

Janne Kalikka

jalajoka@cc.jyu.fi
040 769 5172

Vieras­molekyylien sitoutumisen mallintaminen lipokaliiniproteiineissa

Ohjaaja: Jaakko Akola

Tiivistelmä

Tässä työssä tutkittiin DigA16- ja 1KT7-proteiineja. Proteiineille suoritettiin klassisia MD-simulaatioita käyttämällä Orac4.0-ohjelmistoa. DigA16-proteiinin kokeellisesti mitattu rakenne täydennettiin tietokoneella ja kokonaisesta proteiinista simuloitiin kolmea eri rakennetta. 1KT7-proteiinin simulaation lähtörakenteena oli kokeellisesti mitattu rakenne, jota ei tarvinnut täydentää. Simulaatioita suoritettiin kutakin 10,751 nanosekunnin verran, joista 0,751 ns oli alustusta ja 10 ns simulointia 300 Kelvinissä. Tästä 10 ns vaiheesta kerättiin analysoitu data. Simulaation aikana kokonaisenergia kasvoi tasaisesti. Kasvu oli niin pientä, että sen voidaan katsoa johtuneen täysin MD-simulaatiossa kumuloituvasta pyöristysvirheestä. Tämä on tavallista MD-simulaatioille. Kahdesta DigA16-proteiinin simulaatioista löytyi pieni konformaatiomuutos, jonka aikaskaala on noin 2 ns. Tässä konformaatiomuutoksessa proteiinin aminohappoketjun pään 4-5 reunimmaista aminohappoa muuttavat selvästi asentoaan. Lisäksi tutkittiin miten DigA16-proteiinissa aminohappojen 42 ja 170 välillä olevan rikki-rikki-sidoksen puuttuminen vaikutti proteiinin käyttäytymiseen. Tuloksista voidaan päätellä, että rikki-rikki-sidoksen puuttuminen aiheuttaa loppupään hännän vaeltamisen kauemmas muusta proteiinista. Kuitenkin sidoksen puuttuminen aiheutti vain lokaaleja asennonmuutoksia proteiinin reunalla ja esimerkiksi proteiinin tunnusomaiseen β -tynnyriin sidoksen puuttuminen ei vaikuttanut. Edellä mainittu konformaatiomuutos tapahtui aminohappoketjun toisessa päässä, mutta kuitenkin muutos tapahtui vain niissä simulaatioissa, joissa rikki-rikki-sidos oli irti. Kahdessa simulaatiossa oli mukana vierasmolekyylit, joka oli proteiinin onkalossa. DigA16:n vierasmolekyylinä oli digoksigeniini ja 1KT7:n vierasmolekyylinä retinoli. Kumpikaan vierasmolekyylit ei kokenut asennonmuutosta. Pientä heilumista tapahtui, mutta tämä on tavallista simulaatioissa. DigA16:n hydrofilisiä sivuryhmiä on lähellä digoksigeniiniä ja ne todennäköisesti muodostavat sen kanssa vetysidoksia. 1KT7 puolestaan pitää hydrofobisia fenyylirenkaita ja metyyliiryhmiä retinolin lähellä, mikä oli odotettavissa.

Sisältö

| | |
|--|-----------|
| 1 Johdanto | 4 |
| 1.1 Proteiinit | 4 |
| 1.1.1 Proteiinien avaruudellinen rakenne | 5 |
| 1.1.2 Proteiinien rakenteen kokeellinen mittaus | 7 |
| 1.1.3 Lipokaliinit ja niiden muokkaus | 8 |
| 1.2 Simuloidut proteiinit | 9 |
| 2 Menetelmät | 12 |
| 2.1 Voimakentät | 12 |
| 2.2 Ewaldin summaus | 15 |
| 2.3 MD-algoritmit | 16 |
| 2.3.1 Potentiaalijako moninkertaista aika-askelta käyttävissä simulaatioissa | 16 |
| 2.3.2 Symplektinen ja aikareversiibeli operaattori | 20 |
| 2.3.3 Lämpötilan ja paineen kontrolloiminen termostaatin ja barostaatin avulla | 23 |
| 2.4 Simulaatio käytännössä | 25 |
| 3 Tulokset | 29 |
| 3.1 DigA16-proteiinin simulaatiot | 30 |
| 3.1.1 Energetiikka | 30 |
| 3.1.2 Konformaatio | 35 |
| 3.1.3 Vierasmolekyylin käyttäytyminen | 41 |
| 3.2 1KT7-proteiinin simulaatio | 49 |
| 3.2.1 Energetiikka | 49 |
| 3.2.2 Konformaatio | 50 |
| 3.2.3 Vierasmolekyylin käyttäytyminen | 51 |
| 4 Johtopäätökset | 56 |

1 Johdanto



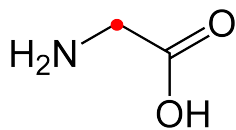
Kuva 1: *DigA16*-proteiini. Oranssilla värillä merkittynä ovat tietokoneella lisätyt aminohapot. Alkuperäisen kokeellisesti mitatun rakenteen aminohapot ovat turkoosilla värillä. Lisäksi kuvassa on erityisesti korostettu rikki-rikki-sidoksiin osallistuvia kystiini-aminohappoja järjestysnumeroltaan 8 ja 115 oikealla puolella sekä 42 ja 170 vasemmalla puolella.

1.1 Proteiinit

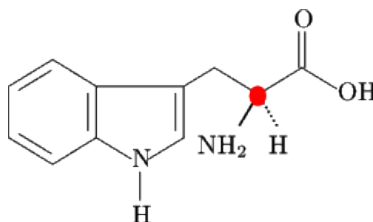
Proteiinit ovat biomolekyylejä, joita esiintyy kasveissa, eläimissä ja bakteereissa. Kuvassa 1 on *DigA16*-proteiini, joka kuuluu lipokaliiniproteiinien perheeseen. Proteiinit toimivat monissa dynaamisissa ja rakenteellisissa rooleissa. Ne esimerkiksi kuljettavat aineita kehossa, muodostavat kehikon solurakenteelle, toimivat solujen rakennusaineina tai toimivat entsyymeinä katalysoimassa kemiallisia muutoksia. Proteiinit myös suojelevat ihmisiä virus- ja bakteeri-infektioita vastaan.

Proteiinit rakentuvat aminohapoista, joita tunnetaan 22. Tosin vain 20:tä näistä on löydetty useammalta kuin yhdeltä lajilta. Proteiineissa aminohapot muodostavat pitkän poly-

peptidiketjun, jossa edellisen aminohapon karboksyyliiryhmään (ns. C-pää) on kovalentilla peptidisidoksella kiinnittynyt seuraavan aminohapon aminoryhmään (ns. N-pää). Yleisesti aminohappoketjun alkupäänä ymmärretään se pää, jossa on vapaa aminoryhmä ($-NH_3^+$) ja loppupäänä se pää, jossa on vapaa karboksyyliiryhmä ($-COO^-$). Kunkin aminohapon keskuksena on hiiliatomi (alfahiili, C_α), johon on kiinnittynyt neljä eri ryhmää: aminoryhmä, ($-NH_2$), karboksyyliiryhmä ($-COOH$), vetyatomi (H) sekä kullekin aminohapolle tunnusomainen sivuryhmä, jota merkitään tavallisesti kirjaimella R . Yksinkertaisimmillaan sivuryhmä on glysiinin vetyatomi (kuva 2), mutta se voi koostua myös kahdeksastatoista atomista kuten tryptofaanin tapauksessa (kuva 3). Tyypillisesti proteiini-nimitystä käytetään molekyyleistä, joissa on yli 50 aminohappoa. Pienemmistä käytetään nimitystä peptidi, esimerkiksi monista hormoneista, joiden molekyyli­massa on alle 5000 u. Aminohapot koostuvat tyypillisesti hiilestä, vedystä, hapesta, typestä ja rikistä [1].



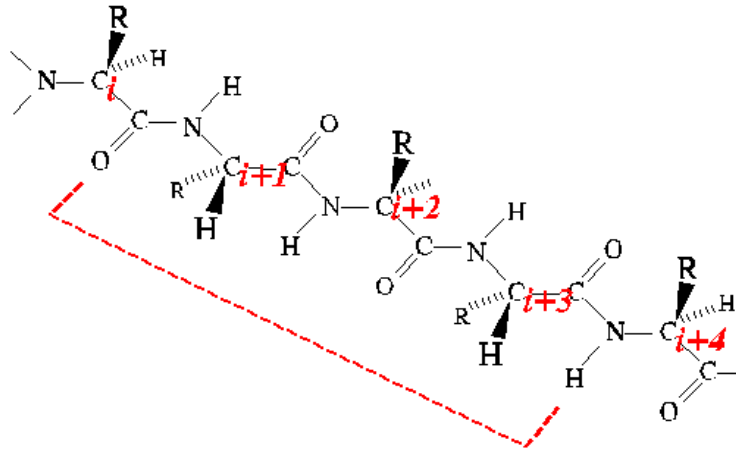
Kuva 2: Kaaviokuva glysiini-aminohaposta. C_α on merkitty punaisella. Kaksi yksittäistä C_α :n kiinnittynyttä vetyatomia on jätetty merkitsemättä. Toinen näistä on glysiinin sivuryhmä [2].



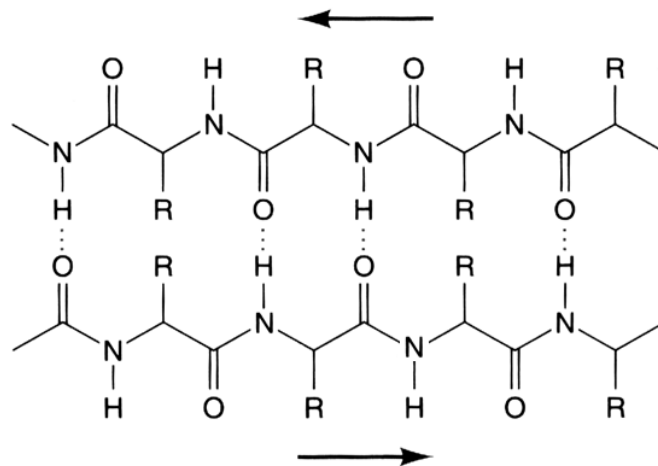
Kuva 3: Kaaviokuva tryptofaani-aminohaposta. C_α on merkitty punaisella [3].

1.1.1 Proteiinien avaruudellinen rakenne

Proteiinien rakenteen käsittely on jaettu neljään tasoon. Primaarirakenne käsittää proteiinin aminohappoketjun järjestyksen, josta näkyy mitä aminohappoja proteiineissa on ja missä järjestyksessä ne ovat ketjuuntuneet. Sekundaarirakenteella viitataan proteiinin eri osissa tavattaviin hieman laajempiin kokonaisuuksiin, jotka rakentuvat aminohappojen välisistä vety- ja rikkisidoksista. Yleisesti hyvinä esimerkkeinä tästä ovat α -kierteet (α -heliksit, engl. α -helix, kuva 4) ja β -levyt (engl. β -sheet, kuva 5). Proteiinin tertiaarirakenne tarkoittaa koko proteiinin kolmiulotteista rakennetta. Kvaternaarirakenteella tarkoitetaan usean proteiinin yhteenliittymän rakennetta. Suoriutuakseen roolistaan biokemiallisissa prosesseissa proteiinilla pitää olla tietty avaruudellinen rakenne. Proteiini pystyykin etsiytymään



Kuva 4: α -heliksin kaaviokuva. Punainen katkoviiva merkitsee vetysidosta neljän aminohapon päähän ketjussa [4].



Kuva 5: β -levyn kaaviokuva. Nuolet osoittavat aminohappoketjun suunnan (alkupäästä loppupäähän) [5].

luonnolliseen tertiaarirakenteeseensa spontaanisti, pelkästään aminohappoketjun rakenteen perusteella.

Tämä spontaani muotoutuminen (engl. folding) johtuu aminohappoketjussa vaikuttavista ei-kovalenteista voimista, jotka riippuen aminohappoketjun järjestyksestä ohjaavat proteiinin oikeaan tertiaarirakenteeseen. Nämä voimat johtuvat vetysidoksista ja heikoista pitkäkantaman vuorovaikutuksista proteiinin atomien välillä, mutta koska proteiinin jokainen atomipari vuorovaikuttaa, on voimien summa riittävän suuri [1]. Tärkein tällainen voima on hydrofobinen näennäisattraktio, jossa proteiinin hydrofobiset osat hakeutuvat yhteen, pienentäen reaktiopinta-alaa liuoksen veden kanssa. Hydrofiliset osat proteiinia taas pyrkivät kosketuksiin veden kanssa, jotta ne voivat muodostaa vetysidoksia vesimolekyylien kanssa. Lisäksi konformaation muodostumiseen vaikuttavat sähköstaattinen Coulombin vuorovaikutus sekä Van der Waalsin vuorovaikutus.

Van der Waals-voima itseasiassa aiheuttaa edellä mainitun hydrofobisen näennäisattraktion, sillä juuri tämä voima ajaa vesimolekyylit ja hydrofiliset molekyylien osat toistensa viereen, jättäen hydrofobiset osat toistensa lähelle. Van der Waals-voima aiheuttaa lisäksi lyhyen kantaman attraktion atomeiden välille. Sähköstaattiset voimat ovat yleensä heikkoja, koska varautuneet osat proteiinia hakeutuvat usein proteiinin pinnalle, jossa ne pääsevät kosketuksiin veden kanssa. Vedellä on kuitenkin suuri dielektrisyysvakio (huoneenlämmössä noin 80), joten kerääntyneet varaukset eivät vuorovaikuta kauas veden läpi. Proteiinin sisälle jääneet varaukset pääsevät vuorovaikuttamaan toistensa kanssa.

Jos proteiini denaturoituu esimerkiksi lämmön tai kemiallisten aineiden vaikutuksesta, sekundaari-, tertiaari- ja mahdollinen kvaternaarirakenne hajoavat. Tämän vuoksi proteiini ei voi suorittaa tavallista kemiallista tehtäväänsä ellei se pääse etsiytymään takaisin luonnolliseen muotoonsa. Proteiini voi myös menettää toiminnallisuutensa, jos se kokee pienen konformaatiomuutoksen jossakin toiminnallisuuden kannalta tärkeässä paikassa, vaikka muut osat pysyisivät entisellään.

1.1.2 Proteiinien rakenteen kokeellinen mittaus

Proteiinien primaarirakenne eli aminohappoketjun aminohappojen järjestys ja usein jopa tarkka avaruudellinen rakenne voidaan mitata. Ennen mittausta tavallisessa näytteessä olevat tuhannet erilaiset proteiinit pitää erottaa toisistaan. Tämä voidaan tehdä esimerkiksi elektroforeesilla tai jollakin molekyyli massa perustuvalla erotusmenetelmällä. Useissa proteiineissa on monta erillistä polypeptidiketjua, jotka pitää erottaa toisistaan ja puhdistaa ennen mittausta. Polypeptidiketjun primaarirakenne mitataan tavallisesti Edmanin reaktion avulla [1, sivu 74]. Edmanin reaktiossa polypeptidiketjujen annetaan reagoida fenylisothiosyanaatin kanssa, joka erottaa ketjusta ensimmäisen aminohapon - sen, jossa on vapaa NH_3^+ -ryhmä. Tämän jälkeen irroitettu aminohappo tunnistetaan kromatograafisesti. Prosessia voidaan toistaa noin 30-40 kertaa, tämän jälkeen liuoksessa on jo niin paljon epäpuhtauksia kesken jääneistä reaktioista, ettei lisää aminohappoja voida enää tunnistaa. Käytännössä proteiinit pitää siis ensin pilkkoa sopivan mittaisiksi pätkiksi, koska proteiinit useimmiten ovat pidempiä kuin 40 aminohappoa.

Proteiinien avaruudellista rakennetta (kuva 1) voidaan mitata kokeellisesti esim. rönt-

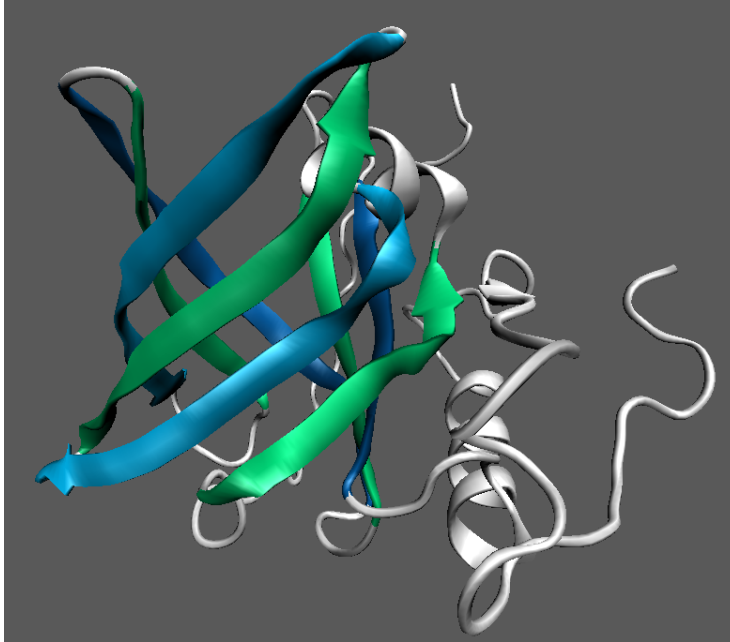
gendiffraktiolla kiteytetystä proteiinimolekyylien muodostamasta hilasta (röntgenkristallografia) tai NMR-mittauksilla (engl. nuclear magnetic resonance) kiteyttämättömästä rakenteesta. Kristallografiassa proteiinista tehdään hyvin vahvaa liuosta, joka muistuttaa lähinnä proteiinimolekyyleistä muodostunutta kidettä. Kiteessä on kuitenkin tyypillisesti 40-60% liuotinta, koska proteiinimolekyylit ovat suuria kokonaisuuksia ja niiden muodot voivat olla hyvinkin monimutkaisia. Itse mittauksessa röntgensäteeseen annetaan diffraktoitua kiteestä, tarkemmin ottaen atomien elektroniverhoista. Diffraktoituneesta säteilystä saadaan mitattua kiteen elektronitiheyden jakauma, josta voidaan päätellä proteiiniatomien paikat ja tyypit.

NMR-mittaus on toinen tapa saada tarkkoja atomaarisia koordinaatteja proteiineista. Tässä mittaustavassa on kuitenkin rajoituksensa. NMR-mittauksessa osa tavallisista hiili-, typpi- ja happi-isotoopeista (^{12}C , ^{14}N ja ^{16}O) pitää korvata isotoopeilla, joissa on pariton määrä nukleoneja (^{13}C , ^{15}N ja ^{17}O), jotta ytimessä olisi pariton spin-tila ja se absorboisi mittauksessa käytettyjä radiosäteilypulseja. NMR-mittauksilla on havaittu, että proteiini on jatkuvassa liikkeessä. Proteiinissa saattaa jopa tapahtua pieniä konformaatiomuutoksia spontaanisti. Röntgendiffraktiomittauksissa saadaan kyllä mittaustuloksena tarkat koordinaatit, mutta koska mittausta suoritetaan suurelle joukolle proteiinimolekyyliä, voivat jotkin proteiineista olla hieman eri asennossa, lisäksi atomien tiedetään olevan jatkuvasti pienessä liikkeessä. Mittaustuloksen koordinaatit ovat siis keskiarvoja atomien paikoille, nämä ovat kuitenkin riittävän tarkkoja esimerkiksi molekyyliidynamiikkasimulaation lähtöarvoiksi. Röntgendiffraktiomittauksissa on myös havaittu, että proteiinin atomit eivät ole kovin tiukkaan pakkautuneet proteiinin sisälläkään ja siellä oleva ylimääräinen tila mahdollisia pieniä konformaatiomuutoksia muuallakin, kuin proteiinin pinnalla [1, sivu 68]. Molekyyliidynamiikkasimulaatioissa on havaittu atomien värähtelevän keskimäärin 0,7 Å:n matkalla noin pikosekunnin aikaskaalalla [1, sivu 69].

1.1.3 Lipokaliinit ja niiden muokkaus

Lipokaliinit ovat joukko proteiineja, joita esiintyy luonnossa monessa paikassa. Niitä löytyy niin selkärangkaisissa ja selkärangattomissa eläimissä, bakteereissa ja kasveissakin. Lipokaliineilla on toisistaan hyvin poikkeavat aminohappoketjut. Kuitenkin yhteistä tälle ryhmälle on ennen kaikkea se, että kaikki lipokaliinit voivat sitoa hydrofobisia molekyyliä sisäänsä. Lipokaliineilla on kahdeksasta tai yhdeksästä β -levystä muodostunut ja sisältä hydrofobinen β -tynnyri (kuva 6). Tämän rakenteen vuoksi lipokaliinit pystyvät sitomaan sisäänsä pienempiä hydrofobisia molekyyliä kuten retinolia, steroideja ja lääkkeitä. Tynnyrin muodostavat β -levyt kiinnittyvät toisiinsa vedysidoksin, muodostaen vinon suunnikkaan muotoisen levyn. Tämä levy on kiertynyt rullalle ja kiinnittynyt reunoista itseensä muodostaen kiinteän tynnyrirakenteen. Yksittäinen β -levy taas muodostuu yhdensuuntaisista polypeptidiketjuista, jotka ovat kiinnittyneet toisiinsa vetysidoksin ja muodostavat vahvan levymäisen rakenteen.

Lipokaliineja, kuten muitakin proteiineja voidaan muokata pilkkomalla niitä, mutatoimalla yksittäisiä paloja ja rakentamalla paloista uusi, hieman erilainen proteiini [1, 7, 8]. Tätä menetelmää kutsutaan kohdennetuksi mutageneesiksi (engl. "site-directed mutagene-



Kuva 6: Kuva DigA16-proteiinista. Kuvassa korostettuna β -levyistä muodostunut tynnyri. Joka toinen β -levy sinisellä ja joka toinen vihreällä värillä korostettuna hahmottamisen helpottamiseksi.

sis”) [6]. Tällä tavalla voidaan vaikuttaa hyvin tarkasti siihen, mitä osaa proteiinista muokataan ja siten voidaan tutkia järjestelmällisesti tietyn osan funktiota ja sitä miten osan muokkaaminen vaikuttaa asiaan. Tällöin nähdään muuttuuko osan funktio, miten funktio suoritetaan, muuttuuko yleensäkin mikään vai oliko tutkittu funktio sittenkin jonkin toisen proteiinin osan aiheuttama? Rasitteena on se, että proteiinien muokkaus tällä tavalla vaatii tarkan tiedon proteiinin rakenteesta. Vaikka proteiinien rakenteet tunnetaan yleisesti, niin tieto ei välttämättä ole tarpeeksi yksityiskohtaista. Toinen hidaste on se, että vaikka mutaation paikka voidaan paikallistaa, aina ei voida ennustaa mitä vaikutuksia mutaatiolla on. Toisin sanoen mutatoitava paikka voidaan määrätä, mutta siitä aiheutuva muutos on silti enemmän tai vähemmän satunnaista.

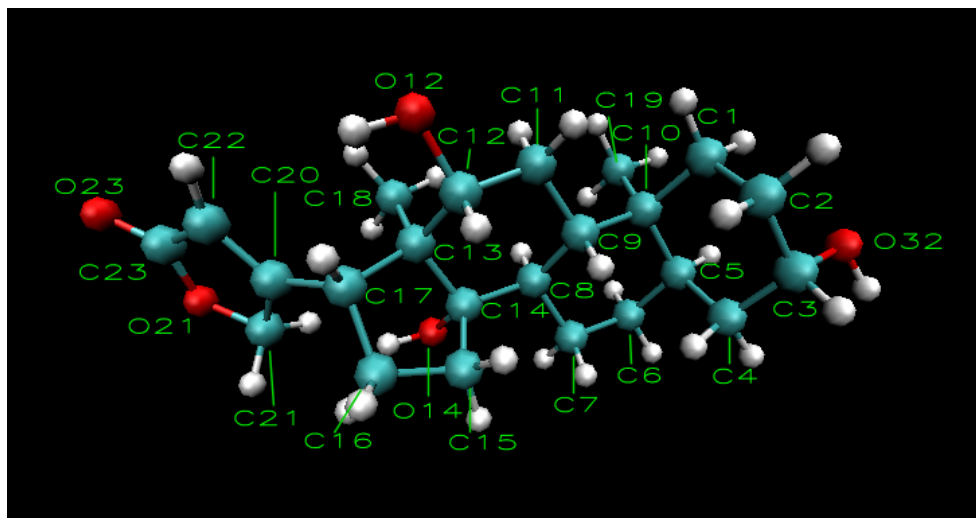
Esimerkkinä keinotekoisesti muokatuista proteiineista on tässäkin simulaatiossa käytetty DigA16 (kuva 1). Se on tehty muokkaamalla erästä kaaliperhosesta löytyvää lipokaliinia, erityisesti sen β -levyistä muodostunutta taskua, jonne se sitoo pienempiä molekyylejä, jotta DigA16 voisi sitoa taskuunsa digoksigeniinimolekyyliin. Tämän se tekeekin hyvin suurella tehokkuudella mikäli DigA16:a on samassa liuoksessa (esim. verenkierrossa) digoksigeniinin kanssa [7].

