävä. Kaikkia hiukkastasolla tapahtuvia vuorovaikutuksia ei voida laskentatehon puitteissa ottaa huomioon, joten tarvitaan makroskooppisen tason numeerisia approksimaatioita. Navier–Stokes-yhtälöt ovat jatkuvassa avaruudessa toimiva matemaattinen malli, jonka numeerisiin ratkaisuihin simulaatiomenetelmät perustuvat. Nesteiden puristumattomuus tekee niiden mallintamisesta erityisen haastavaa esimerkiksi kaasuihin verrattuna.

Reaaliajassa tapahtuvat simulaatiot ovat tiukasti rajoitetun laskenta-ajan vuoksi erityisen vaativia valitun menetelmän matemaattisten ominaisuuksien suhteen. Numeerinen stabiiliisuus on pystyttävä säilyttämään pitkistä aika-askelista ja harvoista laskentapisteistä huolimatta, mutta samalla on vältettävä liiallista energian menettämistä.

Fluidin rakennetta voidaan kuvata kahdella tavalla: paikallaan pysyvän ruudukon pisteissä tai liikkuvina hiukkasina. Ruudukoiden etu on tietorakenteiden yksinkertaisuus ja toteutuksen helppous, kun mallinnettava järjestelmä on yksinkertainen. Useiden keskenään vuorovaikutuksessa olevien aineiden mallintaminen sen sijaan edellyttää aineiden välisten rajapintojen kuvaamista ruudukossa, mikä monimutkaistaa mallia huomattavasti. Hiukkasmenetelmissä rajapintojen paikallistaminen on suoraviivaista, mutta pinnoilla esiintyvien voimien uskottava approksimointi on joka tapauksessa haastavaa.

Luvussa 4 esitelty Stamin ruudukkomenetelmä on yksinkertainen, mutta jokseenkin epätarkka paineprojektioon ja implisiittiseen aika-askellukseen perustuva malli. Suuri osa reaaliajassa toimivista ruudukkomenetelmistä perustuu tähän menetelmään, joten aloittelijalle se on hyvä lähtökohta kehittyneempiin menetelmiin tutustumiselle.

Lähteet

Akinci, Nadir, Gizem Akinci ja Matthias Teschner. 2013. “Versatile Surface Tension and Adhesion for SPH Fluids”. *ACM Trans. Graph.* (New York, NY, USA) 32, numero 6 (marraskuu): 182:1–182:8. ISSN: 0730-0301. doi:10.1145/2508363.2508395. <http://doi.acm.org/10.1145/2508363.2508395>.

Ames, William F. 2014. *Numerical methods for partial differential equations*. Academic press.

Becker, Markus, ja Matthias Teschner. 2007. “Weakly Compressible SPH for Free Surface Flows”. Teoksessa *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 209–217. SCA '07. San Diego, California: Eurographics Association. ISBN: 978-1-59593-624-0. <http://dl.acm.org/citation.cfm?id=1272690.1272719>.

Carlson, Mark, Peter J. Mucha ja Greg Turk. 2004. “Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid”. Teoksessa *ACM SIGGRAPH 2004 Papers*, 377–384. SIGGRAPH '04. Los Angeles, California: ACM. doi:10.1145/1186562.1015733. <http://doi.acm.org/10.1145/1186562.1015733>.

Cohen, Jonathan M., Sarah Tariq ja Simon Green. 2010. “Interactive Fluid-particle Simulation Using Translating Eulerian Grids”. Teoksessa *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 15–22. I3D '10. Washington, D.C.: ACM. ISBN: 978-1-60558-939-8. doi:10.1145/1730804.1730807. <http://doi.acm.org/10.1145/1730804.1730807>.

Elcott, Sharif, Yiyong Tong, Eva Kanso, Peter Schröder ja Mathieu Desbrun. 2007. “Stable, Circulation-preserving, Simplicial Fluids”. *ACM Trans. Graph.* (New York, NY, USA) 26, numero 1 (tammikuu). ISSN: 0730-0301. doi:10.1145/1189762.1189766. <http://doi.acm.org/10.1145/1189762.1189766>.

Enright, Douglas, Ronald Fedkiw, Joel Ferziger ja Ian Mitchell. 2002. "A Hybrid Particle Level Set Method for Improved Interface Capturing". *Journal of Computational Physics* 183 (1): 83–116. ISSN: 0021-9991. doi:<https://doi.org/10.1006/jcph.2002.7166>. <http://www.sciencedirect.com/science/article/pii/S0021999102971664>.