1.2 Simuloidut proteiinit

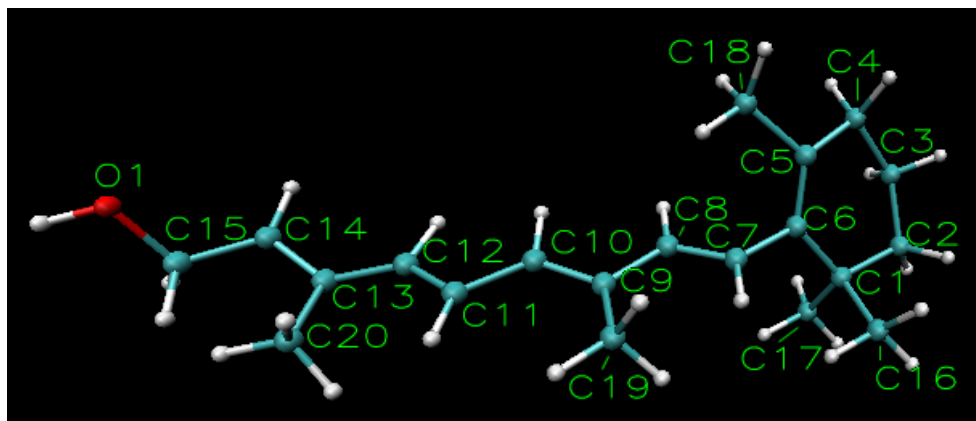
Tässä työssä simuloitiin kahta lipokaliiniproteiinia, DigA16-nimistä luonnossa kaaliperhosessa esiintyvistä lipokaliinista muokattua versiota [9] sekä 1KT7-nimistä [10] nautaeläimis-

sä esiintyvää lipokaliiniproteiinia Orac4.0-ohjelmistolla [11]. DigA16 sitoo sisäänsä digoksigeniinimolekyylin (kuva 7) ja siksi yhdessä kolmesta DigA16:n simulaatioista proteiinin onkalossa oli digoksigeniinimolekyyli. Digoksigeniiniä käytetään mm. ei-radioaktiiviseen merkitsemiseen, jolla voidaan seurata biomolekyylien kulkua elimistössä. Merkitsemisessä helpommin seurattavissa oleva merkkiainemolekyyli kiinnittyy seurattavaan molekyyliin ja siten halutun molekyylin kulkua voidaan seurata. Digoksigeniinin itsensä seuraamiseen käytetään fotoneita sopivalla aallonpituudella, sillä digoksigeniini on fotoaktiivinen molekyyli.

1KT7 sitoo luonnossa retinolimolekyylin (kuva 8) sisäänsä ja simulaatiossa proteiinin onkalossa olikin retinolimolekyyli. Retinoli (A-vitamiini) on luonnossa hyvin aktiivinen ja sitä tarvitaan moneen biokemialliseen prosessiin. Esimerkiksi solujen kasvuun ja näköaistin ylläpitoon. Silmässä retinaali (kehossa osa retinolistä muuttuu retinaaliksi) vaihtelee cis- ja trans-isomeeriensä välillä mahdollistaen näköaistin toiminnan. Valofotoni aiheuttaa konformaatiomuutoksen osuessaan cis-retinaaliin, joka on kiinnittynyt rodopsiinimolekyyliin silmän sauvasoluissa (silmän valonherkissä soluissa). Konformaatiomuutoksen seurauksena syntyy sähköimpulssi, joka kulkeutuu aivoihin ja vie tiedon silmään osuneesta fotonista. Prosessissa retinaali kokee konformaatiomuutoksen ja rodopsiinimolekyyli hajoaa trans-retinaaliksi ja opsiiniksi. Trans-retinaali kierrätetään, sillä se muuttuu trans-retinolin kautta cis-retinaaliksi ja yhdistyy opsiinin kanssa rodopsiiniksi, voidakseen jälleen havaita uuden valonsäteen silmässä [1, sivut 937-943].



Kuva 7: Digoksigeniinimolekyyli DigA16:n kolmannen simulaation alkurakenteesta. Hiiliatomit on merkitty turkoosilla, happiatomit punaisella ja vetyatomit valkoisella. Vetyatomien nimet ovat merkitsemättä, ne ovat muotoa $\{n\}H\{nC\}$, missä nC on vetyyn kiinnittyneen hiiliatomin numero ja n vetyatomin järjestysnumero siinä hiilessä. Esimerkiksi C16:een ovat kiinnittyneet vedyt 1H16 ja 2H16. Poikkeuksena nimeämisessä on happeen kiinnittyneet vedyt, O12:n kiinnittynyt vety on nimeltään HAA, O14:n kiinnittynyt HAB ja O32:n kiinnittynyt HAC.



Kuva 8: Retinolimolekyyli 1KT7:n simulaation alkurakenteesta. Hiiliatomit on merkitty turkoosilla, happiatomit punaisella ja vetyatomit valkoisella. Vetyatomien nimet ovat merkitsemättä, ne ovat muotoa $\{n\}H\{nC\}$, missä nC on vetyyn kiinnittyneen hiiliatomin numero ja n vetyatomien järjestysnumero siinä hiilessä (ks. kuva 7). Poikkeuksena nimeämisessä on happeen kiinnittynyt vety, jonka nimi on HAA.

DigA16-proteiinin kokeellisesti röntgendiffraktiolla selvitettyyn rakenteeseen oli lisätty tietokoneella puuttuvat osat, joiden koordinaatteja ei ole saatu mitattua proteiinikiteen flukтуаatioiden vuoksi. Näiden olemassaolo ja järjestys proteiinissa tiedetään primaarirakenteen mittausten kautta. 1KT7:n kokeellisesti mitatussa tertiäarirakenteessa ei ollut puutteita. DigA16:lla on käyttöä esimerkiksi lääketeollisuudessa, koska se sitoo digoksigeniiniä itseensä hyvin tehokkaasti.

Molempien käytettyjen proteiinien kokeellisesti mitattu avaruudellinen rakenne löytyy vapaasti internetistä suuresta proteiinitietopankista (Protein Databank [12]). DigA16:n tapauksessa rakenteessa oli siis puutteita ja simulaatiota varten tehtiin erillinen ohjelma, jolla nämä puuttuvat osat pystyttiin lisäämään. Tämän jälkeen kokonaisille proteiinimolekyyleille voitiin suorittaa molekyyldynamiikkasimulaatioita (MD-simulaatioita, engl. molecular dynamics simulations).

Kutakin neljää proteiinirakennetta simuloitiin 300 Kelvinissä (noin 27°C) kymmenen nanosekunnin ajan. Simuloinnin aikana tallennettiin atomien liikkeitä ja erilaisia energian arvoja. Ennen varsinaista simulaatiota rakenne alustettiin termalisoimalla se vähitellen kymmenestä Kelvinistä 300 Kelviniin. Esimerkiksi DigA16-simulaatiossa (ilman digoksigeniiniä) oli 17539 atomia $60 \times 55 \times 55$ Ångströmin ortorhombisessa kopissa. Atomeista 2854 kuului itse proteiiniin ja loput atomit olivat vettä tai vedessä olevia ioneita, jotka tasapainottivat simulaatiokopin kokonaisvarauksen neutraaliksi. Muissa simulaatioissa atomeita oli hieman poikkeava, mutta suuruusluokaltaan samanlainen määrä. Yhden simulaation ajaminen kesti hieman yli neljä päivää simuloitua nanosekuntia kohden. Simulaatioita voitiin ajaa rinnakkain, mutta yhtä simulaatiota ei voitu ajaa useammassa säikeessä samanaikaisesti.

2 Menetelmät

Tässä kappaleessa esittelen pintapuolisesti käytetyt simulaatiomenetelmät. Käytetty Orac4.0-ohjelmisto käyttää moniaika-askelista r-RESPA-integraattoria, Nöse-termostaattia lämpötilan säätelyyn ja Particle-Mesh-Ewald-menetelmää Coulombin vuorovaikutuksen mallintamiseen. Tarkemmin nämä on selitetty viitteissä [11, 13, 14, 15].

2.1 Voimakentät

Molekyylejä simuloitaessa atomien välisiä vuorovaikutuksia mallinnetaan klassisella voimakentällä. Teoriassa laskut voisi suorittaa kvanttimekaanisilla menetelmillä, mutta käytännön rajat tietokoneiden laskutehoissa rajoittavat järjestelmien koot satoihin atomeihin. Tätä suurempia kokonaisuuksia simuloitaessa on pakko käyttää erilaisia malleja, jotka ovat huomattavasti nopeampia ja approksimoivat kvanttimekaanisia simulaatioita riittäväällä tarkkuudella. Voimakenttä koostuu useasta osasta, joissa lähinaapurisidoksia kuvataan harmonisella potentiaalilla sidospituuden, -kulman tai kolmen peräkkäisen sidoksen määräämän torsiokulman (kuva 9) funktiona. Harmoninen potentiaalifunktio tulee toisen asteen Taylorin kehitelmästä:

$$V \approx V_0 + V'(x_0) \cdot (x - x_0) + \frac{1}{2} \cdot V''(x_0) \cdot (x - x_0)^2, \quad (1)$$

missä V_0 voidaan määrätä mielivaltaisesti (valitaan $V_0 = 0$) ja $V'(x_0) = 0$, sillä kyseessä on minimi. Näin saadaan

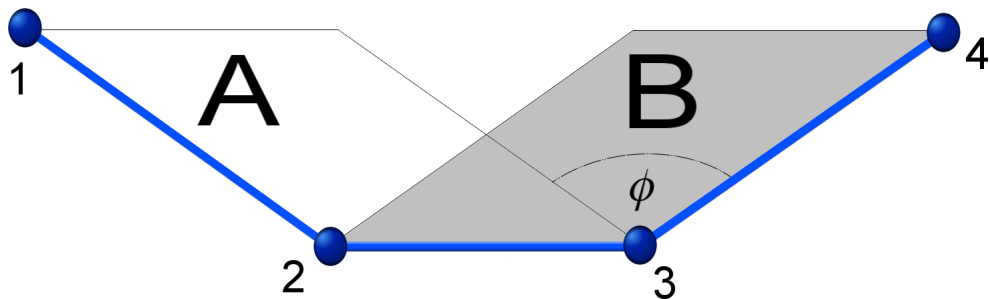
$$V \approx \frac{1}{2} \cdot V''(x_0) \cdot (x - x_0)^2. \quad (2)$$

Toisen asteen termi on nollasta eroava, koska kyseessä on minimi ($V''(x_0) > 0$ lokaalissa minimissä). Tämä approksimaatio kuvaa hyvin atomien käyttäytymistä niin kauan, kun sidokset eivät poikkea tasapainoarvostaan enempää, kuin 10%. Kauempana, kuin kolmen sidoksen päässä toisistaan oleville atomeille voimakenttänä toimii paripotentiali, jossa on atomien varauksista riippuva tavanomainen Coulombin potentiaali ja tähän yhdistettynä Van der Waals-vuorovaikutusta ja Paulin kieltoääntöä kuvaava Lennart-Jones-potentiaali (LJ-potentiaali) [11].

Kaikki halutut atomienväliset kovalentit sidokset kuvataan lähtötiedostoissa ja toisin kuin kvanttimekaanisissa elektronitiheyslaskuissa, sidokset ovat lukittuja. Sidokset määrätään erityisessä topologiatiedostossa, ja simulaation aikana ei siis voi muodostua uusia eikä rikkoutua vanhoja sidoksia. Yhden, kahden tai kolmen sidoksen päässä toisistaan oleviin atomeihin käytetään seuraavanlaisia harmonisia potentiaaleja:

$$V_1(r) = \sum_{\text{sidokset}} K_r (r - r_0)^2, \quad (3)$$

$$V_2(\theta) = \sum_{\text{kulmat}} K_\theta (\theta - \theta_0)^2 \quad (4)$$



Kuva 9: Torsiokulma ϕ atomien 1-4 välillä sekä atomien määräämät tasot A ja B.

ja

$$V_3(\phi) = \sum_{\text{torsiot}} \frac{K_\phi}{2} [1 + \cos(n\phi - \gamma)]. \quad (5)$$

Kullekin potentiaalille on atomityypistä riippuva jousivakionsa, K_r , K_θ ja K_ϕ . Sidos- ja kulmapotentiaaleille tasapainoasemat ovat r_0 ja θ_0 , torsiopotentialin parametreista n :llä muutetaan maksimien lukumäärää ja γ :lla siirretään tarvittaessa origossa sijaitseva maksimi eri paikkaan. Torsiopotentialissa jousivakion lisäksi olevat termit saavat siis tuloksen oskilloimaan kosinifunktion muotoisesti, torsiokulman ϕ funktiona, nollan ja yhden välillä halutulla tavalla.

Torsiokulma ϕ voidaan määrittellä seuraavasti neljälle peräkkäin sitoutuneen atomin ketjulle, jossa atomit olkoon järjestyksessä 1, 2, 3 ja 4. Atomit 1, 2 ja 3 määrittelevät tason A ja atomit 2, 3, ja 4 tason B (kuva 9). Tällöin

$$\cos\phi = \vec{n}_A \cdot \vec{n}_B, \quad (6)$$

missä $\vec{n}_A = \frac{\vec{r}_{12} \times \vec{r}_{23}}{|\vec{r}_{12} \times \vec{r}_{23}|}$ ja $\vec{n}_B = \frac{\vec{r}_{23} \times \vec{r}_{34}}{|\vec{r}_{23} \times \vec{r}_{34}|}$ ovat tasojen A ja B yksikkönormaalivektorit ja \vec{r}_{ij} on atomien i ja j etäisyysvektori.

Lisäksi kovalenttisesti sidotuille atomiketjuille ja -ryhmille voidaan käyttää niin kutsuttuja ristitermi-potentiaaleja, jotka riippuvat lokaalista ympäristöstä eivätkä pelkästään kyseisestä atomiparista. Nämä määrittelevät esimerkiksi, että jos kolmen atomin välinen kulma muuttuu, niin toisen kovalenttin sidoksen tasapainoetäisyys muuttuu kulman muutoksesta riippuen. Näitä käytetään tavallisesti molekyylin värähtelyspektrin mallintamiseen, mutta ne eivät ole käytössä tässä työssä käytetyssä AMBER-voimakentässä. Mahdollisia ristitermejä ovat esimerkiksi sidos-sidos, kulma-sidos ja torsiokulma-kulma.

Atomipareille, jotka ovat toisistaan kauempana kuin kolme sidosta, mutta määritellyn maksimietäisyyden (cutoff radius) sisällä toisistaan, käytetään Coulombin potentiaalia:

$$V_{\text{coul}} = \sum_{i \neq j} \frac{q_i \cdot q_j}{4\pi \cdot \epsilon_0 \cdot \epsilon_c \cdot r_{ij}} \quad (7)$$

ja LJ-potentiaalia:

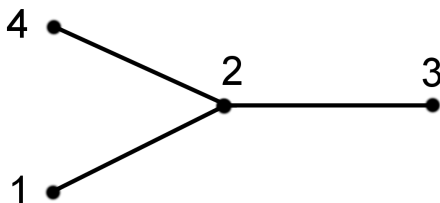
$$V_{LJ} = \sum_{i \neq j} 4\epsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^6 \right), \quad (8)$$

joissa q_i ja q_j ovat atomien varaukset, r_{ij} atomien i ja j välinen etäisyys, ϵ_c on väliaineen suhteellinen permittiivisyys ja lisäksi on parametrejä, jotka riippuvat kyseisen atomiparin atomeiden tyypeistä; ϵ_{ij} ja σ_{ij} . Näistä q_i , q_j , ϵ_{ij} ja σ_{ij} on johdettu pienehköille mallijärjestelmille suoritetuista mittauksista ja kvanttikemiallisista laskuista. Lennart-Jones-potentiaalin lausekkeessa olevat σ_{ij} ja ϵ_{ij} lasketaan kullekin atomiparille atomikohtaisista parametreista siten, että

$$\sigma_{ij} = \frac{1}{2}(\sigma_i + \sigma_j) \quad (9)$$

$$\epsilon_{ij} = \sqrt{\epsilon_i \cdot \epsilon_j} \quad (10)$$

Atomityyppikohtaiset parametrit on huomattavasti helpompaa listata kuin parikohtaiset parametrit kaikille mahdollisille sidospareille mitä atomit voivat muodostaa. Yhtälöistä (9 ja 10) käytetään nimitystä sekoitussäännöt (engl. “nonbonded mixing rules” tai “mixrules”), koska atomien tyyppikohtaiset parametrit ikään kuin sekoitetaan keskenään ja tuloksena on sidokselle sopiva parametrin arvo. Atomeille, joita erottaa täsmälleen neljä kovalenttia sidosta lasketaan Coulombin- ja LJ-potentiaalit, mutta näille voidaan käyttää erityistä pehmennyskerrointa (fudge factor), jonka tarkoituksena on tasoittaa kovalenttisten sidospotentiaalien ja pitkän kantaman potentiaalien vaikutusalueiden rajaa. AMBER-voimakentän mukaisesti simulaatiossa LJ-potentiaalille tämä kerroin oli $\frac{1}{2}$ ja Coulombin potentiaalille $\frac{5}{6}$ [16].



Kuva 10: Tasomallisten sidosten mallintaminen toisen luokan torsiokulma-termillä (engl. “improper torsion”). Tämä torsiokulma lasketaan atomien 1, 2 ja 3 sekä atomien 2, 3 ja 4 määräämien tasojen välisenä kulmana tavallisen torsiokulmatermin tapaan. Tavallisesti minimienergia määritetään kulmille $n \cdot \pi$, $n \in \mathbb{Z}$ yhtälössä (5), jotta saadaan haluttu käyttäytyminen.

Molekyyleissä atomit, jotka muodostavat kovalentteja sidoksia täsmälleen kolmen eri atomin kanssa, muodostavat sidokset usein tasomaisesti. Tämän rakenteen pitämiseen voidaan käyttää yleistä harmonista potentiaalia, jossa energian minimikohdassa sidokset ovat tasossa. Oracin tapauksessa tähän käytetään erityistä toisen luokan torsiokulma-termiä (engl. “improper torsion”), jossa lasketaan “torsiokulma” atomeille, jotka eivät ole suoraan ketjussa vaan kuvan 10 mukaisesti järjestyneinä. Improper torsion-kulma mittaa käytännössä sitä, missä kulmassa atomi 1 (tai 3 tai 4) on muiden kolmen atomin määräämästä

tasosta. Tälle kulmalle käytetään samantyyppistä yhtälön (5) mukaista potentiaalia, kuin tavallisille torsiokulmille. Erona on kuitenkin potentiaalin minimin kulma, joka improper torsionille on 0, jotta atomit pyrkivät tasomaiseen orientaatioon. Tämä saadaan aikaan asettamalla $\gamma = \pm \frac{\pi}{2}$.

2.2 Ewaldin summaus

Alunperin Ewald tarvitsi röntgendiffraktiotutkimuksiinsa vuonna 1912 eksaktin ratkaisun yksittäisen dipolivärähtelijä vaikutukselle tilanteessa, jossa värähtelijöitä oli äärettömässä ja ideaalisessa hilassa. Ongelma käsitteli muunnosta pallo- ja tasoaaltoesitysten välillä. Debyen esiteltyä Ewaldille Riemannin θ -muunnosmetodin sai Ewald ongelman ratkaistua. Vuonna 1921 Ewald alkoi ratkoa jaksollisen hilan sähköstaattista energiaa, joka on tuottanut lopulta juuri sen metodin, jolla simulaatioissa ratkaistaan sähköstaattinen potentiaali [17].

Tämän työn simulaatioissa Coulombin vuorovaikutuksen laskemiseen käytettiin Ewaldin summaus-menetelmää, tarkemmin ottaen “particle-mesh-Ewald”-menetelmää (PME). Tavallisesti sähköstaattisen vuorovaikutuksen laskeminen edes kohtuullisen tarkasti tilanteessa, jossa simuloidaan tuhansista pistevarauksina käsiteltävistä atomeista koostuvaa järjestelmää periodisilla reunaehdoilla olisi hyvin raskas toimenpide, sillä laskennallinen taakka skaalautuu katkaisusäteen (cutoff) kuution mukaan. Coulombin potentiaali on hyvin pitkän matkan vuorovaikutus, joka vaimenee hitaasti. Näinollen katkaisusäteen tulisi olla suuri riittävän tarkan tuloksen saamiseksi. Ewaldin menetelmä antaa huomattavasti pienemmällä laskennallisella taakalla tarkasti sähköstaattisen potentiaalin järjestelmässä, jossa on varattuja hiukkasia jaksollisessa hilassa. Eli esimerkiksi yksinkertaisessa kidehilassa tai vaikkapa periodisilla reunaehdoilla suoritettussa simulaatiossa. Ewaldin summauksessa oletetaan, että alue jossa Ewaldin potentiaalia lasketaan on ympäröity suprajohtavalla materiaalilla ja siten dipolitermi tämän alueen reunalla on nolla (niinsanottu “Tin Foil”-reunaehto). Tämä ei kuitenkaan oleellisesti heikennä menetelmän tarkkuutta.

Summauksen nopeusetu perustuu siihen, että ääretön summa sähköstaattisista energioista $\sum_{i=1}^{\infty} E_i$ muutetaan kahdeksi nopeasti konvergoivaksi summaksi reaali- ja käänteisavaruuksissa:

$$V'_{qd} = \frac{1}{2} \cdot \sum_{ij} q_i q_j \sum_n \frac{1}{|r_{ij}^{\vec{}} + r_n^{\vec{}}|} \cdot \text{erfc}(\alpha \cdot |r_{ij}^{\vec{}} + r_n^{\vec{}}|) \quad (11)$$

$$V_{qr} = \left(\frac{1}{2 \cdot \pi \cdot V} \cdot \sum_{\vec{m} \neq 0}^{\infty} \frac{\exp\left(-\frac{\pi^2 \cdot |\vec{m}|^2}{\alpha^2}\right)}{|\vec{m}|^2} \cdot S(\vec{m}) \cdot S(-\vec{m}) - \frac{\alpha}{\sqrt{\pi}} \cdot \sum_i q_i^2 \right) - V_{intra}, \quad (12)$$

missä

$$S(\vec{m}) = \sum_i^N q_i \cdot e^{2 \cdot \pi \cdot i \cdot \vec{m} \cdot \vec{r}_i} \quad (13)$$

$$V_{intra} = \sum_{ij-excl.} q_i q_j \frac{erf(\alpha \cdot r_{ij})}{r_{ij}} \quad (14)$$

$$erfc(x) = \frac{1}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt \quad (15)$$

$$erf(x) = 1 - erfc(x). \quad (16)$$

Yläpuolella \vec{r}_i on varauksen q_i paikkavektori, $\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$, \vec{r}_n on hilavektori, V on yksikkökopin tilavuus, m on käänteishilavektori ja α on Ewaldin suppenemisparametri. Reaalihilan yhtälössä (11) pilkku tarkoittaa, että potentiaalissa jätetään huomiotta ne atomiparit, joissa atomeita erottaa yksi tai kaksi kovalenttia sidosta. Käänteisvaruuden summan yhtälössä (12) vähennetty V_{intra} on tavallisen hila-avaruuden potentiaalienergia atomipareille, joissa atomeita erottaa yksi tai kaksi kovalenttia sidosta. Yhtälössä (14) summaus käy siis juuri yhtälössä (11) huomiotta jätettyjen atomiparien yli. Yhtälö (16) on virhefunktio.

Tällaisen Ewaldin menetelmän laskennallinen taakka kasvaa kuten N^2 , missä N on järjestelmän hiukkasten lukumäärä. Suurille järjestelmille tämä on siis jo liian raskas ja Ewaldin summan laskemiseen onkin kehitetty algoritmejä, jotka skaalautuvat paremmin järjestelmän koon kasvaessa. Yksi näistä on erityisesti periodisia järjestelmiä silmällä pitäen kehitetty PME. PME:ssä käytetään moniulotteista paloittaista interpolaatiota laskehtaessa $S(\vec{m})$ yhtälössä (13). Yhtälö tulkitaan diskreetiksi Fourier-muunnokseksi (FT, Fourier transform), jossa hilapisteet ovat varattujen hiukkasten paikat. Tätä voidaan approksimoida eri menetelmillä, joissa lasketaan FT tasavälisissä hilapisteissä, jolloin voidaan käyttää niinsanottuja Fast Fourier transform -algoritmejä (FFT), jotka ovat laskennallisesti huomattavasti kevyempiä. Yleensä approksimaatioissa pistemäisiä varauksia käsitellään varausjakaumana (Gaussin jakauman muoto) pistettä ympäröivällä alueella.

2.3 MD-algoritmit

2.3.1 Potentiaalijako moninkertaista aika-askelta käyttävissä simulaatioissa

Molekyylidynamiikkasimulaatioissa voidaan käyttää useita erilaisia iteraattorialgoritmejä. MD-algoritmeissä tarkoituksena on mallintaa järjestelmän käyttäytymistä ajan funktiona ja ratkaista Newtonin liikeyhtälöt tietyin aikavälein. Iteraattoreissa käytetään äärellistä aika-askelta ja samat laskut toistetaan joka aika-askeleen välein. Tavallisesti kaikki voimat, nopeudet ja paikat pidetään vakioina aika-askeleen ajan ja päivitetään vastaamaan uusia arvoja aika-askeleiden välissä. Poikkeuksena tästä ovat tietyt algoritmit, joissa käytetään niinkutsuttuja puoliaskel-nopeuksia tai -paikkoja. Näistä käytetään nimityksiä ennakointi-korjaus-algoritmi (predictor-corrector-algorithm). Yksinkertaisimmillaan iteraattori laskee atomeihin vaikuttavat voimat aika-askeleen alussa, ja uudet paikat ja nopeuden aika-askeleen lopussa eli seuraavan askeleen alussa. Lyhyemmin: ratkaistaan $\vec{F}_i(t)$ potentiaaleista jokaiselle atomille i , tämän jälkeen lasketaan uudet paikat ja nopeudet seu-

raavalla tavalla

$$\vec{r}_i(t + \Delta t) = \vec{r}_i(t) + \vec{v}_i(t) \cdot \Delta t \quad (17)$$

$$\vec{v}_i(t + \Delta t) = \vec{v}_i(t) + \frac{\Delta t \cdot \vec{F}_i(t)}{m_i}, \quad (18)$$

jokaiselle i ja lopuksi päivitetään voimakenttäpotentiaalit uusien atomien sijainteja vastaaviksi. Tällainen yksinkertainen algoritmi tarvitsee toimiakseen pienen aika-askleen. Tavallisesti aika-askleen tulee olla yhtä kertaluokkaa pienempi kuin lyhimmän liikkeen jaksonaika. Nopeimmat värähtelyt biologisissa järjestelmissä esiintyvät tavallisesti vetyatomien ja raskaampien atomien välisissä sidoksissa, joille sopiva aika-askel on 1-2 fs. Jos aika-askel on liian pitkä, niin energian säilymisehto ei toteudu ja simulaatiosta tulee epäluotettavaa dataa tai se saattaa jopa kaatua koska atomit liikkuvat liian pitkiä matkoja aika-askleen aikana ja saattavat osua päällekkäin. Tällöin epärealistisen suuri repulsiivinen voima heittää atomit erilleen ja repii koko simuloitavan järjestelmän kappaleiksi. Kaikille käytetyille algoritmeille on olemassa tietty aika-askel, jota pienemmällä aika-askleen arvoilla algoritmi on stabiili. Tämä tarkoittaa sitä, että energian säilyminen toimii ja simulaatiosta tulee luotettavaa dataa. Aika-askelta voi aina pienentää, mutta jatkuvalla pienentämisellä ei saavuteta merkittävää hyötyä. Simulaation vaatima prosessori-aika on kääntäen verrannollinen aika-askleen pituuteen eli aika-askleen puolittaminen kaksinkertaistaa simulaation suorittamiseen kuluvan ajan.

Kehittyneempiin simulaatioalgoritmeihin luetaan predictor-corrector-algoritmit, verlettyyppiset algoritmit sekä erilaiset moninkertaista aika-askelta käyttävät algoritmit. Verlet-algoritmin eri modifikaatiot muodostavat yhdessä ehkä eniten käytetyn algoritmityyppin. Näitä kutsutaan lyhyesti Verlet-algoritmeiksi.

Useimmat algoritmit, kuten myös Verlet-algoritmi lähtee oletuksesta, että atomien koordinaatteja, nopeuksia ja kiihtyvyyksiä voidaan approksimoida Taylorin sarjalla. Taylorin sarjasta atomin paikalle \vec{r} ajan hetkellä t saadaan seuraavaa:

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \cdot \vec{v}(t) + \frac{1}{2} \cdot \Delta t^2 \cdot \vec{a}(t) + \dots \quad (19)$$

$$\vec{r}(t - \Delta t) = \vec{r}(t) - \Delta t \cdot \vec{v}(t) + \frac{1}{2} \cdot \Delta t^2 \cdot \vec{a}(t) + \dots \quad (20)$$

Loput termit jätetään huomiotta ja lasketaan ylläolevat yhteen:

$$\vec{r}(t + \Delta t) = 2 \cdot \vec{r}(t) - \vec{r}(t - \Delta t) + \Delta t^2 \cdot \vec{a}(t). \quad (21)$$

Nopeudet lasketaan Verlet-algoritmiä käyttävässä ohjelmassa tavallisesti kahden askeleen aikana tapahtuneena paikan muutoksena:

$$\vec{v}(t) = \frac{\vec{r}(t + \Delta t) - \vec{r}(t - \Delta t)}{2 \cdot \Delta t}. \quad (22)$$

Yleisesti Verlet-algoritmeille ominaista on, että kun niissä lasketaan seuraavan aika-askleen nopeuksia ja paikkoja, käytetään tämänhetkisten arvojen lisäksi myös edellisiä

hyväksi. Tämä parantaa algoritmin luotettavuutta, mutta hintana on se, että nopeudet pitää tarvittaessa laskea erikseen. Lisäksi laskettaessa seuraavan askeleen atomien paikkoja lisätään (yhtälön (21) mukaisesti) suureen lukuun $2 \cdot \vec{r}(t) - \vec{r}(t - \Delta t)$ pieni luku $\Delta t^2 \cdot \vec{a}(t)$, joka saattaa huonontaa tarkkuutta.

Toinen Verlet-algoritmi on niinkutsuttu “Leap-frog”-algoritmi. Algoritmi muistuttaa hyvin paljon tavallista Verlet:iä, mutta siinä on nopeudet eksplisiittisesti näkyvissä. Nopeudet tosin “loikkaavat” tasa-aika-askelien yli ja ne lasketaan puolikkailla aika-askelilla. Paikat lasketaan tasa-aika-askelilla:

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \cdot \vec{v}(t + \frac{1}{2} \cdot \Delta t) \quad (23)$$

$$\vec{v}(t + \frac{1}{2} \cdot \Delta t) = \vec{v}(t - \frac{1}{2} \cdot \Delta t) + \Delta t \cdot \vec{a}(t). \quad (24)$$

Käytännössä simulaation alussa $\vec{v}(t + \frac{1}{2} \cdot \Delta t)$ lasketaan ensin. Leap-frog on hieman parempi, kuin yhtälön (21) mukainen algoritmi, sillä se sisältää nopeudet eksplisiittisesti lausekkeissa ja siinä ei ole yhteenlaskua isojen ja pienien lukujen välillä, mikä johtaa tarkempiin tuloksiin. Toisaalta atomien paikat ja nopeudet lasketaan eri ajanhetkille, jolloin laskettaessa kokonaisenergiaa pitää kineettinen energia laskea kahden arvon keskiarvona $E_K(t) = \frac{1}{2} \cdot (E_K(t - \frac{1}{2}\Delta t) + E_K(t + \frac{1}{2}\Delta t))$, sillä nopeudet lasketaan näille puoliaika-askelille. Tarkkaa arvoa molemmille energioille (kineettinen ja potentiaali-) ei voi laskea samalle ajanhetkelle. Tämä tarkoittaa sitä, että kokonaisenergiaa ei voida tarkasti laskea ollenkaan.

Jotkut algoritmit, kuten Oracissa käytetty r-RESPA (reversible reference system propagator algorithm) käyttävät useata eri kokoista aika-askelta. Moniaika-askelisen iteraattorin perusajatuksena on jakaa järjestelmä osiin tarvittavan aikaskaalan perusteella. Jos oletetaan, että simuloitun järjestelmän potentiaali voidaan jakaa osiin

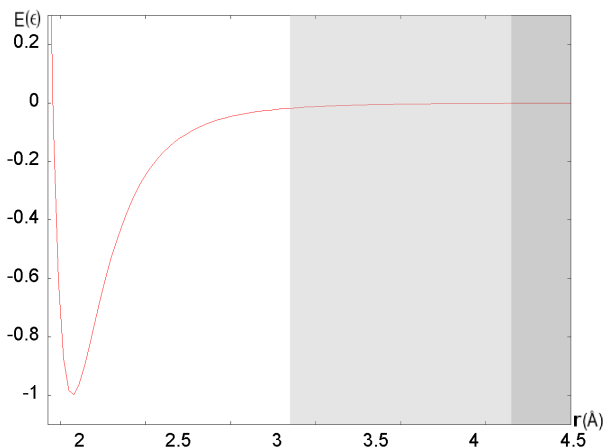
$$V = V_0 + V_1 + \dots + V_n, \quad (25)$$

siten että potentiaaliosien aiheuttamille voimille $F_i = |\partial V_i / \partial x|$ ja niiden aikaderivaatoille $\dot{F}_i^2 = |dF_i/dt|$ pätee seuraavaa

$$F_0^2 \gg F_1^2 \gg \dots \gg F_n^2 \quad (26)$$

$$\dot{F}_0^2 \gg \dot{F}_1^2 \gg \dots \gg \dot{F}_n^2, \quad (27)$$

niin näille eri tahtiin muuttuville voimille voidaan käyttää kullekin omaa aika-askeltansa. Käytännössä mikään reaalinen järjestelmä ei täysin toteuta näitä ehtoja, koska eri potentiaaliosien aikaskaalat menevät osin päällekkäin. Järjestelmä voidaan kuitenkin jakaa osiin potentiaaliosien mukaan ja käyttää osille eri aika-askeleita.



Kuva 11: Lennard-Jones-potentiaalin jako eripituisilla aika-askelilla laskettaviin “kuoriin”. Pienin kuori on $0-4,7$ Å, keskimäinen vaalealla harmaalla merkitty kuori on $4,7-7,3$ Å ja viimeinen kuori tummalla harmaalla merkittynä on säteen arvoilla $7,3-9,7$ Å. Kyseessä on juuri tässä työssä käytetty aika-askelen jako. Kuvassa energian yksikkönä epsilon, joka on simulaatiossa käytetty atomikohtainen parametri.

Aika-askleet eivät välttämättä ole optimaalisia suurimmalle osalle vapausasteita vaan askelen koko valitaan nopeimman värähtelytaajuuden mukaan. Nopeasti värähtelevät vapausasteet toimivat lyhyellä aika-askeleella, mutta hitaammin liikkuvia voidaan simuloida pidemmällä. Vaikka aika-askelen valinta tehdään niinsanotusti “varman päälle”, vähentää menetelmä tarvittavaa laskenta-aikaa kertoimella 4-40. Käytännössä tämä toteutetaan siten, että hitaammin liikkuvat vapausasteet pidetään jäädytettynä suurimman osan ajasta ja niille suoritetaan MD-askelen laskutoimitukset vain joka n :s askel, missä n on tyypillisesti jokin kokonaisluku välillä 2-12. Lyhintä aika-askelta pidemmät askelten pituudet ovat siis pykälää lyhemmän askelen pituuden monikertoja. Tässä työssä simulaatioissa käytettiin molekyylienvälisen Lennard-Jones- ja Coulombin potentiaalin jakoon eräänlaisia “kuoria” kuvan 11 tapaan. Atomien välisestä etäisyydestä riippui, minkä suuruista aika-askelta vuorovaikutusten laskemiseen käytettiin. Kuvassa on sama jako kuin 1KT7:n simulaatiossa. Pienin kuori välillä $0-4,7$ Å tarkoittaa että tällä alueella käytettiin lyhintä aika askelta ($\frac{4}{3}$ ns). Suurimman kuoren alueella $7,3-9,7$ Å, joka jatkuu kuvan ulkopuolelle cutoff-säteelle asti käytettiin pisintä aika-askelta (8 ns). Keskimäisen kuoren aika-askel oli $\frac{8}{3}$ ns. Kovalenttien sidosten aika-askleet olivat $\frac{2}{3}$ ns ja $\frac{1}{3}$ ns. Dig16A:n simulaatiossa kuorien ylärajat olivat $4,2$ Å, $7,4$ Å ja $9,7$ Å. Aika-askleet kuorille olivat kaikissa simulaatioissa samat.

Orac4.0 käyttää velocity Verlet-algoritmiä, jota suoritetaan useissa sisäkkäisissä silmuikoissa - yksi jokaiselle aika-askelen koolle. Lisäksi Oracissa voi jäädyttää kokonaan joi-takin vapausasteita SHAKE-algoritmillä. Tavallisesti kohteena on vetyatomin kovalentti sidos jonkin raskaamman atomin kanssa (yleensä C , N tai O). Tilanne vastaa sitä, että verrattuna raskaamman atomin liikkeeseen vetyatomin oletetaan liikkuvan niin nopeasti tasapainoasemaan, että se on siellä aina, kun raskaammalle atomille tapahtuu jotakin. SHAKE-algoritmi toimii hieman takaperoisesti. Siinä suoritetaan ensin tavallinen MD-askel

järjestelmälle, jonka jälkeen SHAKE laskee jäädytetyn vapausasteen senhetkisen aseman ja tasapainotilan välisen eron avulla korjausvoiman, joka lisätään edellisen askeleen laskuihin ja askel lasketaan uudelleen. Tämän jälkeen vapausaste on tasapainoasemassa ja simulaatio etenee seuraavaan MD-askeleeseen.

2.3.2 Symplektinen ja aikareversiibeli operaattori

Hamiltonin mekaniikan tarkat ratkaisut ovat aina symplektisiä ja aikareversiibeileitä. Tarkoittaen sitä, että laskuja voidaan suorittaa ajassa eteen- tai taaksepäin kun tiedetään jokin järjestelmän tila ja aina saadaan identtiset tulokset riippumatta mitä kautta tuloksiin on päästy. Käytännön syistä simuloimassa pitää käyttää kuitenkin paloittaista integrointia eli äärellistä aika-askelta. Äärellisellä määrällä aika-askelia saatu ratkaisu atomien paikoille ja nopeuksille voi olla myös symplektinen ja ajan suhteen reversiibeli, mutta aina näin ei ole. Liouville-formalismi pohjaa Hamiltonin mekaniikkaan ja sen avulla voidaan rakentaa symplektinen ja reversiibeli iteraattori. Tällaista iteraattoria pidetään luotettavampana kuin sellaista iteraattoria, jolla ei ole näitä ominaisuuksia [18, 19, 20, 21].

Oletetaan, että järjestelmässä on n yleistettyä paikkakoordinaattia q ja n yleistettyä liikemääräkoordinaattia p . Hamiltonin operaattorilla H esitetyt liikeyhtälöt ovat tällöin

$$\dot{q}_i = \frac{\partial H}{\partial p_i} \quad (28)$$

$$\dot{p}_i = -\frac{\partial H}{\partial q_i}, \quad (29)$$

missä $i = 1, 2, 3, \dots, n$. Matriisimerkinnällä tämä voidaan esittää sarakematriisina, jossa on $2n$ alkioita siten, että

$$x = \begin{pmatrix} q \\ p \end{pmatrix} \quad (30)$$

$$\dot{x} = J \frac{\partial H}{\partial x} \quad (31)$$

$$J = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad (32)$$

missä J on $2n \times 2n$ -matriisi, 1 on $n \times n$ yksikkömatriisi ja 0 on $n \times n$ matriisi nollia. Määritellään kanoninen muuttujanvaihdos $x \equiv \{q, p\} \Rightarrow y \equiv \{Q, P\}$, jolloin

$$y = y(x). \quad (33)$$

Tälle pätee, että funktio $H(x)$ on Hamiltonin funktio uudessa y -kannassa eli Hamiltonin liikeyhtälöillä on sama muoto, kuin x -kannassa, siis

$$\dot{y} = J \frac{\partial H}{\partial y}. \quad (34)$$

Derivoidaan yhtälö (33) ajan suhteen käyttäen ketjusääntöä, jolloin saadaan

$$\dot{y} = MJM^T \frac{\partial H}{\partial y}, \quad (35)$$

missä M on Jacobin matriisi ja M^T sen transpoosi. Yhtälöistä (34) ja (35) nähdään, että tehty muuttujanvaihto oli kanoninen jos ja vain jos

$$MJM^T = J. \quad (36)$$

Tämä on niisanottu symplektisyyssehto, jota tullaan käyttämään myöhemmin. Tällä voidaan tarkistaa kätevästi onko mielivaltainen muunnos kanoninen vai ei.

Seuraavaksi johdetaan Verlet-algoritmit: velocity Verlet ja position Verlet. Position Verlet tunnetaan myös "leap frog"-nimellä ja molemmat algoritmit ovat hyvin vakaita ja myöskin symplektisiä. Juuri symplektisyyttä pidetään niiden vakauden pääsyyinä [20, 21, 22, 23].

Lähdetään liikkeelle jostakin ominaisuudesta, joka riippuu implisiittisesti ajasta $(p, q) \equiv x$:n kautta. Nyt

$$\frac{dA(p, q)}{dt} = \sum_{q,p} \left(\dot{q} \frac{\partial A}{\partial p} + \dot{p} \frac{\partial A}{\partial q} \right) = \sum_{q,p} \left(\frac{\partial H}{\partial p} \frac{\partial A}{\partial q} - \frac{\partial H}{\partial q} \frac{\partial A}{\partial p} \right) = iLA, \quad (37)$$

missä summa q :n ja p :n yli käy läpi kaikki vapausasteet. Tässä L on Liouvillean operaattori, joka siis määritellään

$$iL = \sum_{q,p} \left(\dot{q} \frac{\partial A}{\partial p} + \dot{p} \frac{\partial A}{\partial q} \right) = \sum_{q,p} \left(\frac{\partial H}{\partial p} \frac{\partial}{\partial q} - \frac{\partial H}{\partial q} \frac{\partial}{\partial p} \right). \quad (38)$$

Integroidaan yhtälö (37) puolittain, jolloin saadaan

$$A(t) = e^{i \cdot L \cdot t} \cdot A(0). \quad (39)$$

Jos A on tilavektori, voidaan yhtälöä (39) käyttää integroimaan Hamiltonin yhtälöt

$$\begin{bmatrix} q(t) \\ p(t) \end{bmatrix} = e^{i \cdot L \cdot t} \begin{bmatrix} q(0) \\ p(0) \end{bmatrix}.$$

On syytä huomata, että tämä on Hamiltonin likeyhtälöiden formaali ratkaisu ja eksponenttioperaattorilla $e^{i \cdot L \cdot t}$ kerrottuna alkutilan tilavektori määrittää järjestelmän tilan myöhemmellä ajanhetkellä t . Tämä operaattori toteuttaa yhtälön (36) symplektisyys ehdon ja lisäksi se on unitaarinen. Nämä ominaisuudet yhdessä tarkoittavat sitä, että järjestelmän trajektorit on täsmälleen aikareversiibeli.

Käytännön simulaatiot tarvitsevat kuitenkin integraattoriin äärellisen aika-askeleen. Tämä toteutetaan jakamalla aika n :ään Δt :n mittaiseen palaan: $\Delta t = \frac{t}{n}$. Määritellään nyt diskreetti aikapropagaattori $e^{i \cdot L \cdot \Delta t}$ siten, että

$$e^{i \cdot L \cdot t} = \left[e^{i \cdot L \cdot \frac{t}{n}} \right]^n \quad (40)$$

$$e^{i \cdot L \cdot \Delta t} = \exp \left(\sum_{q,p} \left(\dot{q} \frac{\partial}{\partial p} + \dot{p} \frac{\partial}{\partial q} \right) \Delta t \right). \quad (41)$$

Jotta diskreettiä operaattoria voidaan käyttää pitäisi periaatteessa tuntea potentiaalin V kaikki derivaatat. Tämä nähdään jos tutkitaan operaattorin $e^{iL\cdot\Delta t}$ Taylorin kehitelmää. Siinä $\dot{q}\frac{\partial}{\partial q}$ ei kommutoi termin $-\frac{\partial V}{\partial q}\frac{\partial}{\partial p}$ kanssa silloin, kun paikka ja liikemäärä ovat samasta vapausasteesta. Tarvitaan siis symplektisyyden ja reversiibeliyden säilyttävä approksimaatio, joka antaa propagaattorille äärellisen aika-askeleen. Tällöin propagaattorin käyttö käytännön simulaatioissa on mahdollista.

Kahdelle lineaariselle operaattorille A ja B pätee Trotterin yhtälö [24]:

$$e^{(A+B)t} = \lim_{n \rightarrow \infty} \left(e^{A \cdot \frac{t}{n}} \cdot e^{B \cdot \frac{t}{n}} \right)^n. \quad (42)$$

Yhtälössä (41) on oikealla puolen samankaltainen rakenne, kuin yhtälössä (42), sillä tarpeeksi pienelle Δt voidaan approksimoida

$$e^{iL\cdot\Delta t} = e^{(\dot{q}\frac{\partial}{\partial q} + \dot{p}\frac{\partial}{\partial p})\Delta t} \simeq e^{\dot{q}\frac{\partial}{\partial q}\cdot\Delta t} \cdot e^{\dot{p}\frac{\partial}{\partial p}\cdot\Delta t} + O(\Delta t^2), \quad (43)$$

josta on jätetty summa $p:n$ ja $q:n$ yli pois jottei yhtälöstä tulisi liian sekava. Yhtälö (43) pätee täsmälleen rajalla $\Delta t \rightarrow 0$, ja sen tarkkuus on ensimmäistä astetta äärellisille aika-askeleille. Se myös toteuttaa yhtälön (36), koska se on tulo kahdesta sympleettisestä operaattorista. Propagaattori ei ole unitaarinen eli se ei ole aikareversiibeli, koska kaksi eksponenttioperaattoria eivät kommutoi. Tämä voidaan korjata puolittamalla aika-askel ja käyttämällä approksimaatiota

$$e^{(A+B)\cdot t} \simeq e^{A\cdot\frac{t}{2}} \cdot e^{B\cdot\frac{t}{2}} \cdot e^{B\cdot\frac{t}{2}} \cdot e^{A\cdot\frac{t}{2}} = e^{A\cdot\frac{t}{2}} \cdot e^{B\cdot t} \cdot e^{A\cdot\frac{t}{2}}. \quad (44)$$

Saatu propagaattori on aikareversiibeli, ja sen tarkkuus on toista astetta [25]. Huomataan, että unitaarinen operaattori parantaa siis samalla tarkkuutta [25, 26]. Näin yhtälöstä (41) saadaan

$$e^{iL\cdot\Delta t} = e^{(\dot{q}\frac{\partial}{\partial q} + \dot{p}\frac{\partial}{\partial p})\cdot\Delta t} \simeq e^{\dot{p}\frac{\partial}{\partial p}\frac{\Delta t}{2}} \cdot e^{\dot{q}\frac{\partial}{\partial q}\Delta t} \cdot e^{\dot{p}\frac{\partial}{\partial p}\frac{\Delta t}{2}} + O(\Delta t^2). \quad (45)$$

Operaattorin $e^{a\frac{\partial}{\partial x}}$ toiminta mielivaltaiseen funktioon $f(x)$ vastaa $f(x):n$ Taylorin kehitelmään pisteen x ympäristössä kohdassa $x + a$ eli

$$e^{a\frac{\partial}{\partial x}} f(x) = f(x + a). \quad (46)$$

Yhtälön (46) mukaan käyttämällä Hermiittistä operaattoria (45) alkutilan tilavektoriin $\begin{bmatrix} q(0) \\ p(0) \end{bmatrix}$ saadaan laskettua koordinaatin ja liikemäärän arvot myöhemmällä ajanhetkellä Δt . Näin saatu algoritmi on ekvivalentti velocity Verlet'n kanssa:

$$\begin{aligned} p\left(\frac{\Delta t}{2}\right) &= p(0) + F(0) \cdot \frac{\Delta t}{2} \\ q(\Delta t) &= q(0) + \left(\frac{p\left(\frac{\Delta t}{2}\right)}{m}\right) \cdot \Delta t \\ p(\Delta t) &= p\left(\frac{\Delta t}{2}\right) + F(\Delta t) \cdot \frac{\Delta t}{2}. \end{aligned} \quad (47)$$

Jokainen velocity Verlet'n kolmesta muunnoksesta on symplektinen yhtälön (36) ehdon mukaisesti ja kolmen symplektisen operaattorin yhdistelmä on myös symplektinen. Koko algoritmi on aikareversiibeli, koska siinä käytetty yhtälön (45) propagaattori on unitaarinen. Lisäksi, jos operaattorit $\dot{q} \frac{\partial}{\partial q}$ ja $-\frac{\partial V}{\partial q} \frac{\partial}{\partial p}$ vaihdetaan keskenään saadaan algoritmi, joka on ekvivalentti niinkutsutun "leap frog"-algoritmin kanssa [27],

$$\begin{aligned} q\left(\frac{\Delta t}{2}\right) &= q(0) + \frac{p(0)}{m} \cdot \frac{\Delta t}{2} \\ p(\Delta t) &= p(0) + F\left(q\left(\frac{\Delta t}{2}\right)\right) \cdot \Delta t \\ q(\Delta t) &= q\left(\frac{\Delta t}{2}\right) + \frac{p(\Delta t)}{m} \cdot \frac{\Delta t}{2}, \end{aligned} \tag{48}$$

tämä tunnetaan myös nimellä "position Verlet" [22], koska se näyttää hyvin analogiselta velocity Verlet'n kanssa. Itse asiassa aika-akselin siirto $\frac{\Delta t}{2}$:lla tekisi algoritmeista täysin identtiset. Puoliaskleet eivät tosin ole formaalisti määriteltyjä [28], koska ne ovat yhtälön (44) oikealla puolella täyden askeleen approksimaatiossa. Tämän vuoksi velocity ja position Verlet-algoritmien trajektorit eivät ole numeerisesti identtiset.