Fedkiw, Ronald, Jos Stam ja Henrik Wann Jensen. 2001. "Visual Simulation of Smoke". *Teoksessa Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 15–22. SIGGRAPH '01. New York, NY, USA: ACM. ISBN: 1-58113-374-X. doi:10.1145/383259.383260. <http://doi.acm.org/10.1145/383259.383260>.

Hairer, Ernst, Christian Lubich ja Gerhard Wanner. 2003. "Geometric numerical integration illustrated by the Störmer/Verlet method". *Acta Numerica* 12:399–450.

Harris, Mark J. 2005. "Fast fluid dynamics simulation on the GPU." *SIGGRAPH Courses* 220.

Macklin, Miles, ja Matthias Müller. 2013. "Position Based Fluids". *ACM Trans. Graph.* (New York, NY, USA) 32, numero 4 (heinäkuu): 104:1–104:12. ISSN: 0730-0301. doi:10.1145/2461912.2461984. <http://doi.acm.org/10.1145/2461912.2461984>.

Macklin, Miles, Matthias Müller, Nuttapong Chentanez ja Tae-Yong Kim. 2014. "Unified Particle Physics for Real-time Applications". *ACM Trans. Graph.* (New York, NY, USA) 33, numero 4 (heinäkuu): 153:1–153:12. ISSN: 0730-0301. doi:10.1145/2601097.2601152. <http://doi.acm.org/10.1145/2601097.2601152>.

Moin, P, ja J Kim. 1980. "On the numerical solution of time-dependent viscous incompressible fluid flows involving solid boundaries". *Journal of computational physics* 35 (3): 381–392.

- Molemaker, Jeroen, Jonathan M. Cohen, Sanjit Patel ja Jonyong Noh. 2008. “Low Viscosity Flow Simulations for Animation”. Teoksessa *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 9–18. SCA '08. Dublin, Ireland: Eurographics Association. ISBN: 978-3-905674-10-1. <http://dl.acm.org/citation.cfm?id=1632592.1632595>.
- Monaghan, J J. 2005. “Smoothed particle hydrodynamics”. *Reports on Progress in Physics* 68, numero 8 (heinäkuu): 1703–1759. doi:10.1088/0034-4885/68/8/r01. <https://doi.org/10.1088%2F0034-4885%2F68%2F8%2Fr01>.
- Monaghan, J. J., ja R. A. Gingold. 1977. “Smoothed particle hydrodynamics: theory and application to non-spherical stars”. *Monthly Notices of the Royal Astronomical Society* 181, numero 3 (joulukuu): 375–389. ISSN: 0035-8711. doi:10.1093/mnras/181.3.375. eprint: <http://oup.prod.sis.lan/mnras/article-pdf/181/3/375/3104055/mnras181-0375.pdf>. <https://dx.doi.org/10.1093/mnras/181.3.375>.
- Müller, Matthias, David Charypar ja Markus Gross. 2003. “Particle-based Fluid Simulation for Interactive Applications”. Teoksessa *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 154–159. SCA '03. San Diego, California: Eurographics Association. ISBN: 1-58113-659-5. <http://dl.acm.org/citation.cfm?id=846276.846298>.
- Solenthaler, B., ja R. Pajarola. 2009. “Predictive-corrective Incompressible SPH”. Teoksessa *ACM SIGGRAPH 2009 Papers*, 40:1–40:6. SIGGRAPH '09. New Orleans, Louisiana: ACM. ISBN: 978-1-60558-726-4. doi:10.1145/1576246.1531346. <http://doi.acm.org/10.1145/1576246.1531346>.
- Stam, Jos. 1999. “Stable Fluids”. Teoksessa *Siggraph*, 99:121–128.
- . 2002. “A Simple Fluid Solver Based on the FFT”. *J. Graph. Tools* (Natick, MA, USA) 6, numero 2 (syyskuu): 43–52. ISSN: 1086-7651. doi:10.1080/10867651.2001.10487540. <http://dx.doi.org/10.1080/10867651.2001.10487540>.
- . 2003. “Real-time fluid dynamics for games”. Teoksessa *Proceedings of the game developer conference*, 18:25.

Turek, Stefan. 1999. *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach*. Nide 6. Springer Science & Business Media.

Wang, Xiao-Ping, Ya-Guang Wang ja Zhouping Xin. 2010. “Boundary layers in incompressible Navier-Stokes equations with Navier boundary conditions for the vanishing viscosity limit”. *Communications in Mathematical Sciences* 8 (4): 965–998.

Zsolnai, Károly, ja László Szirmay-Kalos. 2012. “Real time simulation and control of Newtonian fluids using the Navier-Stokes equations”. Tutkielma. https://users.cg.tuwien.ac.at/zsolnai/gfx/fluid_control_msc_thesis/.