2.3.3 Lämpötilan ja paineen kontrolloiminen termostaatin ja barostaatin avulla

Simulaatiossa käytettiin termostaattia pitämään simulaation lämpötila vakiona NVT -ajoissa eli suurimmassa osassa ajoja ja lisäksi muutamassa ajossa barostaattia pitämään paine vakiona, kun simulaatiota ajettiin NPT -ajona. Käytännössä tällainen termostaatti toimii hyvin, lämpötilan fluktuaatiot suoritetuissa simulaatioissa olivat esimerkiksi 300 K lämpötilassa ± 5 K. Lämpötila lasketaan MD-simulaatioissa tavallisesti hiukkasten kokonaisliike-energiasta. Orac käyttää Nosé-barostaattia ja -termostaattia. Käsittelen tässä barostaatin toimintaa, sillä termostaatin toiminnallisuuden saa määräämällä yhden barostaatin virtuaalisista muuttujista äärettömäksi [11]. Perusideana on laajentaa faasiavaruutta virtuaalisilla vapausasteilla, jotka vastaavat makroskooppisia termodynaamisia muuttujia, kuten painetta tai lämpötilaa [29, 30, 31, 32, 33, 34, 35]. Sopivilla muutoksilla järjestelmän Lagrangen operaattoriin saadaan virtuaaliset vapausasteen tuottamaan haluttua makroskooppista muuttujaa vastaava trajektoria simulaatiota ajettaessa. Tällä tekniikalla voidaan simuloida molekyyliä eri ensembleissa sen sijaan, että kaikki simulaatiot olisivat mikrokanonisessa ensemblessa, jossa energia, tilavuus ja hiukkasluku säilyvät. Simulaatiota voidaan ajaa NPH ensemblessä isotrooppisella tai anisotrooppisella paineella, kanonisessa NVT ensemblessä tai NPT ensemblessä, jossa paine ja lämpötila pidetään kumpikin halutulla tarkkuudella vakioina.

Simuloitava järjestelmä jaetaan osiin siten, että yksi molekyyli on oma osansa. Suuremmille kokonaisuuksille saattaa olla hyödyllistä määrittää useampi osa samaan molekyyliin. Atomeille määritellään skaalatut koordinaatit suhteessa oman osansa massakeskipisteeseen. Massakeskipisteille on omat koordinaattinsa. Lisäksi määritellään kaksi virtuaalista para-

metria (“massaa”) barostaatille W ja termostaatille Q . Ensin otetaan käyttöön muuttujien skaalaus:

$$\begin{aligned} r_{ik\alpha} &= R_{i\alpha} + l_{ik\alpha} = \sum_{\beta} h_{\alpha\beta} \cdot S_{i\beta} + l_{ik\alpha} \\ \dot{R}'_{i\alpha} &= \dot{R}_{i\alpha} \cdot s \\ \dot{l}'_{ik\alpha} &= \dot{l}_{ik\alpha} \cdot s, \end{aligned} \quad (49)$$

missä indeksi i viittaa molekyyliin, k atomiin molekyylissä ja α viittaa koordinaattiakselin suuntaiseen komponenttiin. $R_{i\alpha}$ ovat massakeskipisteen koordinaatit ja $S_{i\beta}$ ovat skaalatut koordinaatit i :nnen molekyylin massakeskipisteelle. $l_{ik\alpha}$ on i :nnen molekyylin k :nnen atomin koordinaatti koordinaatistossa, jonka akselit ovat yhdensuuntaisia laboratoriokoordinaatiston kanssa, mutta origo on senhetkisessä molekyylin massakeskipisteessä. Matriisi h viittaa paineeseen ja skalaari s lämpötilaan. h :n sarakkeet ovat simulaatiokopin reunojen karteesiset komponentit suhteessa kiinnitettyyn koordinaatistoon, ja simulaatiokopin tilavuus on $\Omega = \det(h)$. h :n elementit määräävät simulaatiokopin koon ja muodon, jotka siis voivat muuttua simulaation aikana. Virtuaalisilla vapausasteilla laajennettu Lagrangen operaattori on

$$\begin{aligned} L &= \frac{1}{2} \sum_i^N M_i \cdot s^2 \cdot \dot{S}_i^T \cdot h^T \cdot h \cdot \dot{S}_i + \frac{1}{2} \sum_{ik} m_{ik} \cdot s^2 \dot{l}_{ik}^T \cdot \dot{l}_{ik} + \frac{1}{2} \cdot W \cdot s^2 \cdot \text{tr}(\dot{h}^T \cdot \dot{h}) \\ &+ \frac{1}{2} \cdot Q \cdot \dot{s}^2 - V - P_{ext} \cdot \Omega - \frac{g}{\beta} \cdot \ln(s), \end{aligned} \quad (50)$$

missä N on molekyylien (tai suuremmissa simulaatioissa ryhmien) lukumäärä, P_{ext} on ulkoinen paine, ja W ja Q ovat edellä mainitut simulaatiokohtaisesti valitut parametrit. Muuttuja g riippuu kaikkien vapausasteiden lukumäärästä. Termodynaamisia muuttujia P ja T , mallinnetaan erityisillä virtuaalisilla potentiaaleilla V_P ja V_T , jotka esiintyvät L :n lausekkeessakin,

$$\begin{aligned} V_P &= P \cdot \det(h) \\ V_T &= \frac{g \cdot \ln(s)}{\beta}. \end{aligned} \quad (51)$$

Tarkempaa analyysiä tämän muokatun Lagrangen operaattorin ominaisuuksista on viitteissä [15, 34]. Lisäksi virtuaalisia muuttujia Q ja W on tutkittu viitteissä [36, 37, 38].

Hamiltonin operaattoriin saa Lagrangen operaattorista Legendren muunnoksella

$$H(p, q) = \sum \dot{q} \cdot p - L(q, \dot{q}). \quad (52)$$

Sitä varten pitää laskea konjugoidut liikemäärät muuttujille $S_{i\alpha}$, $l_{ik\alpha}$, $h_{\alpha\beta}$ ja s :

$$\begin{aligned} T_i &= M_i \cdot G \cdot s^2 \cdot \dot{S}_i \\ p_{ik} &= m_{ik} \cdot s^2 \cdot \dot{l}_{ik} \\ P_h &= s^2 \cdot W \cdot \dot{h} \\ p_s &= Q \cdot \dot{s}, \end{aligned} \quad (53)$$

missä $G = h^t h$ on symmetrinen matriisi. Hamiltonin operaattori saa nyt laskettua:

$$\begin{aligned}
H &= \frac{1}{2} \sum_i^N \frac{\dot{T}_i \cdot G^{-1} \cdot \dot{T}}{M \cdot s^2} + \frac{1}{2} \sum_{ik} \frac{p_{ik}^t \cdot P_{ik}}{m_{ik} \cdot s^2} + \frac{1}{2} \frac{\text{tr}(P_h^t \cdot P_h)}{s^2 \cdot W} \\
&+ \frac{p_s^2}{2 \cdot Q} + V + P \cdot \Omega + \frac{g \cdot \ln(s)}{\beta}.
\end{aligned} \tag{54}$$

Tässä muuttujina ovat skaalatut ja virtuaaliset muuttujat. Vaikka operaattori on todellakin Hamiltonin operaattori, josta vastaavat liikeyhtälöt saa käyttämällä yhtälöä (31) muuttujalla $x \equiv (S_{i\alpha}, l_{ik\alpha}, h_{\alpha\beta}, s, T_{i\alpha}, p_{ik\alpha}, \pi_{\alpha\beta}, p_s)$, niin on huomattavasti kätevämpää operoida reaalilla muuttujilla, sillä virtuaalisilla muuttujilla on tiettyjä hankaluuksia, esimerkiksi aika ei kulje vakionopeutta [29, 36]. Reaaliin muuttujiin pääsee käsiksi seuraavasti:

$$\begin{aligned}
T_{i\alpha} &\rightarrow \frac{T_{i\alpha}}{s} \\
p_{ik\alpha} &\rightarrow \frac{p_{ik\alpha}}{s} \\
(P_h)_{\alpha\beta} &\rightarrow \frac{(P_h)_{\alpha\beta}}{s} \\
p_s &\rightarrow \frac{p_s}{s} \\
t &\rightarrow \frac{t}{s}.
\end{aligned} \tag{55}$$

ja lisäksi otetaan käyttöön massakeskipisteliikemäärä

$$P_i = G^{-1} \cdot T_i, \tag{56}$$

jolla voidaan laskea simulaatiossa atomien nopeudet ilman tietoa h :sta. Tällä tavoin saadaan aikaan Verlet-tyyppinen jako nopeuksien ja koordinaattien välille [38]. Nopeudet saadaan yhtälöllä

$$\dot{S}_i = \frac{P_i}{M}. \tag{57}$$

Yhtälöstä (54) voidaan johtaa propagaattoriksi samankaltainen eksponenttifunktio, kuin (45). Johto on kuitenkin monimutkainen ja löytyy kokonaisuudessaan viitteestä [15].

2.4 Simulaatio käytännössä

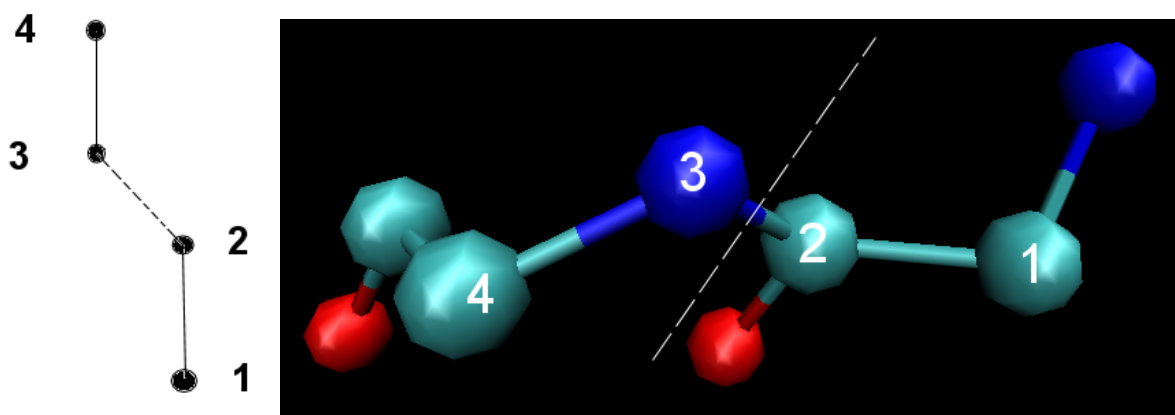
Simulaation suorittamiseen tarvitaan siis lähtörakenne, joka sisältää atomien koordinaatit ja tiedot molekyylin topologiasta sekä parametrit voimakenttään. Proteiinien kolmiulotteista rakennetta on selvitetty laajalti kokeellisesti, ja tiedot on koottu Protein Data Bank-tietokantaan [12]. Sieltä saa vapaasti käyttöönsä koordinaattitiedostoja proteiineille PDB-formaatissa. Joskus koordinaattitiedostoista puuttuu osia proteiinista, koska proteiineissa voi olla melko vapaasti heiluvia osia, jotka eivät mittauksissa asetu aloilleen kiderakenteessa, ja niin näille osille ei saada luotettavaa mittaustulosta. DigA16-proteiini on

yksi näistä ja rakenteesta puuttuikin aminohappoketjun alkupäästä neljä, väliltä silmukasta viisi ja loppupäästä häntää 18 aminohappoa. Silmukasta puuttuvat aminohapot oli jo lisätty käyttäen Refmac-ohjelmistoa [39], ja hännistä puuttuvat osat lisäsimme itse. Tässä vaiheessa on hyvä korostaa, että vaikka koordinaatit puuttuivat, niin proteiinin aminohappoketjusta tiedetään silti siihen kuuluvat aminohapot sekä niiden järjestys. Piti siis vain asetella palaset paikoilleen järjestyksessä.

Aminohappojen lisäämiseen tehtiin yksinkertainen ohjelmaa, joka lisäsi aminohapon kerrallaan. Ohjelma tutki kaikki asennot tiettyjen parametrien puitteissa ja valitsi niistä energeettisesti edullisimman. Käyttäjän valitsemia parametreja olivat:

1. peptidisidoksen pituus (käytettiin 1,35 Å),
2. pienin etäisyys kahdelle vedytomille, joista toinen on lisättävässä aminohapossa ja toinen vanhassa proteiinissa (käytettiin 1,35 Å),
3. pienin etäisyys kahdelle atomille, joista toinen on lisättävässä aminohapossa ja toinen vanhassa proteiinissa ja ainakin toinen ei ole vetyatomi (käytettiin 1,7-2,0 Å) sekä
4. eri orientaatioiden etsimisen askelluskulma (käytettiin 5°).

Mekaaninen aminohapon asennon etsiminen, tai “pyörittely”, tapahtui kovalenttisidosten suuntaisien akselien ympäri. Tästä syystä askellus on määritelty asteina.



Kuva 12: Aminohaponlisäysohjelman toiminta. Uuden aminohapon ja vanhan proteiinin rajapinta, jossa atomit 1 ja 2 ovat olemassaolevan proteiinin rakenteen ja atomit 3 ja 4 lisättävän uuden aminohapon rungossa. Oikealla kuva DigA16:n rungosta, jossa katkoviiva merkitsee kahden aminohapon rajapintaa. Punaisella happi, sinisellä typpi ja turkoosilla hiili. Atomit 1 ja 2 ovat aminohaposta numero 167 ja atomit 3 ja 4 aminohaposta numero 168.

Kuvassa 12 on kuvattuna ne kolme kovalenttia sidosta, joiden muodostamien akselien ympäri proteiinia kierrettiin kun ohjelma etsi orientaatioita. Kuvassa atomi 1 on proteiinin

viimeisen aminohapon C_α , 2 on saman aminohapon rungon (backbone) viimeinen C, 3 on lisättävän aminohapon rungon N ja 4 on lisättävän aminohapon C_α . Ensin ohjelma asetti uuden aminohapon siten, että atomi numero 3 on peptidisidoksen päässä atomista numero 2 ja kulma 1-2-3 on 120° . Tämän jälkeen suoritettiin uuden aminohapon kierto atomien 2, 3 ja 4 määräämässä tasossa käyttäen keskipisteenä atomia 3 siten, että kulma 2-3-4 oli 118° .

Kun rajapinnan kulmat ja etäisyydet oli saatu lukittua pystyttiin uutta aminohappoa kiertämään atomien välisten sidosten määräämien akselien ympäri muuttamatta lukittuja kulmia tai etäisyyksiä. Käytän kierroista merkintää, jossa on niiden atomien numerot, jotka määräävät kiertoakselin. Esimerkiksi “1-2” tarkoittaa kiertoa, jossa kiertoakseli ja atomien 1 ja 2 välinen sidos yhtyvät. Kierrossa käytettiin kolmea sisäkkäistä silmukkaa, ts. jokaiselle kierron 1-2 askeleelle käytiin kierron 2-3 kaikki askeleet läpi ja jokaiselle kierron 2-3 askeleelle käytiin kaikki kierron 3-4 askeleet läpi.

Jokaisen sisimmän silmukan (kierron 3-4 askeleen) suorittamisen jälkeen tarkistettiin ovatko hyväksyttävän asennon rajoitukset voimassa (minimietäisyydet). Tällöin ohjelma laski asennolle kokonaispotentiaalienergian ja tallensi muistiin energian sekä sitä vastaavan orientaation. Lopuksi näistä tallennetuista orientaatioista valittiin energeettisesti edullisin asento uudelle aminohapolle. Ohjelma käytti potentiaalin laskemiseen Coulombin- ja LJ-potentiaaleja. Kovalenttisivosten potentiaalit jätettiin laskematta, koska sidospituudet ja kulmat olivat joka tapauksessa lukittu ja ainoastaan atominelikoille laskettava torsioenergia olisi antanut eri tuloksen eri asennoille. Laskussa summattiin kaikkien atomiparien potentiaalienergiat, jos toinen atomi oli lisättävässä aminohapossa ja toinen vanhassa proteiinissa, ja atomien välillä oli enemmän kuin kolme kovalenttia sidosta. Kovalenttisivosten tuoma rajoite vaikutti vain uuden aminohapon ja vanhan proteiinin rajapinnan läheisyydessä. Laskuissa käytettiin AMBER-voimakentän parametreja ja topologiatietoja.

Joissakin tilanteissa aminohapolle tuli valittuksi sellainen asento, jossa oli mahdotonta lisätä tästä seuraavaa aminohappoa rakenteeseen. Tällöin piti valita asennoksi joku muu kuin energeettisesti paras asento. Asennoksi valittiin kuitenkin aina energeettisesti mahdollisimman edullinen asento, kunhan seuraavan aminohapon pystyi lisäämään rajoitusten mukaisesti. Rajoitukset olivat muuttumattomia lukuunottamatta minimietäisyyttä atomeille, joista toinen ei ollut vety. Tätä arvoa piti joskus pienentää, jotta saatiin aminohappo lisättyä kohtalaisella vaivalla. Pienin arvo, jota käytettiin oli $1,7 \text{ \AA}$ alkuperäisen $2,0 \text{ \AA}$:n sijaan.

DigA16:n generoitua rakennetta käytettiin lähtökohtana simulaatiolle, jossa proteiinin annettiin liikkua ja etsiä energeettisesti hyvä orientaatio. Lisäksi yhdessä simulaatiossa määriteltiin hännässä oleva rikki-rikki-sidos, joka oli jätetty generoitaessa huomiotta. Oracilla saatiin helposti “vedettyä” hännässä oleva rikkiatomi paikalleen oikealle sidosetäisyydelle siten, että Oracin parametreissa määriteltiin atomien välille sidos. Sidoksen tasapainoetäisyys oli tavallisen rikki-rikki-sidoksen pituinen, joten kaukana tästä olevat atomit tunsivat valtavan voiman toisiaan kohti. Kuitenkaan atomit eivät liikkuneet liian nopeasti (mikä olisi rikkonut molekyylin), vaan alkusimulaatiossa käytetty “very cold start”-parametri asetti ylärajan atomien liikkumisnopeudelle ja sidos saatiin tasapainoetäisyydelle hallitusti.

1KT7:n simulaatiossa lähtörakennetta ei tarvinnut muokata, sillä se oli Protein Data Bank-tietokannassa täydellisenä. 1KT7:ssa olevien kolmen rikki-rikki-sidoksen tapauksessa atomit olivat siis jo valmiiksi sidoksetäisytydellä. Kuitenkin 1KT7:lle suoritettiin samat alustusajot kuin DigA16:lle, koska haluttiin varmistaa, että simulaatio toimii oikein.

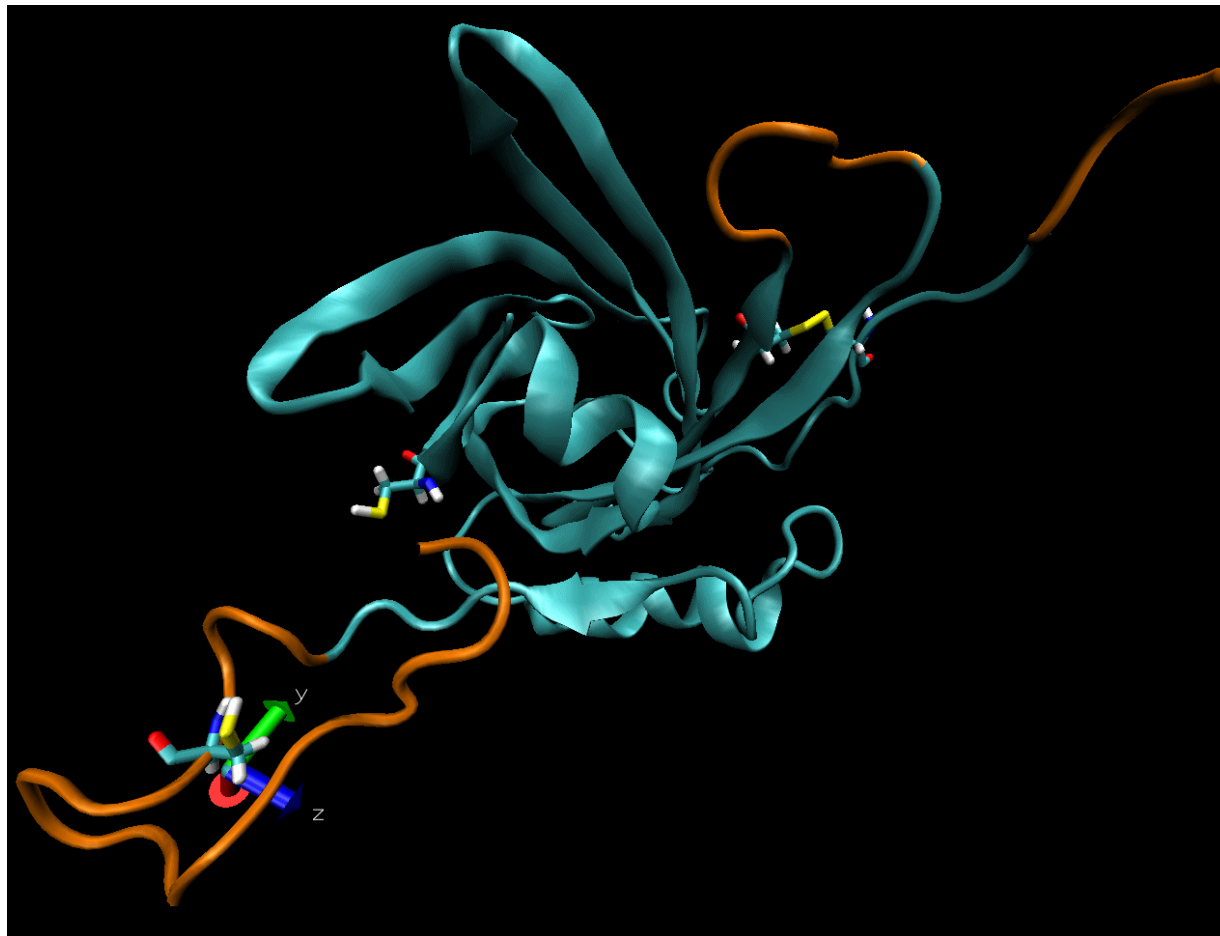
Topologia- ja parametritiedot kummallekin proteiinille olivat enimmäkseen valmiina AMBER:in voimakenttäparametripaketissa, mutta joitakin jousivakio-tasapainoarvo-pareja sidoksille, kulmille ja torsiokulmille piti kehittää itse. Lisäksi kaikki pistevaraukset ja täydellinen topologia proteiinien sitomille vierasmolekyylille (digoksigeniini ja retinoli) piti tehdä itse. AMBER:ista puuttuvat varaukset laskettiin DFT:llä (density functional theory). DFT:tä käytetään yleisesti elektronirakennelaskuihin molekyyleille ja materiaaleille. DFT:ssä lasketaan molekyylin elektronikaasun tiheys avaruudessa, tästä lasketaan millaisen sähkökentän ulkopuolella oleva tarkkailija molekyyylille havaitsee ja lopulta sovitetaan atomeille pistevaraukset, jotka tuottavat vastaavan sähkökentän ulkopuoliselle tarkkailijalle. Näitä pistevarauksia käytettiin digoksigeniinin ja retinolin atomien varauksina (atomiydinten varaukset huomioiden) molekyylidynamiikkasimulaatiossa.

Kun lähtörakenne ja -tiedot oli kokonaisuudessaan valmiina, voitiin alkaa simuloimaan Oracilla. Ensin proteiini ja simulaatioon lisätty vesi piti termalisoida 300 Kelviniin rauhallisesti. Aluksi ajettiin lyhyt yhden pikosekunnin pituinen ajo 10 K:n lämpötilassa käyttäen “very cold start”-parametria, joka estää simulaation epäonnistumisen tilanteessa, jossa alkuasetelma ei olisi aivan fysikaalinen. Tämän jälkeen ajettiin kolme kappaletta 150 ps ajoja: ensin 100 K:ssä, sitten 200 K:ssä ja lopuksi 300 K:ssä. Näissä lämpötila pidettiin haluttuna koko ajon ajan uudelleenskaalaamalla atomien nopeuksia tarvittaessa. Tällaisesta alustuksesta käytetään tavallisesti nimeä “hylkäysvaihe” (engl. “reject-phase”).

Tämän jälkeen ajettiin yksi 150 ps ajo 300 K:ssä käyttäen termostaattia ja barostaattia. Simulaatiokopin koon annettiin muuttua vapaasti paineen pysyessä yhdessä ilmakehässä. DigA16:n simulaatiossa, jossa proteiinin sisällä oli vierasmolekyyli oli kopin koon muutos alle 0,1 Å kaikkien koordinaattiakselin suhteen eli kyseessä oli lopulta vain tarkistus, että simulaation paine oli hyvä ja että järjestelmä oli vakaa. Muissa DigA16:n simulaatioissa kopin koko muuttui hieman enemmän, kuitenkin alle 1 Å suuntaansa. 1KT7:n simulaatiossa kopin koko muuttui $59 \times 56 \times 54$ Ångströmistä hieman alle $58 \times 55 \times 53$ Ångströmiin eli pieneni noin 1 Å joka suunnassa. Lopuksi järjestelmillä ajettiin vielä viimeinen termalisoitaja (150 ps 300 K:ssä) käyttäen termostaattia, joka piti lämpötilan lähellä tavoitearvoa. Yhteensä alustuksessa simuloitiin 751 ps, josta 450 ps simuloitiin 300 K:ssä. Näiden alustusten jälkeen pystyttiin kymmenen nanosekunnin tuotantoajot aloittamaan. Yhden nanosekunnin simuloimiseen meni käytettävissä olevalta tietokoneelta hieman yli neljä päivää. Tietokoneessa oli neljä kappaletta 64-bittisiä 2,4 GHz:n kellotaajuudella toimivia kaksiytimisiä AMD Opteron-proessoreita, yhteensä siis kahdeksan ydintä. Yhtä simulaatiota ei voitu ajaa rinnan usealla prosessorilla, mutta samaan aikaan voitiin eri ytimillä ajaa useita eri simulaatioita.

Tulosten analysointiin käytettiin pääasiassa VMD-ohjelmistoa [40]. Tällä tehtiin avaruudellisen rakenteen kuvat, Ramachandran kaaviot sekä laskettiin RMS-poikkeamat. Kuvia ja piirrettiin erikseen Gnuplot-ohjelmistolla.

3 Tulokset



Kuva 13: *DigA16*-proteiini ilman toista *S-S*-sidosta eli "prot1". Väritys kuten kuvassa 1.

Tässä työssä tutkittiin *DigA16*-proteiinia (kuvat 1 ja 13), jonka kokeellisesti tunnettuun rakenteeseen oli lisätty puuttuvat aminohapot tietokoneohjelmien avulla sekä 1KT7-proteiinia (kuva 14). *DigA16*:ssa on kaksi mahdollista paikkaa rikki-rikki-sidokselle, näistä toinen jätettiin kahdessa simulaatiossa auki (eli atomien välille ei määritelty kovalenttia sidosta) ja tämän vaikutusta proteiiniin tutkittiin. *DigA16*:sta simuloitiin kolme eri rakennetta, jotka on nimetty käsittelyn helpottamiseksi. Molemmat potentiaaliset rikki-rikki-sidokset ovat kahden kystiini-aminohapon rikkiatomien välillä. Ensimmäinen oli ilman toista rikki-rikki-sidosta aminohappojen 42 ja 170 välillä, josta käytetään nimitystä "prot1" (kuva 13). Toinen sisältää kummatkin rikki-rikki-sidokset, jotka ovat aminohappojen 8 ja 115 välillä ja aminohappojen 42 ja 170 välillä. Tästä käytetään nimitystä "prot2" (kuva 1). Kolmas simuloitu rakenne *DigA16*:sta oli kuten "prot1" (eli yhdellä *S-S*-sidoksella), mutta simulaatiossa proteiinin taskussa oli vierasmolekyylä (digoksigeniini). Tähän viitataan nimityksellä "prot3". Kuvassa 1 on "prot2", jossa tietokoneella lisätyt proteiinin osat ovat oranssilla

värillä merkittynä. Lisäksi kuvassa on näkyvissä kystiini-aminohapot, jotka muodostavat rikki-rikki-sidokset. Nämä sidokset ovat merkittäviä, koska ne pitävät proteiinin päät lähellä muuta proteiinia. Sidoksen puuttuessa proteiinin häntä lähtee vaeltelemaan melko vapaasti ja kauas β -tynnyrirakenteesta. Tämä huomattiin simulaatiossa “prot1” (kuva 13), jossa jätettiin 1 rikki-rikki-sidos pois. Simuloitaessa rakennetta “prot2” tällaista liikkumista ei havaittu.

1KT7-proteiinista simuloitiin yhtä rakennetta. Tämän proteiinin avaruudellinen rakenne on kokonaisuudessaan mitattu kokeellisesti, joten siihen ei tarvinnut erikseen lisätä osia, kuten DigA16:n tapauksessa. Rakenteessa olevat S-S-sidokset olivat kystiini-aminohappojen 4 ja 160, 70 ja 174 sekä 120 ja 129 välillä. Simuloituun rakenteeseen kuului vierasmolekyylä (retinoli) 1KT7:n sisällä, β -tynnyrin muodostamassa onkalossa, jonka on havaittu sitovan retinolia. Simulaation aikana retinoli pysyikin hyvin proteiinin sisällä.

Molekyylidynamiikkasimulaatioissa voidaan tutkia molekyyliden rakennetta, dynamiikkaa ja energetiikkaa. Lisäksi voidaan tutkia molekyyliden välisiä vuorovaikutuksia, kuten 1KT7- ja “prot3”-simulaatioissa tehtiin. Proteiinien konformaatiota voidaan tutkia myös Ramadanchan kaaviolla. Tämä on kunkin aminohapon kahden torsioikulman visualisointi koordinaatistoon, jossa jokaisesta aminohaposta tulee siis yksi piste. Näistä kulmista käytetään nimiä ϕ - ja ψ -kulmat (kuva 15). Koordinaatistossa vaakakselilla on ϕ -kulma ja pystyakselilla ψ -kulma, ja kumpikin saa arvoja väliltä $[-\pi, \pi]$ (nollakohta keskellä). Tietyt alueet Ramadanchan kaaviossa vastaavat sekundaarisia rakenneosia, ja nämä on esitetty kuvassa 16. Punaiset alueet kuvaavat sallittuja konformaatioita. Tällöin atomien paikoille voi asettaa van der Waals-säteiset pallot ilman, että ne menevät toistensa päälle. Keltaiset alueet ovat alueita, jotka ovat sallittuja, jos laskuissa käytetään hieman pienempiä van der Waals-säteitä. Käytännössä keltaiset alueet ovat sallittuja, mikäli aminohappoina on vain glysiiniä. Luonnon vasenkätiset aminohapot eivät voi muodostaa suurempaa kokonaisuutta vasenkätistä α -heliksiä, vaikka se onkin mahdollinen konformaatio (oikea yläneljännes).

3.1 DigA16-proteiinin simulaatiot

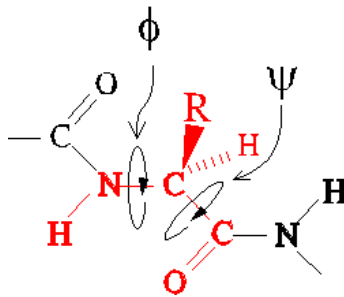
3.1.1 Energetiikka

Simulaatioiden kokonaisenergiat (kuva 17) olivat vakaita. Punainen käyrä on simulaatiosta “prot1”, jossa toista rikki-rikki-sidosta ei huomioitu, sininen simulaatiosta “prot2”, jossa se huomioitiin ja purppura käyrä on simulaatiosta “prot3”. Energiat ovat erisuuret, koska simulaatioissa on eri määrät atomeita ja siten eri määrä sidoksia ja pidemmän kantaman paripotentiaaleja. Simulaatioiden “prot1” ja “prot2” energiat ovat tosin lähellä toisiaan, koska ero atomeiden lukumäärässä on pieni. Kolmannessa simulaatiossa mukana oli digoksigeniinimolekyylä, ja energia on selvästi alempana kuin kahdessa ensimmäisessä. Huomatavaa on kuitenkin, että energiat kohoavat hitaasti ja tasaisesti. Tämä on merkki siitä, että kohoava energia on seurausta simuloinnin aikana kumuloituvasta pyörästysvirheestä, joka riippuu käytetystä aika-askeleesta.

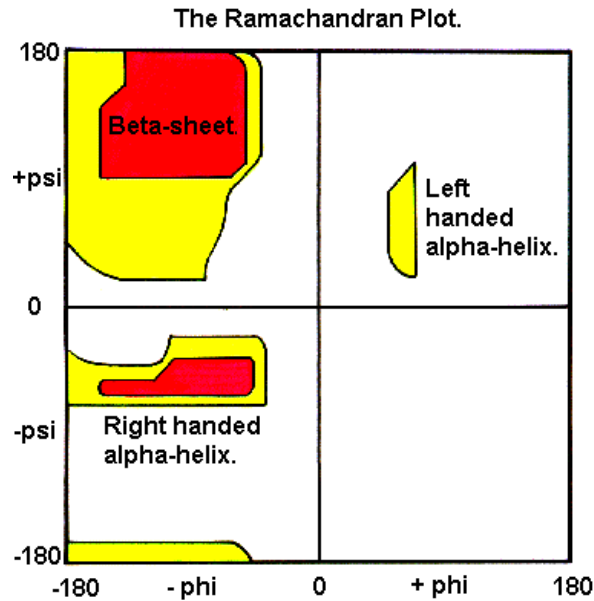
Proteiinin sisäisten vuorovaikutusten potentiaalienergia (kuva 18) vaihtelee kummasakin simulaatiossa “prot1” ja “prot2” nopeasti kohtuullisen pienellä välillä. Loppua kohti



Kuva 14: 1KT7-proteiini. Itse proteiini näkyy kuvassa turkoosilla. Proteiinin sisällä näkyy retinolimolekyylä, jossa turkoosi väri tarkoittaa hiiliatomia, punainen happiatomia ja valkoinen vetyatomia. Retinoli on lipokaliiniproteiineille tyypillisessä onkalossa 1KT7:n sisällä.



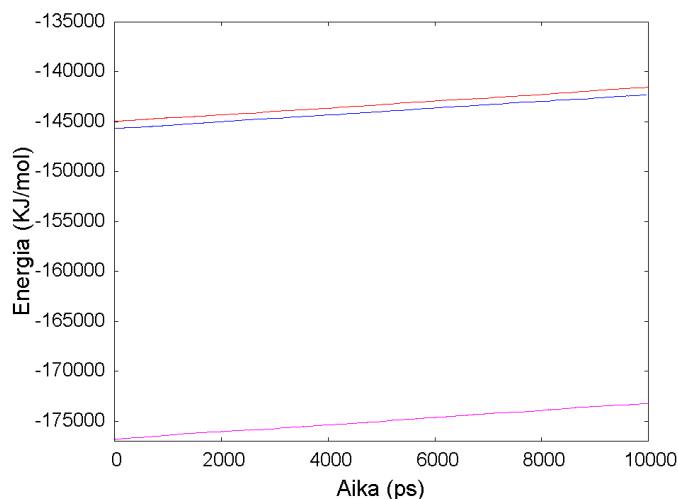
Kuva 15: Aminohapon ϕ - ja ψ -kulmat aminohappoketjussa [4].



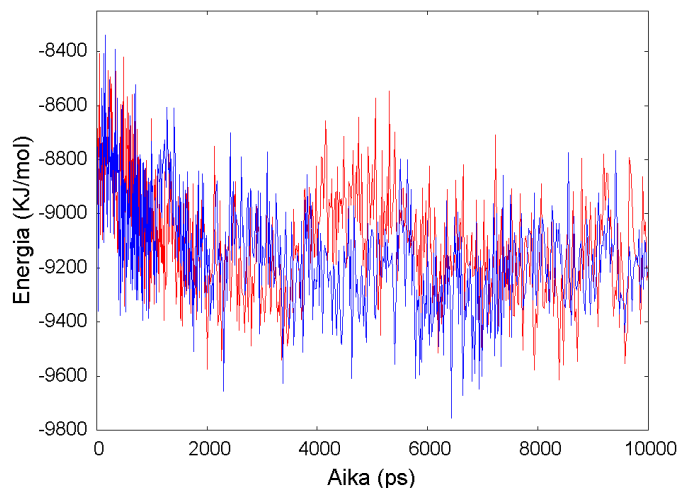
Kuva 16: *Ramachandran kaavion sekundaarisia rakenneosia vastaavat alueet [41].*

liike hidastuu ja energia laskee hieman, tämä johtuu siitä, että proteiini löytää vähitellen energeettisesti edullisemmän orientaation. Erityisen mielenkiintoiseksi osoittautui simulaation “prot1” proteiinin potentiaalienergiakäyrä, sillä siinä oli selvä kumpu ja simulaatiosta paljastuikin tarkemmalla tutkimisella muutaman aminohapon ketjun konformaatiomuutos, josta enemmän myöhemmin. Saman suuntaisia tuloksia antaa proteiinin ja veden välisen potentiaalienergian kuvaaja (kuva 19). Myöhemmin myös simulaatiosta “prot3” löytyi vastaava konformaatiomuutos. Potentiaalienergia vaihtelee nopeassa tahdissa, mutta loppua kohden vaihtelu hidastuu ja energiataso nousee hiukan. Proteiinin (ja hiukan myös veden) löytämän paremman orientaation vapauttama energia kompensoituu proteiinin ja veden välisessä vuorovaikutuksessa.

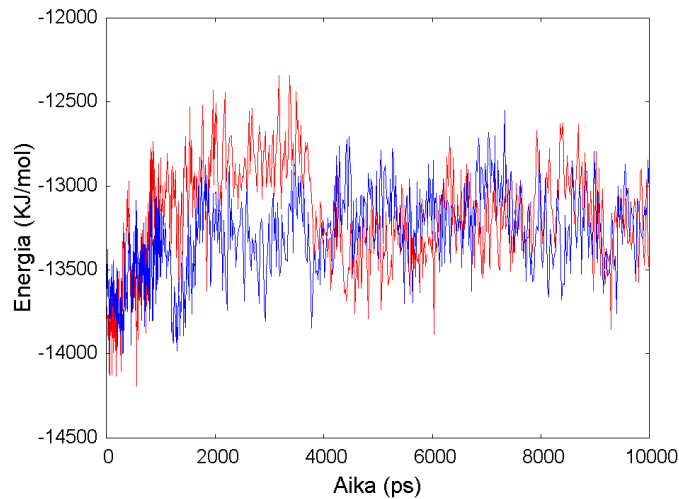
Simulaation “prot3” tulokset olivat suurilta osin odotettuja. Tosin simulaation tuloksia tutkittaessa havaittiin pieni vastoinkäyminen, sillä simulaation suorittamiseen käytetyn tietokoneen vaihtaminen aiheutti sen, että proteiinin tarkkaa trajektoria tiheällä tallennusvälillä (9,6 ps) ei ollut saatavilla. Muu data simulaatiosta on kunnossa ja kelvollista, mutta trajektoria ei voinut analysoida niin tarkasti kuin edeltävissä simulaatioissa. Simulaation kokonaisenergia näkyy kuvasta 17 ja se käyttäytyy kuten simulaatioiden “prot1” ja “prot2” energiat. Syykin on sama: kumuloituva pyöristysvirhe, joka johtuu äärellisestä aika-asteleesta. Proteiinin potentiaalienergiassa näkyy pieni kohouma noin 2,5-4,5 nanosekunnin ympärillä (kuva 20) ja samansuuruinen kuoppa näkyy proteiinin ja liuoksen välisessä sidosenergiassa kuvassa 21. Tämä johtune konformaatiomuutoksesta, jollainen löytyi simulaation trajektoria tarkastelemalla.



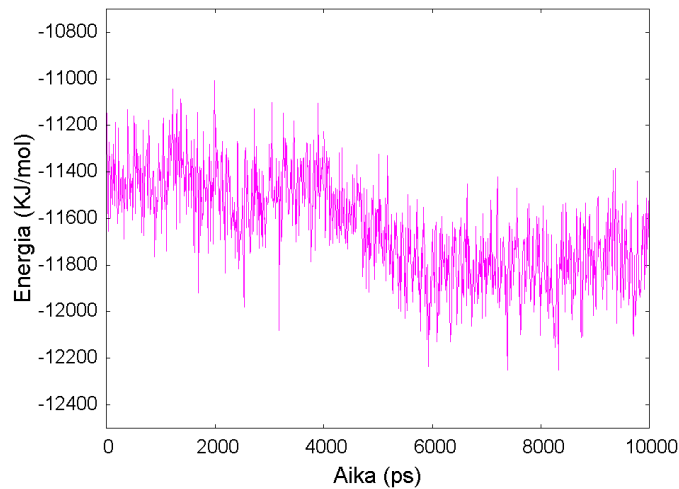
Kuva 17: Kokonaisenergia 300 Kelvinin lämpötilassa. Pystyakselilla energia (KJ/mol) ja vaaka-akselilla aika (ps). Punainen käyrä on simulaatiosta “prot1”, jossa toinen rikki-rikki-sidos jätettiin huomiotta ja sininen käyrä on simulaatiosta “prot2”, jossa molemmat otettiin huomioon. Purppura käyrä on simulaatiosta “prot3”, joka sisälsi vierasmolekyylin.



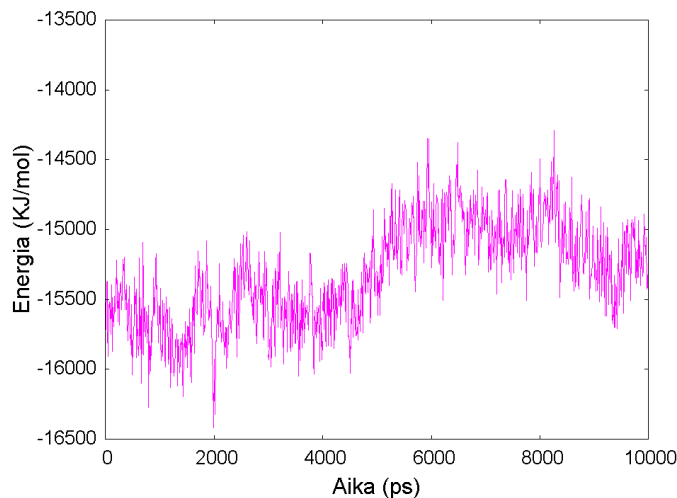
Kuva 18: Proteiinin potentiaalienergia simulaatioissa “prot1” ja “prot2”. Pystyakselilla potentiaalienergia (KJ/mol) ja vaaka-akselilla aika (ps).



Kuva 19: *Liuksen ja proteiinin välinen sidosenergia simulaatioissa “prot1” ja “prot2”. Väritys ja skaala kuten kuvassa 17.*



Kuva 20: *Simulaation “prot3” proteiinin potentiaalienergia. Proteiinin sisällä oleva vieras-molekyyl - eli sen sisäinen potentiaalienergia sekä koko proteiinin kanssa muodostettujen vuorovaikutusten potentiaalienergia - on laskettu mukaan proteiinin potentiaalienergiaan. Väritys ja skaalaus kuten kuvassa 17.*



Kuva 21: Simulaation “prot3” liuoksen ja proteiinin välinen sidosenergia. Väritys ja skaalaus kuten kuvassa 17.

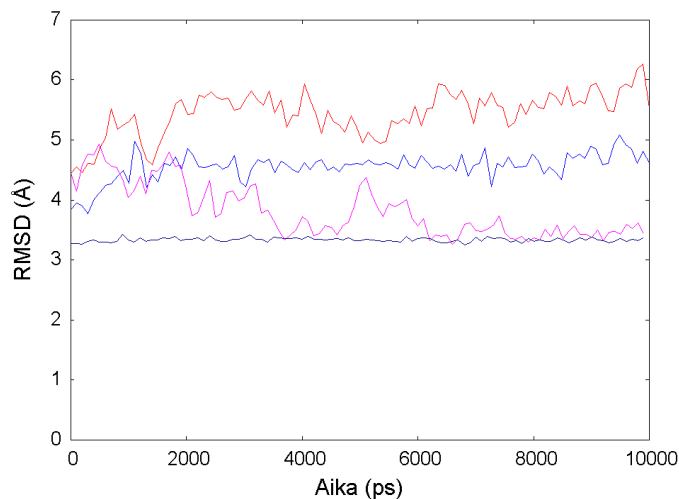
3.1.2 Konformaatio

Kuvassa 22 on proteiinin atomeiden koordinaattien RMS-poikkeama kullakin ajanhetkellä (root-mean-square-deviation, neliöjuuri poikkeaman neliön keskiarvosta) verrattuna kokeellisiin koordinaatteihin. Kuvissa on DigA16:n simulaatioiden lisäksi vertailun vuoksi myös kuvaajat 1KT7:n simulaation datasta ja näihin palataan myöhemmässä kappaleessa. Kuvassa 23 on RMS-poikkeama, joka on laskettu rajoittuen proteiinin rungon (backbone) atomeihin. Punainen käyrä on simulaatiosta “prot1” ja se nousee kummassakin tapauksessa huomattavasti korkeammalle, kuin sininen käyrä. Tämä johtuu juuri vapaana (ilman S-S-sidosta) vaeltavan hännän liikkeen aiheuttamasta vaikutuksesta RMS-poikkeamaan.

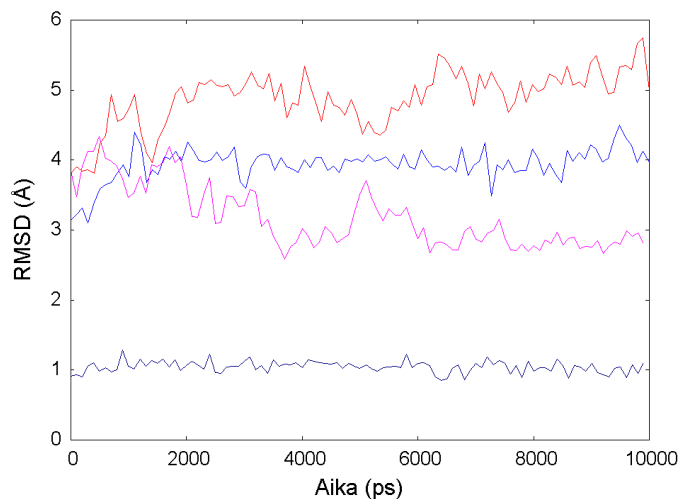
Mielenkiintoista on se, että “prot3”:n RMS-poikkeama (purppura käyrä kuvissa 22 ja 23) laskee alun jälkeen. Tämä tarkoittaa sitä, että proteiini palaa lähemmäs kokeellisia koordinaatteja termalisoinnin jälkeen varsinaisen simulaation aikana. β -tynnyrin alueelta laskettu RMS-poikkeama (kuva 24) näyttää siltä, että vierasmolekyylillä stabiloi β -tynnyriä DigA16:n simulaatiossa “prot3” verrattuna kahteen muuhun. Hieman yllättäen “prot2”:n käyrä on korkeammalla, kuin “prot1”:n. Tasapainottaako puuttuva S-S-sidos digoksigeniinin puuttuessa β -tynnyrin? Selvää on ainakin se, että digoksigeniini saa β -tynnyrin pysymään vakaampana. RMS-poikkeama “prot3”:lle on noin puolet siitä, mitä se on “prot2”:lle.

Simulaation “prot1” loppupään hännän kahden rikkiatomin välisen etäisyyden kuvaaja (kuva 25) kertoo hyvin, miten sidoksen jättäminen pois antaa hännän heilua lähes vapaasti. Toisen sidoksen kohdalla (sininen käyrä), missä rikit on sidottu toisiinsa on etäisyys hyvin tarkasti vakio. Punaisessa käyrässä sen sijaan rikkiatomien välinen etäisyys pysyy suurena ja vaihtelee huomattavasti.

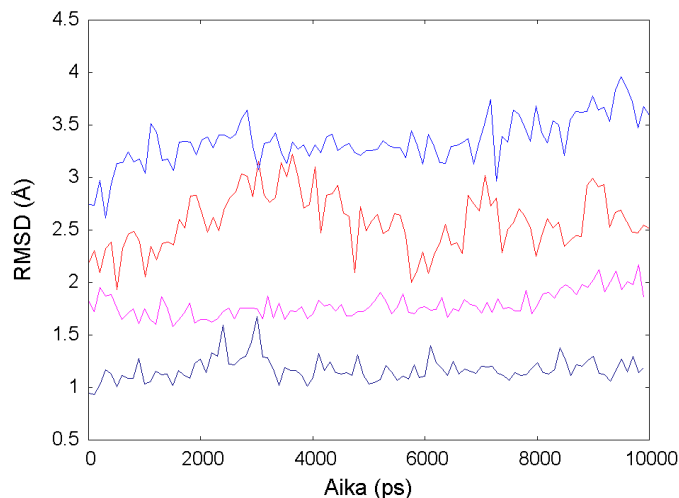
Kuvassa 26 kahden simulaation lopputulokset on aseteltu alkuperäisen, kokeellisesti saadun rakenteen päälle käyttäen pienimmän RMS-poikkeaman tuottavaa asentoa. Tässä tapauksessa RMS-poikkeamaa laskettaessa huomioitiin vain aminohapot 5-166 eli ne



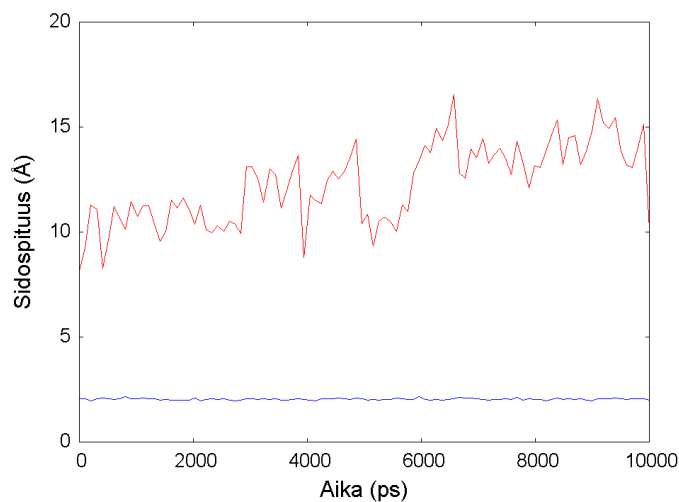
Kuva 22: Koko proteiinin RMS-poikkeama ajan funktiona. Kunkin ajanhetken koordinaatteja on verrattu kokeellisesti mitattuihin koordinaatteihin. Kuvassa punaisella “prot1”, sinisellä “prot2”, purppuralla “prot3” ja mustalla 1KT7.



Kuva 23: Proteiinin rungon RMS-poikkeama ajan funktiona. Kunkin ajanhetken koordinaatteja on verrattu kokeellisesti mitattuihin koordinaatteihin. Kuvassa punaisella “prot1”, sinisellä “prot2”, purppuralla “prot3” ja mustalla 1KT7.



Kuva 24: β -tynnnyriin rajoitettu RMS-poikkeama simulaatioista. Punaisella “prot1”, sinisellä “prot2”, purppuralla “prot3” ja mustalla 1KT7. DigA16:n RMS-poikkeama on laskettu aminohappojen 24-133 koordinaateista ja 1KT7:n aminohappojen 22-139 koordinaateista.



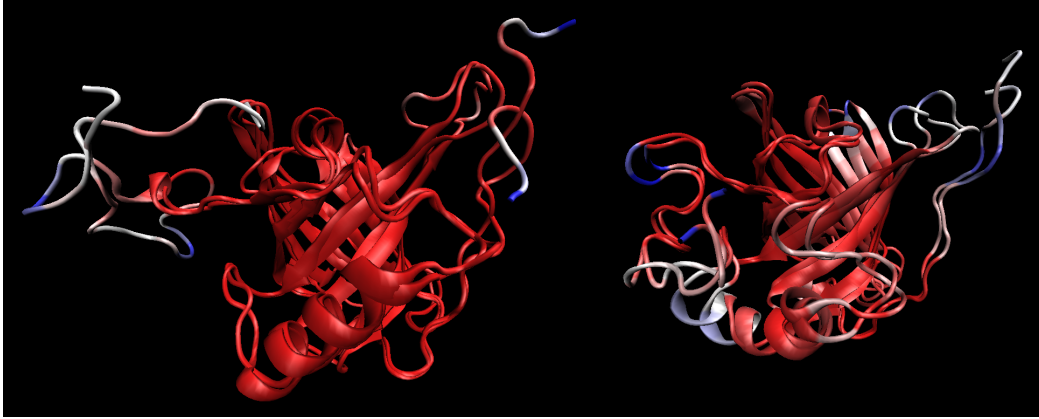
Kuva 25: Rikki-atomien välinen etäisyys simulaatiossa “prot1”. Nämä rikkiatomiparit muodostavat mahdollisesti kaksi rikki-rikki-sidosta luonnossa. Punaisella värillä on aminohappojen numero 42 ja 170 ja sinisellä aminohappojen 8 ja 115 rikkiatomien etäisyys. Vaaka-akselilla aika (ps) ja pystyakselilla S-S-etäisyys Ångströmeinä.

aminohapot, joiden asento oli havaittu kokeellisesti. Tämän jälkeen proteiinit väritettiin sen mukaan miten hyvin ne osuvat päällekkäin. Punaiset alueet ovat lähimpänä toisiaan ja väri muuttuu valkoisen kautta siniseksi, kun ero kasvaa. Oikealla puolella, simulaation “prot2” kuvassa, kaikkein lähimpänä kokeellista orientaatiota ovat proteiinin keskellä olevat β -levyt. Lisäksi on huomattavaa, että ilman rikki-rikki-sidosta alkupuolen hännäntynkä liikkuu suhteellisesti vähemmän kuin sidoksen ollessa läsnä. Myös aminohappojen 118-124 muodostama silmukka liikkuu suhteessa vähemmän simulaatiossa “prot1”. Simulaatiossa “prot2” huomataan α -heliksin (aminohapot 140-152, oikeanpuoleisessa kuvassa alareunassa) ja sen ympäristön suhteellisen liikkeen olevan suurempi, kuin simulaatiossa “prot1”. Tämä johtuu juuri siitä, että värityksen “maksimi” pitää skaalata suurimman liikkeen mukaan, mikä simulaatiossa “prot1” oli suurempi kuin simulaatiossa “prot2”.

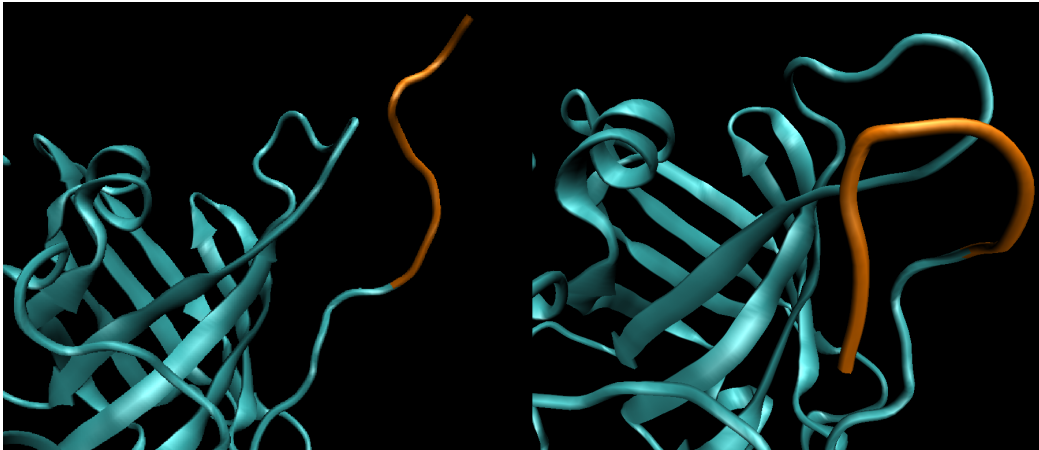
DigA16:n simulaatiossa “prot1” havaittiin kuvan 18 punaisesta energiakäyrästä mielenkiintoinen konformaatiomuutos. Energiakäyrässä havaittiin ensin kohouma, jonka vuoksi trajektorin animaatiota VMD:ssä tarkasteltiin kyseiseltä ajanjaksolta tarkemmin ja havaittiinkin lopulta kuvassa 27 kuvattu hännän tekemä konformaatiomuutos. Konformaatiomuutokseen osallistuu varsinaisesti 4-5 ensimmäistä aminohappoa proteiinin aminohappopoketjusta, vaikka tietysti kyseisen kaltainen konformaatiomuutos vaikuttaa jonkin verran myös kauemmas lähiympäristöönsä. Konformaatiomuutoksen aikaskaala on kahden nanosekunnin luokkaa, joten sellaisen havaitseminen vain kymmenen nanosekunnin simulaatiossa lienee harvinaista. Kyseisessä simulaatiossa toinen potentiaalinen rikki-rikki-sidos on auki, mutta se on toisessa päässä proteiinia. Alkupään rikki-rikki-sidos, joka on lähellä konformaatiomuutoksen paikkaa, on puolestaan kiinni.

Simulaatiossa “prot3” kokeelliseen rakenteeseen verrattaessa (kuva 28) lähinnä proteiinista ulkonevat hännät ja silmukat ovat liikkuneet eri asentoon. Proteiinin β -tynnyrirunko on käytännössä samassa asennossa kuin kokeellisesti mitatussa rakenteessa. Mielenkiintoisena yksityiskohtana nousee esiin alkupään häntä (kuvassa 28 oikealla ylhäällä sinisellä värillä), joka kokee “prot1”:n konformaatiomuutosta vastaavan muutoksen (kuva 29). Noin 2,5-4,5 nanosekunnin kohdalla simulaatiota proteiinin energiakäyrässä (kuva 20) on pieni kohouma ja samassa kohtaa trajektorin animaatiota näyttää siltä, että alkupään häntä melkein lukkiutuu taipuneeseen olevaan asentoon, kun se sitä aiemmin on heilunut suorempana käyden vain ajoittain lyhyesti taipuneena. Heti tämän jälkeen proteiinin sisäisten vuorovaikutusten energia laskee noin parin nanosekunnin ajan ja jää lopuksi laskeneelle tasolle.

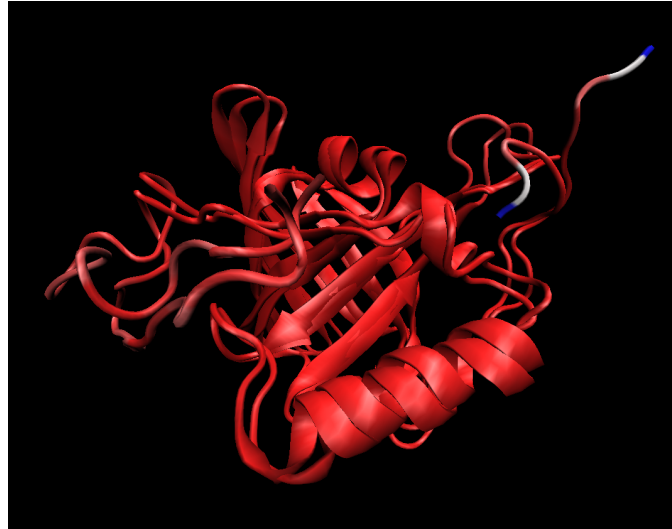
Ramachandran kaaviosta nähdään, että “prot1”-simulaatiossa (kuva 30) aminohapot, jotka ovat olleet (oikeakätisen) α -heliksi- ja β -levykonformaatioiden välissä ovat liikkuneet hieman kohti α -heliksi-konformaatiota. Lisäksi β -levykonformaatiossa olevat aminohapot ovat liikkuneet vähemmän sallitulle alueelle. Simulaatiossa “prot2” (kuva 31) α -heliksi- ja β -levykonformaatioalueiden välissä olevat aminohapot ovat liikkuneet oikealle kohti positiivista ϕ -kulmaa ja aavistuksen alas kohti α -heliksikonformaatiota. Lisäksi enemmän sallitulla α -heliksialueella olevat aminohapot ovat levyttäneet laajemmalle pysyen kuitenkin sinisellä alueella. Digoksigeniinin kanssa simuloidussa “prot3”-rakenteessa (kuva 32) α -heliksi- ja β -levykonformaatioalueiden välissä olevat aminohapot ovat liikkuneet ensinäkin alaoikealle kohti positiivista ϕ -kulmaa ja enemmän negatiivista ψ -kulmaa. Toiseksi,



Kuva 26: *DigA16:n* poikkeama kokeellisesta rakenteesta 10 ns simulaation jälkeen. Vasemmalla ilman toista rikki-rikki-sidosta, oikealla sidoksen kanssa. Punainen väri tarkoittaa, että koordinaatit ovat lähellä kokeellisia, sininen tarkoittaa, että koordinaatit ovat muuttuneet enemmän. Väriskaala on suhteellinen eli oikeanpuoleisessa “*prot2*”-simulaation kuvassa β -tynnyrissä on enemmän sinistä koska tynnyrin ulkopuoliset hännät eivät ole liikkuneet niin paljoa, kuin “*prot1*”-simulaatiossa.



Kuva 27: Vasemmalla kuvankaappaus VMD:llä simulaatiosta “*prot1*” kohdasta 3,3 ns. Oikealla vastaavasta osasta proteiinia kuvankaappaus kohdasta 9,6 ns. Kummassakin kuvassa on korostettuna oranssilla värillä aminohapot 1-5.



Kuva 28: *DigA16:n* rakenteen “*prot3*” poikkeama kokeellisesta 10 ns simulaation jälkeen. Väritys kuten kuvassa 26.



Kuva 29: Vasemmalla kuvankaappaus VMD:llä simulaatiosta “*prot3*” kohdasta 1,0 ns. Oikealla samasta simulaatiosta kohdasta 5,5 ns. Kummassakin kuvassa korostettuna oranssilla värillä aminohapot 1-5.

ψ -kulman nollalinjalle on kerääntynyt joukko aminohappoja. Tämä joukko koostuu aminohapoista 159-184, jotka ovat alkurakenteessa hajallaan kaavion vasemmalla puoliskolla (negatiivisen ϕ -kulman alueella).

Simulaation “prot1” Ramachandran kaaviota analysoitiin tarkemmin ja tutkittiin, missä konformaatioissa tietyt aminohapot olivat olleet. Tutkittaviksi aminohapoiksi valittiin ne, jotka olivat valkoisella (ns. “kielleyllä”) alueella loppukonformaation Ramachandran kaaviossa, eli ne eivät muodosta mitään sekundaarista rakennosaa. Taulukossa 1 on listattuna nämä aminohapot ja niiden simulaation aikaiset konformaatiot Ramachandran kaavion perusteella. Useat näistä aminohapoista olivat olleet valtaosan ajasta jonkin konformaation alueella ja kävikin ilmi, että monet β -levyt lyhenevät ja pitenevät simulaation aikana vähäksi aikaa ja jotkut jopa jakautua kahdeksi erilliseksi levyksi (kuva 33), kun joku aminohappo levyn puolivälistä muuttaa konformaatiotaan hetkellisesti pois β -levyn konformaatioalueelta. Pitää kuitenkin muistaa, että muutokset ovat lyhytaikaisia ja konformaatiot palautuvat ennalleen.

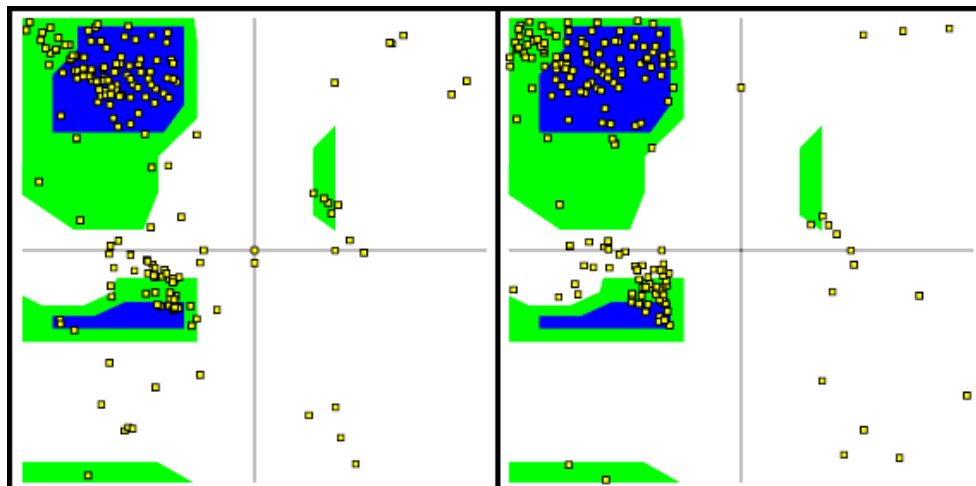
Simulaatiolle “prot2” tarkasteltiin myös tarkemmin niitä aminohappoja, jotka ovat loppukonformaatiossa kielleyllä alueella Ramachandran kaaviossa. Taulukossa 2 on lueteltu nämä aminohapot ja niiden muut konformaatiot. Huomattavaa on se, että “prot2”:n Ramachandran kaaviossa moni aminohappo oli α -heliksikonformaation vieressä loppukonformaatiossa. Nämä aminohapot tuntuivat liikkuvan α -heliksikonformaation alueella ja hieinan sen ulkopuolella. Välillä osa aminohapoista kävi β -levykonformaation alueellakin. Toisin kuin “prot1”:n simulaatiossa, “prot2”:n simulaatiossa ei havaittu β -levyn jakautumista kahtia lainkaan ja vain yhden levyn lyhentymistä ajoittain.

Simulaatiossa “prot3” kielleyllä alueella olevat aminohapot käyttäytyvät kuten kahdessa muussakin DigA16:n simulaatiossa. Huomattava osa aminohapoista on β -levy- ja α -heliksirakenteen välissä ja/tai α -heliksirakenteen reunalla ja alueella. Toinen suurempi ryhmä on ne aminohapot, jotka viettävät koko simulaation kielleyllä alueella muodostamatta mitään sekundaarisia rakenteita. Jotkin aminohapot käyvät välillä jonkin sekundaarirakenteen alueella Ramachandran kaaviossa, mutta eivät jää sinne pysyvästi vaan palaavat kielleylle alueelle.

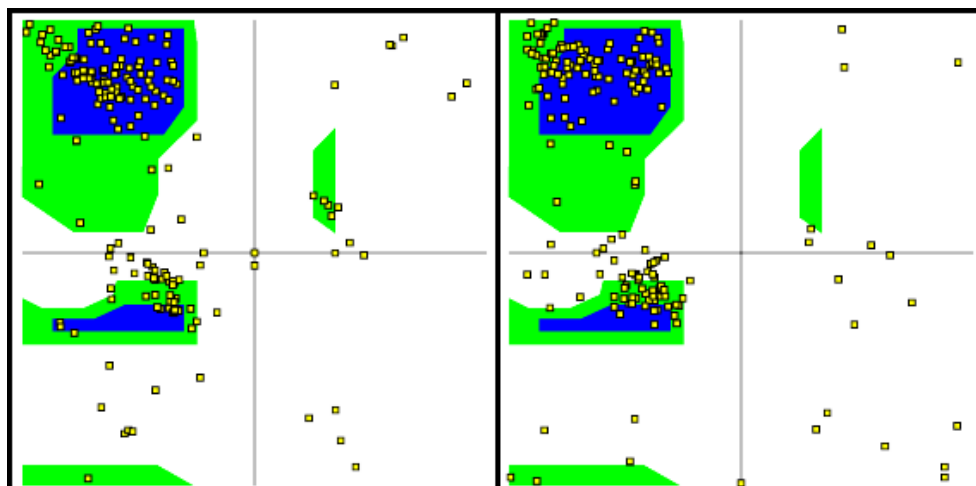
3.1.3 Vierasmolekyylin käyttäytyminen

Kuvassa 34 on simulaation “prot3” loppurakenteesta digoksigeniinimolekyylä, ja sen lähellä olevat DigA16:n onkaloon loukkuun jääneet vesimolekyylit sekä proteiiniin kuuluvat onkalon sisäpinnalla olevat vety- ja happiatomit, jotka osallistuvat vetysidoksiin digoksigeniinin kanssa. Vesimolekyylit ovat eri puolilla vierasmolekyylä onkalon syvässä päässä ja muodostavat siellä sidoksia proteiiniin ja digoksigeniiniin kanssa. Sivuilla digoksigeniini sitoutuu suoraan proteiiniin vety- ja happiatomeihin vetysidoksin.

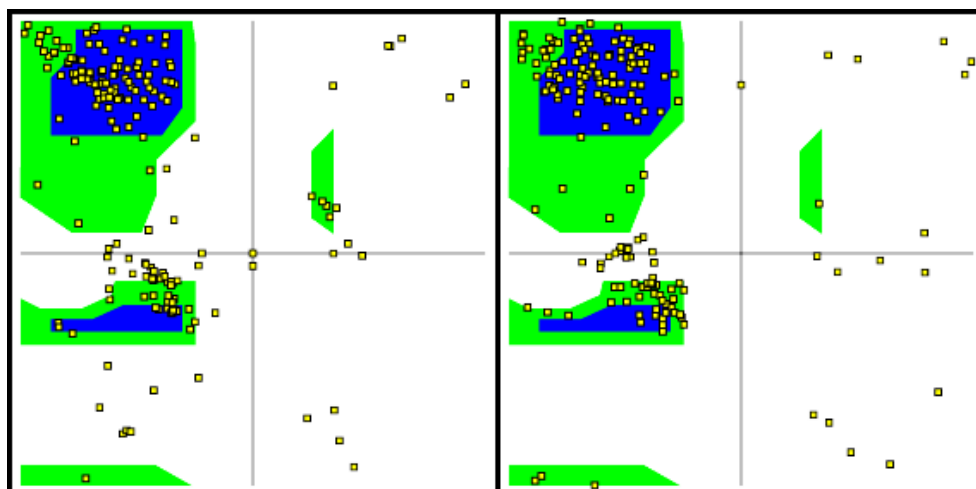
Simuloitaessa DigA16:a vierasmolekyylin kanssa simulaatiossa “prot3” voitiin seurata myös digoksigeniinimolekyylin lähistöllä olevia vesimolekyylejä, joiden kanssa digoksigeniini tai DigA16 voi muodostaa vetysidoksia. Yksi tällainen vesimolekyylä oli simulaatiossa tunnuksella “HOH 207”. Kuvassa 35 on HOH 207:n sidoksetäisyys digoksigeniiniin atomiin “O12” (toisessa päädyssä molekyylä) ja DigA16:n aminohapon GLU70 atomiin “O”



Kuva 30: Simulaation “prot1” Ramachandran kaaviot. Vasemmalla lähtörakenne, oikealla rakenne simulaation päätyttyä. Sininen alue vastaa kuvan 16 punaista, enemmän sallittua aluetta ja vihreä saman kuvan keltaista aluetta.



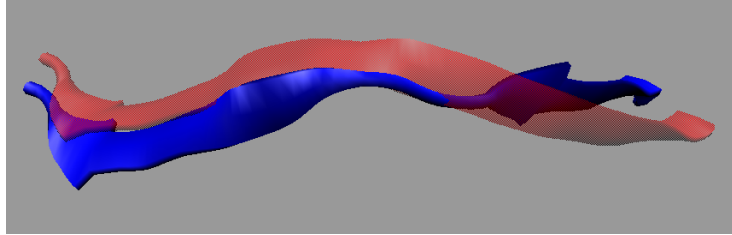
Kuva 31: Simulaation “prot2” Ramachandran kaaviot. Vasemmalla lähtörakenne, oikealla rakenne simulaation päätyttyä. Väritys kuten kuvassa 30.



Kuva 32: Simulaation “prot3” Ramachandran kaaviot. Vasemmalla lähtörakenne, oikealla rakenne simulaation päätyttyä. Väritys kuten kuvassa 30.

Taulukko 1: Simulaation “prot1” loppukonformaation Ramachandran kaaviossa kielletyllä alueella olevien aminohappojen konformaatiot simulaation ajalta.

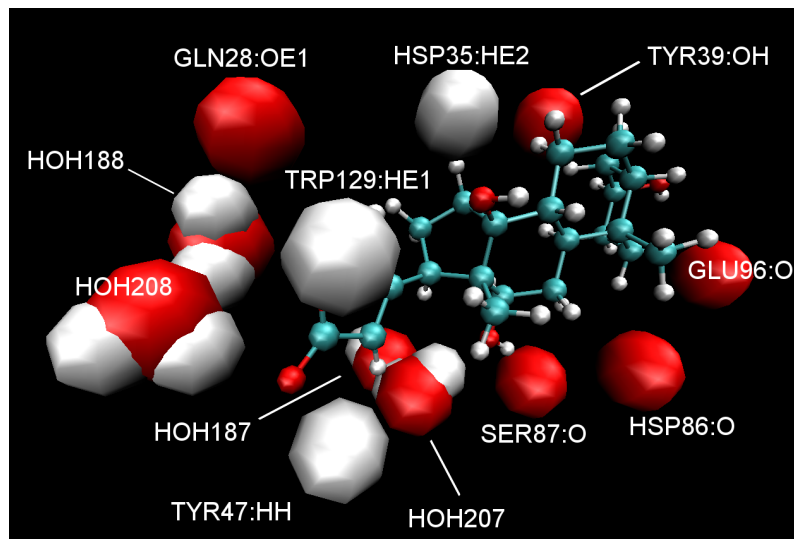
| Aminohappo | Muita konformaatioita | Aminohappo | Muita konformaatioita |
|------------|--|------------|--|
| GLY6 | vähän β -levy | LEU102 | α -heliksi |
| ASN16 | α -heliksi | ASN106 | vasen α -heliksi |
| GLY24 | aina kielletyllä | TYR117 | α -heliksi |
| ALA31 | β -levy | GLU119 | α -heliksi |
| TYR39 | vähän α -heliksi ja β -levy | LYS121 | α -heliksi, vasen α -heliksi |
| GLY40 | aina kielletyllä | LYS122 | α -heliksi |
| CYS42 | vasen α -heliksi | GLY123 | β -levy |
| GLY43 | β -levy | LEU127 | β -levy |
| GLY51 | aina kielletyllä | MET135 | vähän α -heliksi ja β -levy |
| GLY64 | aina kielletyllä | GLY151 | vähän α -heliksi ja β -levy |
| GLU70 | β -levy | PRO153 | α -heliksi |
| GLY71 | β -levy | GLN158 | α -heliksi |
| GLY77 | aina kielletyllä | LYS159 | α -heliksi |
| ILE81 | α -heliksi | SER166 | α -heliksi |
| GLY82 | aina kielletyllä | ALA168 | vähän α -heliksi ja β -levy |
| GLY92 | vähän β -levy | LYS171 | aina kielletyllä |



Kuva 33: Simulaation “prot1” aminohapot 65-76, jotka muodostavat yhden kahdeksasta β -tynnyrin β -levystä. Sinisellä alkuorientaatio ja punaisella läpikuultavalla loppuorientaatio. Huomaa alkuorientaatiossa 2-osainen β -levy.

Taulukko 2: Simulaation “prot2” loppukonformaation Ramachandran kaaviossa kielletyllä alueella olevien aminohappojen konformaatiot simulaation ajalta.

| Aminohappo | Muita konformaatioita | Aminohappo | Muita konformaatioita |
|------------|--|------------|--|
| ASP5 | vähän α -heliksi ja β -levy | ASP118 | vähän α -heliksi |
| TRP19 | α -heliksi | GLU119 | α -heliksi, vähän β -levy |
| SER20 | α -heliksi | LYS121 | vähän vasen α -heliksi |
| GLY24 | aina kielletyllä | LYS122 | α -heliksi |
| GLY40 | aina kielletyllä | GLY123 | vähän β -levy |
| GLY43 | vähän β -levy | GLY139 | aina kielletyllä |
| GLY51 | aina kielletyllä | GLU140 | α -heliksi |
| HIS63 | vasen α -heliksi | ALA141 | α -heliksi |
| ALA64 | aina kielletyllä | LYS142 | α -heliksi |
| GLY71 | β -levy | TYS148 | α -heliksi |
| GLY77 | aina kielletyllä | GLY151 | vähän α -heliksi ja β -levy |
| SER59 | α -heliksi | VAL154 | α -heliksi, vähän β -levy |
| LYS80 | α -heliksi | LYS159 | α -heliksi, vähän β -levy |
| GLY82 | vähän vasen α -heliksi | PHE165 | α -heliksi, vähän β -levy |
| GLY91 | vasen α -heliksi | GLU167 | α -heliksi |
| GLY92 | β -levy | ALA168 | α -heliksi |
| ASP105 | α -heliksi | CYS170 | α -heliksi, β -levy |
| ASN106 | aina kielletyllä | LYS171 | α -heliksi |
| LYS107 | α -heliksi | SER178 | β -levy |
| ASN108 | α -heliksi | | |

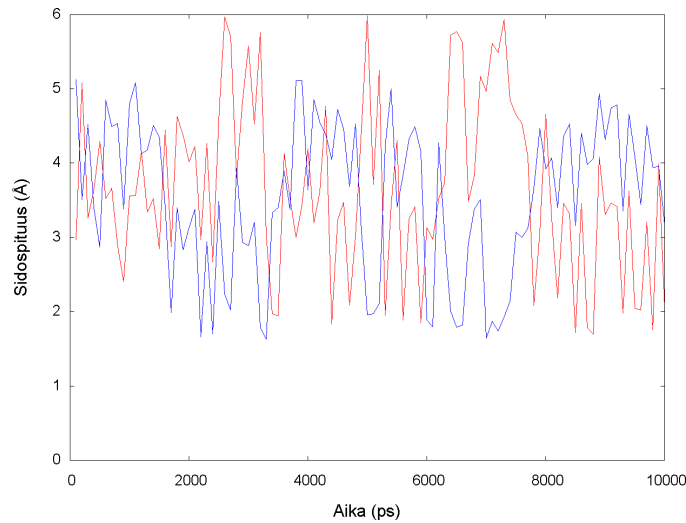


Kuva 34: Digoksigeniinimolekyylä (pienet pallot) ja lähellä olevat vety- ja happiatomit “prot3”-simulaation loppurakenteessa. Punaiset ovat happea, valkoiset vetyä ja turkoosit hiiltä, HOH-alkuiset ovat DigA16:n onkalossa loukussa olevia vesimolekyylejä. Onkalo avautuu oikealle ylös.

($-COO^-$ -ryhmässä). Käyristä näkyy selvä symmetria, kun toinen happi on kauempana on toinen lähempänä. Vesimolekyylä näyttää olevan suurimman osan ajasta sidottu kumpaankin, sekä proteiiniin että vierasmolekyyläin, ja se heilahtelee näiden välissä kuten massakappale kahden jousen varassa. Vesimolekyylä on keskimäärin 3,6 Å etäisyydellä kummastakin, tosin se heilahtelee 2-6 Å alueella.

Lähistöllä on toinenkin vesimolekyylä, tunnukseltaan “HOH 187”, joka liikkuu digoksigeniiniin ja HOH 207:n väliin silloin, kun HOH 207 on kaukana digoksigeniinistä. Ilmeisesti HOH 187 muodostaa tällöin vetysidoksen kumpaankin, digoksigeniiniin (O12:n vieressä olevaan vetyyn HAA) ja HOH 207:aan. Tällöin muodostuu kahden vesiatomin ketju digoksigeniinistä DigA16:aan. Kumpikin vesimolekyylä pysyy koko simulaation ajan lähistöllä eikä liiku kovin kauaksi. Muut atomit, jotka osallistuvat alueen vetysidoksiin ovat aminohapon SER69 “OG”-happi (aminohapon sivuryhmässä) ja saman aminohapon “O”-happi.

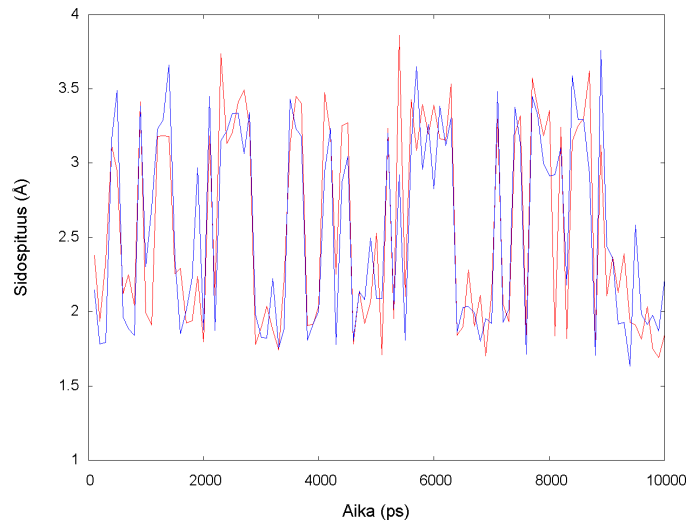
Toisesta päästä digoksigeniiniä löytyy pysyvä yhden vesimolekyyläin (HOH 208) muodostama silta digoksigeniiniin ja proteiiniin välillä. HOH 208 on toisesta vetyatomistaan sitoutunut digoksigeniiniin happiatomiin “O23” ja toisesta proteiiniin aminohapon TRP26 atomiin “O”. Kuvassa 36 näkyvät sidospituudet, joiden mukaan HOH 208 pysyy koko ajan digoksigeniiniin ja proteiiniin puolella välissä. Sidospituudet ovat koko ajan välillä 2-3,5 Å eli sidokset pysyvät koko simulaation ajan. Toinen alueella oleva vesimolekyylä (HOH 188) muodostaa vetysidoksia proteiiniin ja HOH 208:n kanssa. Kuvassa 37 on esitetty HOH 188:n sidospituudet HOH 208:n ja proteiiniin aminohapon ARG58 atomiin “HE” (sivuryhmässä) kanssa. Käyristä näkee, että sidos proteiiniin kanssa on pysyvä, mutta vesimolekyyläin kanssa hieman häilyvä. Vesimolekyylä HOH 188 tuntuukin heilahtelevan enemmän, kuin naapu-



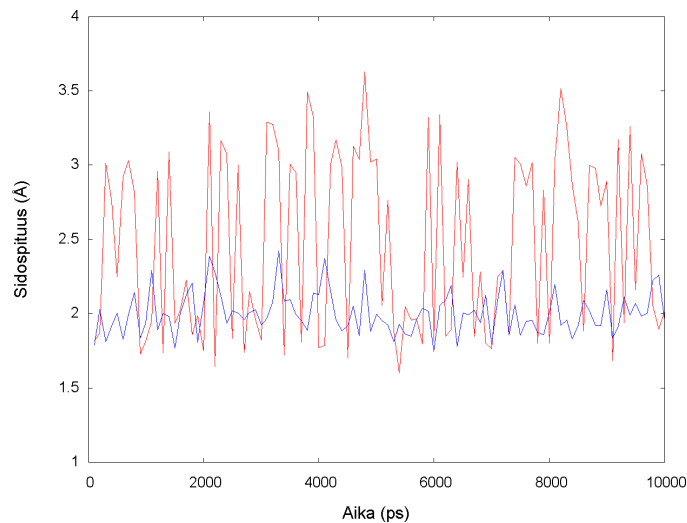
Kuva 35: Vesimolekyylin “HOH 207” sidoksetäisyydet vierasmolekyyliin (digoksigeniini) ja proteiiniin (DigA16) simulaatiossa “prot3”. Punainen käyrä on etäisyys vesimolekyylin vetyatomista digoksigeniinin happiatomiin “O12” ja sininen käyrä on etäisyys vesimolekyylin toisesta vetyatomista DigA16:n aminohapon GLU70 happiatomiin “O”.

rinsa ja muodostaa lyhytkestoisia sidoksia useiden proteiinin atomien kanssa. HOH 188 käy sidospituuden päässä seuraavista proteiinin atomeista: aminohapon GLN28 atomi “OE1” (sivuryhmässä), aminohapon ARG58 atomi “HG1” (sivuryhmässä), aminohapon TYR59 atomi “O” ja aminohapon TRP44 atomi “O”.

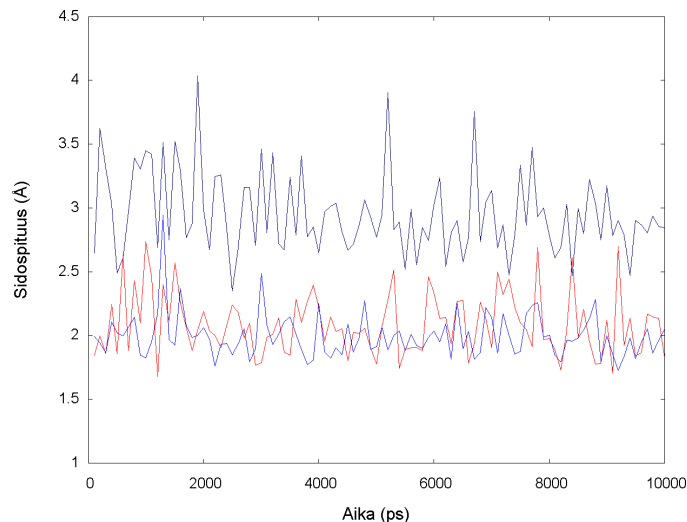
Näiden lisäksi digoksigeniini muodostaa vetysidoksia suoraan proteiinin kanssa. Kuvassa 38 on näistä kolmen sidoksetäisyydet simulaation ajalta. Kaksi näistä etäisyyksistä (aminohapon TYR47 ja HSP35 kanssa) näyttävät pysyvän hyvin vakioina, kun taas yksi (aminohapon TYR39 kanssa) vaihtelee voimakkaasti. TYR39:n tapauksessa kyseessä on heikko sidos, eikä sidos ole koko ajan olemassa vaan murtuu ja muodostuu uudelleen välillä. Muidenkin vastaavia paikkoja, missä vetysidos digoksigeniinistä proteiiniin on todennäköisesti ainakin ajoittain olemassa on lueteltu taulukossa 3. Näissä tapauksissa atomien etäisyydet vaihtelevat voimakkaastikin, mutta ovat ajoittain vetysidoksen etäisyydellä toisistaan. Tällöin on todennäköistä, että välille on muodostunut lyhytaikainen vetysidos.



Kuva 36: Vesimolekyylin “HOH 208” sidoksetäisyydet vierasmolekyyliin (digoksigeniini) ja proteiiniin (DigA16) simulaatiossa “prot3”. Punainen käyrä on etäisyys vesimolekyylin vetyatomista digoksigeniinin happiatomiin “O23” ja sininen käyrä on etäisyys vesimolekyylin toisesta vetyatomista DigA16:n aminohapon TRP26 happiatomiin “O”.



Kuva 37: Vesimolekyylin “HOH 188” sidoksetäisyydet vesimolekyyliin HOH 208 ja proteiiniin (DigA16) simulaatiossa “prot3”. Punainen käyrä on etäisyys HOH 188: vetyatomista HOH 208:n happiatomiin ja sininen käyrä on etäisyys vesimolekyylin HOH 188 happiatomista DigA16:n aminohapon ARG58 vetyatomiin HE.



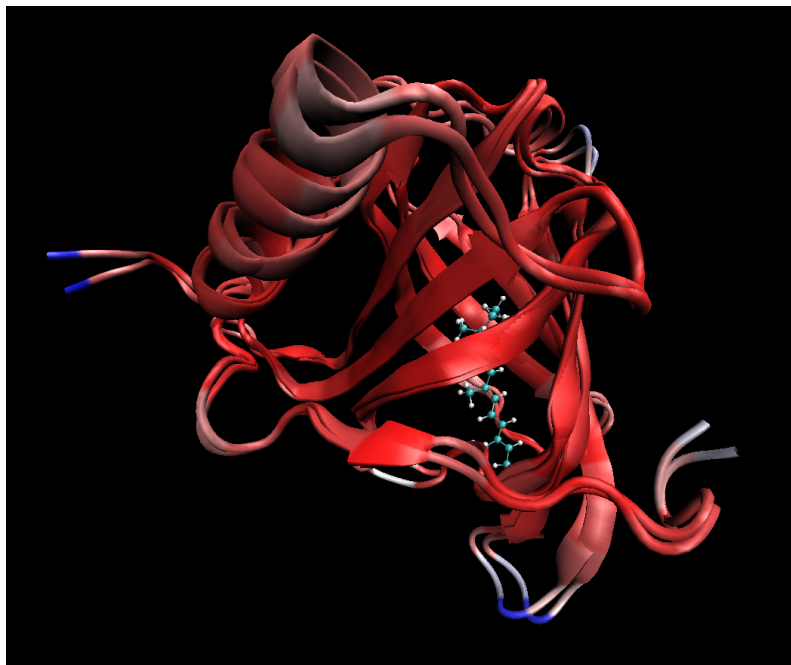
Kuva 38: Digoksigeniinin sidoksetäisyyksiä proteiiniin simulaatiossa “prot3”. Punainen käyrä on etäisyys digoksigeniinin atomista “O23” proteiinin aminohapon TRP47 atomiin “HH”, sininen käyrä on etäisyys digoksigeniinin atomista “O14” proteiinin aminohapon HSP35 atomiin “HE2” ja musta käyrä on etäisyys digoksigeniinin atomista “1H3” proteiinin aminohapon TYR39 atomiin “OH”.

Taulukko 3: Atomipareja simulaatiossa “prot3”, missä atomit ovat ajoittain vetysidoksen etäisyydellä toisistaan. Proteiinin atomin listauksessa on ilmoitettu ensin atomin aminohapon 3-kirjaiminen lyhenne ja aminohapon numero polypeptidiketjussa, jonka jälkeen on kaksoispisteellä erottuna atomin lyhenne aminohapossa. Digoksigeniinin listauksessa on pelkkä atomin lyhenne (ks. kuva 7). O-kirjainta sisältävät nimet ovat happiatomeita ja H-kirjainta sisältävät vetyatomeita.

| Atomi proteiinissa | Atomi digoksigeniinissä |
|--------------------|-------------------------|
| GLN28:OE1 | 2H21 |
| HSP86:O | 1H1 |
| SER87:O | HAA |
| GLU96:O | HAC |
| TRP129:HE1 | O21 |

3.2 1KT7-proteiinin simulaatio

3.2.1 Energetiikka

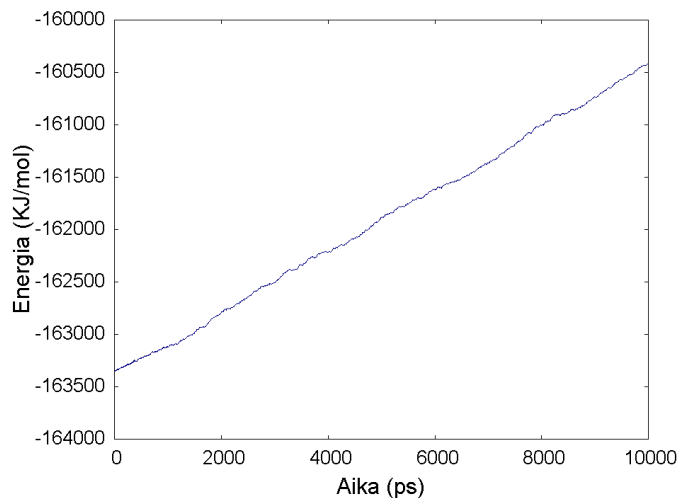


Kuva 39: 1KT7-proteiini väritettynä kuten DigA16 kuvassa 26. Sisällä retinolimolekyylä.

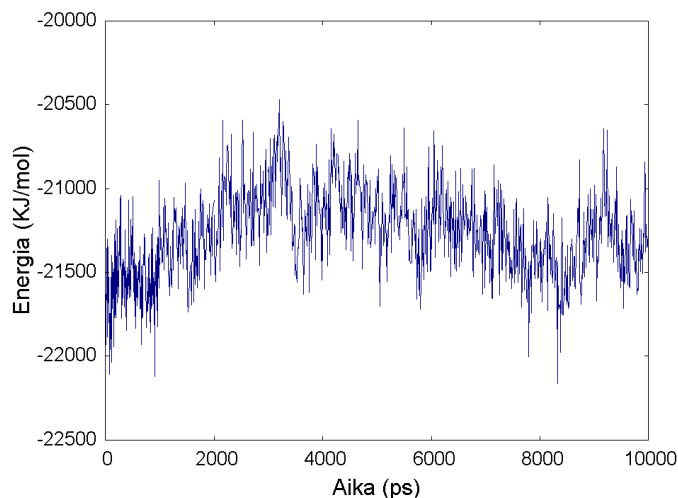
1KT7-proteiinin simulaatiossa kävi samoin, kuin DigA16:n “prot3”-simulaation kanssa eli simulaation suorittamiseen käytetyn tietokoneen vaihdon vuoksi tarkkaa tiheään tallennusvälin (9,6 ps) trajektoria ei ole saatavilla. Muuta dataa voi kuitenkin käyttää. 1KT7:n konformaatio näyttää vakaammalta kuin DigA16. Kuvasta 39 näkee, että käytännössä ainoat osat, jotka simulaation aikana liikkuiivat olivat aivan reunimmaisets aminohapot alku- ja loppupäässä proteiinia sekä muutama ulkoneva silmukka. Samalta näyttää proteiinien koordinaattien RMS-poikkeamat (kuvat 22, 23 ja 24), missä koko proteiinin, proteiinin rungon ja β -tynnyrin alueen RMS-poikkeamat ovat pienempiä, kuin DigA16:n simulaatioissa. Erityisesti rungon RMS-poikkeama pysyy hyvin pienenä koko simulaation ajan. Huomautuksena mainittakoon, että tässä proteiinissa ei ole samanlaisia pitkiä häntiä, kuin DigA16:ssa, jotka voivat liikkua helposti ympäriinsä. Joka tapauksessa 1KT7 käyttäytyy huomattavasti vakaammin simulaatioissa kuin DigA16. Kuvassa β -tynnyri on vähemmän punainen, kuin DigA16:n simulaatioissa, johtuen suhteellisesta väriskaalasta.

Simulaation kokonaisenergia (kuva 40) käyttäytyy hyvin. Se nousee alle 2% alkuarvosta simulaation aikana. Nousu johtuu kumuloituvasta pyöristysvirheestä, kuten DigA16:n tapauksessakin. Proteiinin sisäinen sekä proteiinin ja veden välinen potentiaalienergia näyttävät jokseenkin korreloivan keskenään (kuvat 41 ja 42). Tässäkin tapauksessa proteiinin atomeiksi luetaan myös vierasmolekyylä. Huomattavaa on, että potentiaalienergiat käyttäytyvät päinvastoin verrattuna DigA16:aan. Proteiinin sisäinen energia kasvaa aluksi, kun

taas DigA16:n laskee ensin.



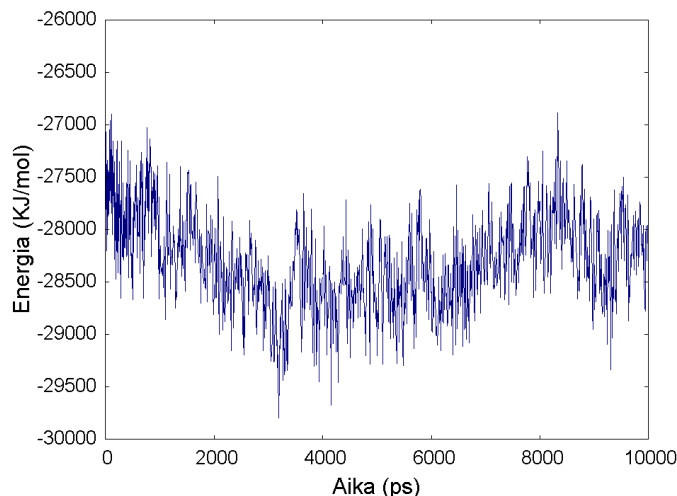
Kuva 40: 1KT7-proteiinin simulaation kokonaisenergia. Pystyakselilla energia (KJ/mol) ja vaaka-akselilla aika (ps). Huomaa energia-akselin skaalaus. Energia nousee simulaation aikana alle 2%.



Kuva 41: 1KT7-simulaation proteiinin potentiaalienergia. Skaalaus kuten kuvassa 40.

3.2.2 Konformaatio

Ramachandran kaaviosta (kuva 43) näkee, että α -heliksi- ja β -levykonformaatioalueiden välissä olevat aminohapot liikkuvat kohti α -heliksialuetta. Muuten Ramachandran kaavioissa alkua- ja loppukonformaatioille ei näy muita muutoksia kuin tavallista pientä satunnaisvaihtelua. Myös 1KT7:n Ramachandran kaaviosta tutkittiin loppukonformaatioissa



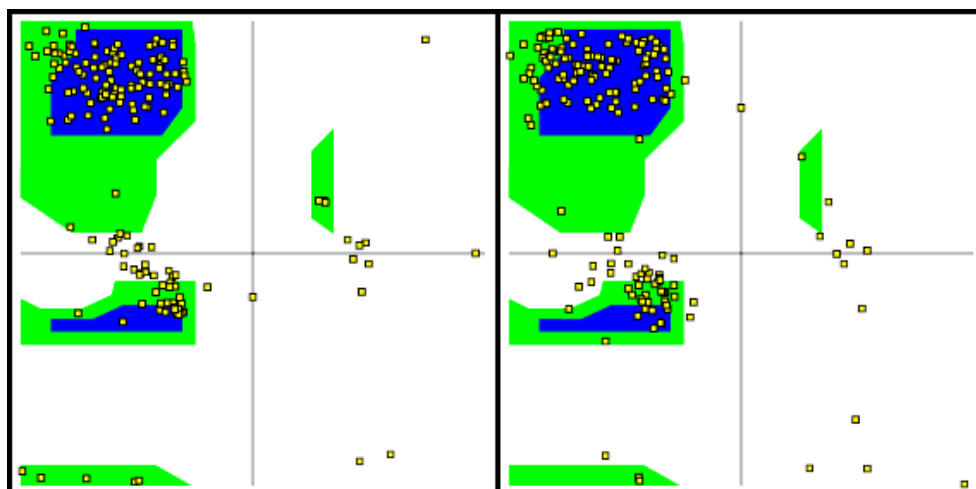
Kuva 42: 1KT7-simulaation proteiinin ja liuksen välinen sidosenergia. Skaalaus kuten kuvassa 40. Tässäkin simulaatio-ohjelma laskee retinolimolekyylin kuuluvaksi proteiiniin ja käyrä kuvaa energiaa proteiinin ja veden sekä retinolin ja veden muodostamissa sidoksissa kuten kuvassa 21.

kielletyllä alueella olleiden aminohappojen konformaatiot koko simulaation aikana. Taulukossa 4 ovat nämä aminohapot ja alueet, millä ne ovat liikkuneet. Yleisesti ottaen 1KT7:n aminohapoilla tuntuu olevan pienemmät liikkuma-alueet. Aminohapoissa ei ollut samankaltaisia “vaeltajia” kuin mitä DigA16:ssa löytyi. Yksi yhtäläinen ryhmä DigA16:n kanssa löytyi ja se on aminohapot, jotka ovat juuri α -heliksirakenteen alueen ulkopuolella loppukonformaatioissa ja muina aikoina enimmäkseen α -heliksin alueella.

3.2.3 Vierasmolekyylin käyttäytyminen

Retinoli näyttää muodostavan huomattavasti vähemmän vetysidoksia proteiinin kanssa kuin digoksigeniini DigA16:n simulaatioissa. Tämä oli odotettavissa, sillä retinoli on hydrofobinen molekyyli ja 1KT7:n onkalo on myös hydrofobinen. 1KT7:n onkalossa ei ole lainkaan vesimolekyylejä loukussa vierasmolekyylin ja proteiinin välissä. Retinoli muodostaa vetysidoksia vain onkalon suuaukon läheisyydessä, jossa retinolin atomi “O1” (kuva 8) muodostaa kolmen vetyatomien kanssa sidoksia (kuva 44). Kahden näistä kanssa sidos näyttää pysyvältä (aminohapon GLN98 atomit “H” ja “HB2”) ja kolmannen (LEU97:n atomi “HD23”) kanssa sidos on ajoittain olemassa. GLN98:n “H” on kiinnittynyt aminohapon rungon typpiin ja “HB2” ensimmäiseen sivuryhmän hiileen. Retinolin happeen “O1” kiinnittynyt vetyatomi “HAA” ei näytä muodostavan pysyviä vetysidoksia minkään proteiinin hapen kanssa. Kuvassa 45 on “HAA”:ta lähimpänä oleva proteiinin happiatomin (PHE96:n “O”) ja “HAA”:n etäisyys simulaatioissa, kuvajasta näkee, ettei välille muodostu vetysidoksia. Toisaalta “HAA” on suuaukon reunalla ja moni vapaa vesimolekyyli tulee lähelle ja muodostaa hetkittäin vetysidoksen “HAA”:n kanssa.

Vetysidoksia muodostavien happiatomien tai varautuneiden vetyatomien sijaan 1KT7:n

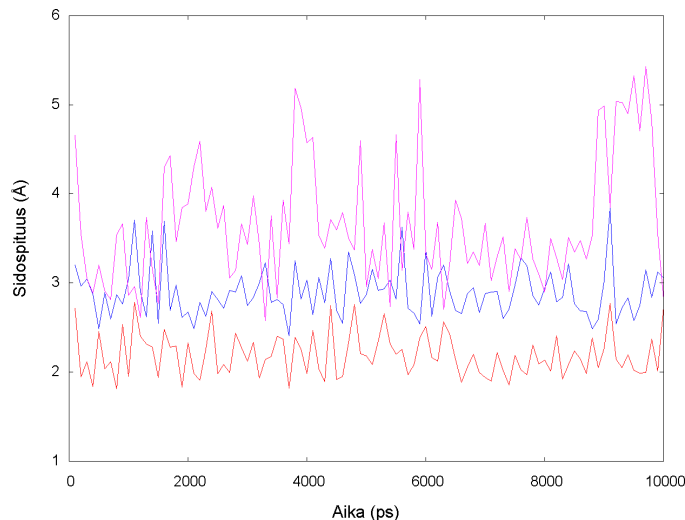


Kuva 43: 1KT7-proteiinin simulaation Ramachandran kaaviot. Vasemmalla lähtörakenne, oikealla rakenne simulaation päätyttyä. Väritys kuten kuvassa 30.

Taulukko 4: 1KT7:n simulaation loppukonformaation Ramachandran kaaviossa kielletyllä alueella olevien aminohappojen konformaatiot simulaation ajalta.

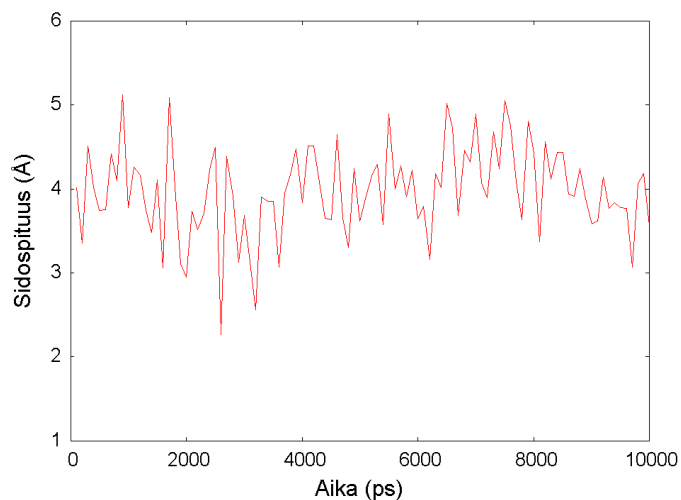
| Aminohappo | Muita konformaatioita | Aminohappo | Muita konformaatioita |
|------------|--|------------|-------------------------------|
| CYS4 | vähän β -levy | SER95 | α -heliksi |
| SER7 | α -heliksi | PHE96 | α -heliksi |
| SER8 | α -heliksi | GLY100 | vähän β -levy |
| PHE20 | α -heliksi | ASP110 | aina kielletyllä |
| GLY22 | aina kielletyllä | TYR111 | aina kielletyllä |
| MET27 | α -heliksi | GLU112 | vähän α -heliksi |
| GLY34 | vähän β -levy | ARG121 | α -heliksi |
| ASN40 | vasen α -heliksi | GLY127 | aina kielletyllä |
| GLU49 | α -heliksi | SER142 | vähän α -heliksi |
| ASN50 | vähän α -heliksi ja β -levy | GLY143 | aina kielletyllä |
| GLY51 | aina kielletyllä | VAL152 | α -heliksi |
| ASN66 | α -heliksi, vähän β -levy | LEU159 | vähän α -heliksi |
| THR80 | β -levy | CYS160 | vähän vasen α -heliksi |
| GLU81 | α -heliksi | ARG163 | aina kielletyllä |
| ALA84 | aina kielletyllä | GLY172 | aina kielletyllä |
| VAL93 | α -heliksi | | |

onkalon sisäpinnalla on runsaasti fenyyliareneja (engl. phenyl sidegroup) ja metyyliareneja (CH_3) proteiinin aminohappojen sivuryhmistä. Fenyyliareneja keskittyvät neljälle kuudelta sivusta; kuvassa 46 eteen, ylös, vasemmalle ja alas. Onkalon suuaukolla (oikealla kuvassa, retinolin ainoan happiatomin lähellä) ja yhdellä pitkällä sivulla (kuvassa takana) ei ole areneja lainkaan. Pitkällä sivulla on useita metyyliareneja ja suuaukolla proteiini muodostaa harvat vetysidoksensa retinolin kanssa.

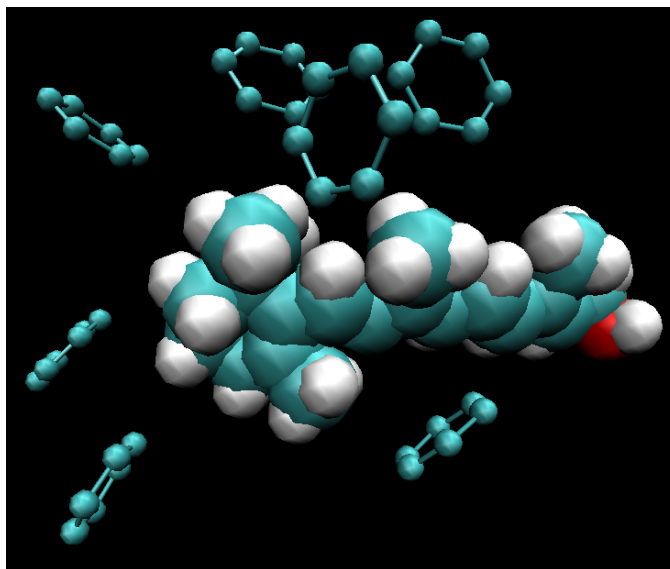


Kuva 44: Retinolin sidospituuksia proteiiniin 1KT7:n simulaatiossa. Retinolin atomista “O1” proteiinin eri atomeihin. Punainen käyrä aminohapon GLN98 atomiin “H”, sininen käyrä aminohapon GLN98 atomiin “HB2” ja purppura käyrä aminohapon LEU97 atomiin “HD23”.

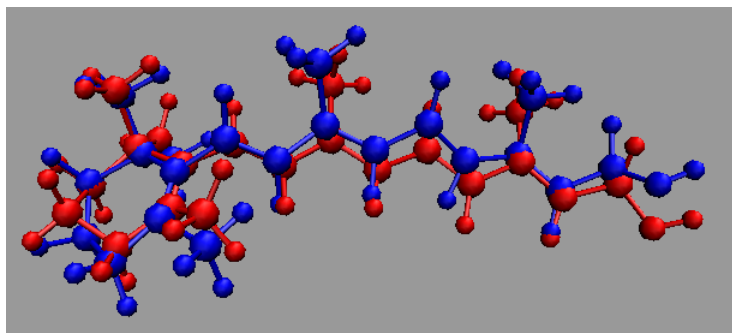
Retinolin asennonmuutosten selvittämiseksi alku- ja loppurakenteet asetettiin päällekkäin pienimmän RMS-poikkeaman mukaan, ja RMS-poikkeama laskettiin vain β -tynnyrin alueen aminohapoista. Tämän jälkeen proteiini piilotettiin ja retinoli näytettiin kummassakin rakenteessa (kuva 47). Kuvasta näkyy, että retinoli on pysynyt lähes liikkumatta koko simulaation ajan. Pitkä hiiliketju on hieman kiertynyt eri asentoon, mutta muita muutoksia ei ole. DigA16:n tapauksessa digoksigeniinin alku- ja loppuasennot olivat ehkä vielä lähempänä toisiaan. Molemmat vierasmolekyylit olivat siis hyvin vakaita koko simulaation ajan.



Kuva 45: Retinolin vetyatomin "HAA" sidoksetäisyys proteiinin aminohapon PHE96 happiatomiin "O" 1KT7:n simulaatiossa.



Kuva 46: Retinolimolekyylä 1KT7:n simulaation alkukonformaatiosta ja sitä ympäröivät fenyylirenkaat proteiinin aminohappojen sivuryhmistä. Renkaat ovat aminohapoista (myötävään, alkaen oikealta alhaalta) TYR90, PHE77, PHE45, PHE137, PHE135, TYR133, PHE36. Kuvassa hiiliatomit ovat turkoosilla, happiatomit punaisella ja vetyatomit valkoisella värillä.



Kuva 47: Retinoli alku- ja loppurakenteessa. Rakenteet on sovitettu päällekkäin pienimmän proteiinin β -tynnyrin alueen RMS-poikkeaman mukaan ja tämän jälkeen näkyviin on jätetty vain retinoli molemmista rakenteista. Sininen on alkurakenteen ja punainen loppurakenteen retinoli.

4 Johtopäätökset

Tässä työssä simuloitiin kahta lipokaliiniproteiinia, DigA16:sta ja 1KT7:ää. DigA16 on geneettisesti muokattu, koska sen vierasmolekyylin sitovan onkalon aluetta on muokattu kohdennetulla mutageneesillä. Kumpikin proteiini voi sitoa lipokaliineille tyypilliseen tapaan vierasmolekyylin, ja DigA16 sitoo digoksigeniinin ja 1KT7 retinolin. Digoksigeniini on hormoni, jota käytetään sydänlääkkeenä. Retinoli eli A-vitamiini on ihmisille ja eläimille elintärkeä vitamiini, joka vaikuttaa moneen asiaan solujen kasvusta näkökykyyn.

Simulaatioita suoritettiin yhteensä neljä. DigA16:a simuloitiin kahdella eri rakenteella ilman vierasmolekyylä ja 1KT7:a simuloitiin yhdellä rakenteella vierasmolekyylin kanssa. Eri simulaatiot on nimetty ymmärtämisen helpottamiseksi. “Prot1” on DigA16 proteiini ilman vierasmolekyylä, ja tässä simulaatiossa toinen S-S-sidos (aminohappojen 42 ja 170 välillä) jätettiin auki. “Prot2” on kuten “prot1”, mutta molemmat S-S-sidokset ovat kiinni (aminohappojen 8 ja 115 sekä 42 ja 170 välillä). “Prot3” on kuten “prot1” (eli toinen S-S-sidos auki), mutta simulaatiossa on mukana vierasmolekyylä (digoksigeniini). 1KT7:n simulaatiolle ei tarvita mitään erityistä nimeä. Tässä simulaatiossa mukana on vierasmolekyylä, retinoli.

Kaikkissa neljässä simulaatiossa suoritettiin ensin termalisointi 10 K:stä 300 K:iin, jonka kokonaiskesto oli 751 ps. Tämän jälkeen suoritettiin 10 ns *NVT*-simulaatiota 300 K:ssä. Tästä 10 ns simulaatiosta kerättiin analysoitava data. Proteiimirakenteiden visualisointiin ja analysointiin käytettiin VMD-ohjelmistoa, ja erilliset kuvaajat piirrettiin käyttäen Gnuplot-ohjelmaa.

Tulokset toivat mukanaan pari yllätystä. Ikävänä yllätyksenä tuli tarkemman trajektoridatan toimimattomuus kahdessa simulaatiossa, sillä VMD ei pystynyt lukemaan tiedostoja. Harvemmin välein ja eri formaattiin tallennettu trajektoridata toimi kuitenkin ja tästä rakennettiin trajektorin animaatio, jota päästiin tutkimaan. Positiivisena asiana oli havaitut konformaatiomuutokset simulaatioissa “prot1” ja “prot3”. Vaikka muutokset ovatkin hännän päässä, en usko että vastaavia löytyy usein. Tällaiset pienet konformaationmuutokset kuulunevat proteiinin satunnaiseen käyttöön, sillä muutos tapahtui nimenomaan siinä hännässä, mikä oli sidottu rikki-rikki-sidoksella ja siten samanlainen kuin “prot2”:ssa. Kuitenkaan “prot2”:ssa ei tapahtunut vastaavaa konformaationmuutosta.

Tynnyrimäinen rakenne pysyi kaikissa simulaatioissa muuttumattomana, ja tynnyrirungon ulkopuoliset hännät ja silmukat liikkuvat odotetusti eniten. Pitää kuitenkin muistaa, että vaikka tietokoneella lisätyissä hännissä kunkin aminohapon orientaatio valittiin energeettisesti edullisimmaksi senhetkisessä tilanteessa, ei koko hännän potentiaalienergia ollut mitenkään kontrolloitu. Tämä saattaa osaltaan selittää häntien suurta liikettä DigA16-simulaatioissa, mutta on myös mahdollista, että hännät vain liikkuvat enemmän, kun niillä ei ole mitään vakaata sekundaarista rakennetta, joka tarjoaisi selkeän energian minimiarvon, johon atomit voisivat jäädä.

Proteiinin loppupuolella olevan α -heliksin suhteellisen suuri liike simulaatioissa “prot1” ja “prot2” on hieman yllättävää, varsinkin, kun ilman rikki-rikki-sidosta ajatussa “prot1”-simulaatiossa heliksi ei liikkunut niin paljoa kuin “prot2”-simulaatiossa. On mahdollista, että rikki-rikki-sidos toimii eräänlaisena tukipisteenä, jonka vuoksi hännän loppupään lii-

ke pystyi kampeamaan heliksiä eri asentoon. Kuitenkaan vierasmolekyylin kanssa tehdyssä simulaatiossa tätä kampeamista ei havaittu, ja vierasmolekyyli näyttää siis stabiloivan proteiinia. Tämä oli kyllä odotettavissa, sillä ei proteiini sitoisi vierasmolekyyliä aktiivisesti, ellei se olisi energeettisesti edullista.

Energia käyttäytyi simulaatioissa odotetusti. Kokonaisenergia nousi hieman simulaation aikana ja nousu selittyi ohjelman sisäisellä kumuloituvalla pyöristysvirheellä johtuen äärellisestä aika-askeleesta. Yleisesti voidaan sanoa, että simulaatiot olivat vakaita. DigA16-proteiinin sisäinen energia laski, joka on merkki siitä, että proteiini siirtyi kohti energeettisesti edullisempaa asentoa simulaation aikana. 1KT7-proteiinilla sisäinen energia nousi hieman, mikä on yllättävää sillä trajektorin ja Ramachandran kaavion perusteella 1KT7 oli vakaampi kuin DigA16. RMS-poikkeaman kuvaajasta näkee, miten merkittävä rikki-rikki-sidos DigA16:n loppupään hännässä on. Ilman sidosta RMS-poikkeama kasvaa huomattavan korkealle ja nousu on jatkuvaa. Sidoksen kanssa proteiinin konformaatio pysyy huomattavasti lähempänä alkuasentoa eikä vaeltele, kuten ilman sidosta. β -tynnyriin rajoitetun RMS-poikkeaman kuvaaja paljasti, että suurin RMS-poikkeama DigA16-proteiinilla on simulaatiossa “prot2”, sitten simulaatiossa “prot1” ja pienin simulaatiossa “prot3”. Tämä näyttää siltä, että β -tynnyri stabiloituu toisen rikki-rikki-sidoksen ollessa irti ja mikäli proteiini on absorboinut digoksigeniinimolekyylin. Mikäli digoksigeniiniä ei ole ja rikki-rikki-sidos on kiinni, β -tynnyri on hieman epävakaampi. Kuitenkaan simulaatioissa ei ollut DigA16:sta sellaista rakennetta, missä olisi ollut vierasmolekyyli ja molemmat rikki-rikki-sidokset olisivat olleet kiinni. Tämän vuoksi varmaa analyysiä vierasmolekyylin ja rikki-rikki-sidoksen vaikutuksista ei voi tehdä.

Proteiinien onkaloiden sisäpinnat olivat odotetunlaisia. DigA16:lla oli paljon potentiaalisia vetysidospaikkoja, joissa useissa atomit pysyivät lähes koko simulaation ajan vetysidoksetäisyydellä vierasmolekyylistä tai onkaloon jääneistä vesimolekyyleistä. 1KT7:lla taas oli fenyylirenkaita ja metyyliiryhmiä, mikä täsmää sidottujen vierasmolekyylien tyyppieihin: digoksigeniinillä on monta hydroksyyliiryhmää valmiina vetysidoksiin, kun taas retinolissa on tasan yksi happiatomi. Vierasmolekyylit pysyivät molemmissa simulaatioissa (“prot3” ja 1KT7) hyvin vakaina, eikä mainittavia asennonmuutoksia tapahtunut.

Viitteet

- [1] Thomas M. Devlin, Textbook of biochemistry with clinical correlations, 4th edition, Wiley-Liss, Inc., New York, NY, ISBN 0-471-1545-2 (1997).
- [2] Wikipedia-projektin osanottajat. Glysiini [Internet]. Wikipedia, 9.4.2008. Haettu 5.6.2008 08.46 osoitteesta <http://fi.wikipedia.org/w/index.php?title=Glysiini&oldid=4602852>.
- [3] Wikipedia-projektin osanottajat. Tryptofaani [Internet]. Wikipedia, 30.5.2008. Haettu 5.6.2008 08.50 osoitteesta <http://fi.wikipedia.org/w/index.php?title=Tryptofaani&oldid=4868118>.
- [4] Tulane University, New Orleans. Biokemian kurssimateriaali, [Internet]. Haettu 23.6.2008 11.36 osoitteesta <http://www.tulane.edu/~biochem/med/second.htm>.
- [5] Wikipedia-projektin osanottajat. Beta sheet [Internet]. Wikipedia, 20.5.2008. Haettu 23.6.2008 08.21 osoitteesta http://en.wikipedia.org/w/index.php?title=Beta_sheet&oldid=213824360.
- [6] Tieteen tietotekniikan keskus (CSC). Bioinformatiikan sanasto, [Internet]. Haettu 6.6.2008 14:16 osoitteesta <https://extras.csc.fi/biosciences/sanasto/html/>.
- [7] S. Schlehuber ja A. Skerra. *Lipocalins in drug discovery: from natural ligand-binding proteins to "anticalins"*. Drug Discovery Today **10**, pp. 23-33 (2005).
- [8] Marc Ostermeier, Andrew E. Nixon, Jae Hoon Shim ja Stephen J. Benkovic. *Combinatorial protein engineering by incremental truncation*, Proc. Natl. Acad. Sci. **96**, No. 7, pp. 3562-3567 (1999).
- [9] PDB ID: 1LKE, I.P. Korndörfer, S. Schlehuber ja A. Skerra, *Structural mechanism of specific ligand recognition by a lipocalin tailored for the complexation of digoxigenin*, J. Mol. Biol. **330**, pp. 385-396 (2003).
- [10] PDB ID: 1KT7, V. Calderone, R. Berni ja G. Zanotti, *High-resolution structures of retinol-binding protein in complex with retinol: pH-induced protein structural changes in the crystal state*, J. Mol. Biol. **329**, pp. 841-850 (2003).
- [11] P. Procacci, E. Paci, T. Darden, M. Marchi, *ORAC: A Molecular Dynamics Program to Simulate Complex Molecular Systems with Realistic Electrostatic Interactions*, J. Comp. Chem. **18**, No. 15, pp. 1848-1862 (1997).
- [12] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov ja P.E. Bourne, *The Protein Data Bank*, Nucleic Acids Research **28**, pp. 235-242, (2000). (<http://www.pdb.org/>)

- [13] P. Procacci, T. Darden ja M. Marchi, *A very fast molecular dynamics method to simulate biomolecular systems with realistic electrostatic interactions*, J. Phys. Chem. **100**, pp. 10464-10469 (1996).
- [14] P. Procacci ja M. Marchi, *Multiple time steps algorithms for the atomistic simulations of complex molecular systems*, Proceedings of the NATO-ASI School, Erice 11-21, (1998).
- [15] M. Marchi ja P. Procacci, *Coordinates Scaling and Multiple Time Step Algorithms for Simulation of Solvated proteins in the NPT ensemble*, J. Chem. Phys. **109**, pp. 5194-5202 (1998).
- [16] D.A. Case, T.E. Cheatham III, T. Darden, H. Gohlke, R. Luo, K.M. Merz, Jr., A. Onufriev, C. Simmerling, B. Wang ja R. Woods, *The Amber biomolecular simulation programs*, J. Computat. Chem. **26**, pp. 1668-1688 (2005).
- [17] T. Darden, L. Perera, L. Li ja L. Pedersen, *New tricks for modelers from the crystallography toolkit: the particle mesh Ewald algorithm and its use in nucleic acid simulations*, Structure **7**, R55-R60 (1999).
- [18] J.M. Sanz-Serna, Acta Numerica, ISBN 0521410266, pp. 1-243 (1992).
- [19] S.K. Gray, D.W. Noid ja B.G. Sumpter, *Symplectic integrators for large scale molecular dynamics simulations: A comparison of several explicit methods*, J. Chem. Phys. **101**, pp. 4062-4072 (1994).
- [20] J.J. Biesiadecki ja R.D. Skeel, *Dangers of multiple-time-step methods*, J. Comp. Phys. **109**, pp. 318-328 (1993).
- [21] P.J. Channel ja C. Scovel, *Symplectic integration of Hamiltonian systems*, Nonlinearity **3**, pp. 231-259 (1990).
- [22] M.E. Tuckermann, B.J. Berne ja G.J. Martyna, *Reversible multiple time scale molecular dynamics*, J. Chem. Phys. **97**, pp. 1990-2001 (1992).
- [23] P. Procacci ja B.J. Barnes, *Computer simulation of solid C_{60} using multiple time-step algorithms*, J. Chem. Phys. **101**, pp. 2421-2431 (1994).
- [24] H.F. Trotter, *On the product of semi-groups of operators*, Proc. Am. Math. Soc. **10**, pp. 545-551 (1959).
- [25] H. de Raedt ja B de Raedt, *Applications of the generalized Trotter formula*, Phys. Rev. A **28**, pp. 3575-3580 (1983).
- [26] H. Yoshida, *Construction of higher order symplectic integrators*, Phys. Lett. A **150**, pp. 262-268 (1990).

- [27] S.J. Toxvaerd, *Comment on constrained molecular dynamics of macromolecules*, J. Chem. Phys. **87**, pp. 6140-6143 (1987).
- [28] M.E. Tuckermann, B.J. Berne ja G.J. Martyna, *Reply to Comment on: Reversible multiple time scale molecular dynamics*, J. Chem. Phys. **99**, pp. 2278-2279 (1993).
- [29] D. Frenkel ja B. Smit, *Understanding Molecular Simulations*, Academic Press, San Diego (1996).
- [30] D.C. Rapaport, *The Art of Molecular Dynamics Simulation*, Cambridge University Press, Cambridge (UK) (1995).
- [31] S. Nosé, Computer Simulation in Materials Science, M. Meyer ja V. Pontikis (editorit), Kluwer Academic Publishers, sivu 21 (1991).
- [32] S. Nosé, *Constant temperature molecular dynamics methods*, Prog. Theor. Phys., Supplement No.103, pp. 1-46 (1991).
- [33] M. Ferrario, Computer Simulation in Chemical Physics, M.P. Allen ja D.J. Tildesley (editorit), Kluwer Academic Publishers, sivu 153 (1993).
- [34] G.J. Martyna, D.J. Tobias ja M.L. Klein, *Constant pressure molecular dynamics algorithms*, J. Chem. Phys. **101**, pp. 4177-4189 (1994).
- [35] H.C. Andersen, *Molecular dynamics simulations at constant pressure and/or temperature*, J. Chem. Phys. **72**, pp. 2384-2393 (1980).
- [36] S. Nosé, *A molecular dynamics method for simulations in the canonical ensemble*, Mol. Phys. **52**, pp. 255 - 268 (1984).
- [37] M. Ferrario ja J.-P. Ryckaert, *Constant pressure-constant temperature molecular dynamics for rigid and partially rigid molecular systems*, Mol. Phys. **54**, pp. 587-603 (1985).
- [38] P. Procacci ja B.J. Berne, *Multiple time scale methods for constant pressure molecular dynamics simulations of molecular systems*, Mol. Phys. **83**, pp. 255 - 272 (1994).
- [39] G.N. Murshudov, A.A. Vagin ja E.J. Dodson, *Refinement of macromolecular structures by the maximum-likelihood method*, Acta Crystallog. D **53**, pp. 240-255 (1997).
- [40] W. Humphrey, A. Dalke ja K. Schulten, *VMD - Visual Molecular Dynamics*, J. Molec. Graphics **14**, pp. 33-38 (1996).
- [41] School of Crystallography, Birkbeck, University of London. Ramachandran plot, [Internet]. Haettu 10.6.2008 osoitteesta http://www.cryst.bbk.ac.uk/PPS95/course/3_geometry/rama.html.