

TARTU ÜLIKOOL
LOODUS- JA TEHNOLOOGIATEADUSKOND
MOLEKULAAR- JA RAKUBIOLOOGIA INSTITUUT
BIOINFORMAATIKA ÕPPETOOL

Triin Edula

**Raamjärjestamise programmide võrdlus eelnevalt kokkupanud genoomijärjestuste
alusel**

Bakalaureusetöö

Juhendaja PhD Reidar Andreson

TARTU 2014

Sisukord

Kasutatud lühendid ja mõisted.....	4
Sissejuhatus.....	6
1. Kirjanduse ülevaade.....	7
1.1 Sekveneerimine.....	7
1.1.1 Esimese põlvkonna sekveneerimismeetod.....	8
1.1.2 Järgmise põlvkonna sekveneerimismeetodid.....	9
1.1.2.1 454 pürosekveneerimine.....	11
1.1.2.2 Solexa/Illumina.....	12
1.1.2.3 SOLid System.....	13
1.2 Genoomide kokkupanemine.....	15
1.3 Genoomide kokkupanemise programmid.....	17
1.3.1 SSAKE.....	17
1.3.2 SHARCGS.....	18
1.3.3 Velvet.....	20
1.3.4 SGA.....	21
1.3.5 Soapdenovo2.....	23
1.4 Raamjärjestamine.....	24
1.5 Raamjärjestamise programmid.....	26
1.5.1 Bambus.....	26
1.5.2 SSPACE.....	27
1.5.3 MIP Scaffolder.....	28
1.5.4 GRASS.....	29
1.5.5 SCARPA.....	29
1.5.6 L_RNA_Scaffolder.....	30
1.6 Parameetrid genoomide kokkupanemise kvaliteedi hindamiseks.....	32
1.6.1 Kvantitatiivsed parameetrid.....	33
1.6.2 Kvalitatiivsed parameetrid.....	35
2. Uurimus.....	38
2.1 Töö eesmärgid.....	38
2.2 Materjal ja metoodika.....	38
2.2.1 Kasutatavad raamjärjestamise programmid.....	38

2.2.2 Kasutatavad andmed.....	39
2.2.3 Kasutatavad parameetrid.....	40
2.3 Tulemused ja arutelu.....	40
Kokkuvõte.....	42
Summary.....	43
Kasutatud kirjandus.....	44
Kasutatud veebiaadressid.....	48

Kasutatud lühendid ja mõisted

BAC – (*Bacterial Artificial Chromosome*) – bakteri kunstlik kromosoom

bp – (*base pair*) – aluspaar

cDNA – (*complementary DNA*) – mRNA suhtes komplementaarne DNA ahel

ddNTP – (*dideoxynucleotide*) – didesoksünukleotiid – Sangeri ensümaatilisel sekveneerimismeetodil kasutatav trifosfaadi analoog

FASTA - tekstipõhine nukleotiidsed ja aminohappelise järjestuse esitusviis, milles nukleotiididele ja aminohapetele vastavad kindlad märgised

FASTQ – tekstipõhine nukleotiidsed ja aminohappelise järjestuse talletusviis, mis sisaldab ka infot vastavate järjestuste kvaliteedi kohta

Gbp – (*giga base pair*) – miljard aluspaari

Genoomi kokkupanemine – (*assembling*) – sekveneerimisandmete alusel genoomi taaskonstrueerimise protsess

Genoomi kokkupanemise programm – (*assembler*) – vahend sekveneerimisandmete alusel genoomi taaskonstrueerimiseks

HGP – (*the Human Genome Project*) – „Inimese genoomi projekt“

kbp – (*kilo base pair*) – tuhat aluspaari

KL – kaaslugemid – (*mate pair reads*) – DNA fragmendi mõlema otsa sekveneerimise teel saadud järjestused. Analoogsed paarisotsaliste lugemitega, kuid suurema inserdisuurusega

K'mer - kindlaksmääratud pikkusega alamjärjestused lugemist

Kokkupandud genoom – (*assembly*) – sekveneerimisandmete alusel taaskonstrueeritud genoom

Mbp – (*mega base pair*) – miljon aluspaari

NGS – (*Next Generation Sequencing*) - järgmise põlvkonna sekveneerimine

OLC – (*Overlap Consensus Layout*) – graafi meetod, mis konstrueeritakse esimese põlvkonna sekveneerimisandmete alusel

Paarislugemid – (*paired read*) – koondnimetus paarisotsalistele ja kaaslugemitele

PL – paarisotsalised lugemid (*paired-end reads*, PE) – DNA fragmendi mõlema otsa sekveneerimise teel saadud järjestused. Analoogsed kaaslugemitega, kuid väiksema inserdisuurusega

Raamjärjestamine – (*scaffolding*) – kontiigide ühendamisprotsess pikemateks liitjärjestusteks

Raamjärjestus – (*scaffold*) – kontiigidest ja kontiigide vahele jäävatest tühimikest koosnev liitjärjestus

Raamjärjestamise programm – (*scaffolder*) – vahend kontiigide ühendusprotsessi läbiviimiseks

ÜL – üheotsaline lugem (*single-end read, SE*) – DNA fragmendi ühe otsa sekveneerimise teel saadud järjestus

Sissejuhatus

Esimene täielikult sekveneeritud genoom kuulub bakteriofaagile Φ -X174 (Godson jt., 1987). Edukas viiruse genoomi DNA järjestuse kindlaks tegemine kannustas ette võtma mahukamaid projekte ja üks kulukamaid nende seas on „Inimese genoomi projekt“ (International Human Genome Sequencing Consortium, 2004). Tänapäevaks on välja töötatud järgmise põlvkonna sekveneerimismeetodid (NGS), mis nõuavad üha vähem ajalisi ja rahalisi ressursse. Selline soodus olukord on vallandanud sekveneerimisbuumi, mis on tõstatanud uue probleemi – mida teha saadud andmetega? Üha rohkem tuntakse vajadust sekveneeritud DNA järjestusi analüüsida ja tõlgendada. Paralleelselt sekveneerimismeetodite arenguga proovitakse välja töötada üha efektiivsemaid programme andmete töötlemiseks. Spetsiifilisele andmemassiivile sobiva programmi leidmine on oluline ülesanne võimalikult täpseks andmetõlgenduseks.

Käesolev bakalaureusetöö koosneb teoreetilisest osast ja tegevuskavandist. Töö teoreetiline pool annab lühiülevaate sekveneerimismeetoditest ja laialdasemalt kasutusel olevatest sekveneerimistulemuste (DNA järjestuste) töötlusprogrammidest. See töö keskendub eelkõige andmetöötluse viimasel etapil kasutatavate genoomi kokkupanemistulemuste parandamiseks mõeldud tarkvarade (raamjärjestamise programmid) efektiivsusega seotud võrdlemisele. Selle uurimise jaoks moodustati nimekiri parameetritest, millega saab raamjärjestamise programmide tõhusust hinnata ning koostati tegevuskava hindamise realiseerimiseks. Töös keskenduti just DNA sekveneerimistulemuste kokkupaneku analüüsile.

Käesolev töö on valminud Tartu Ülikooli molekulaar- ja rakubioloogia instituudi bioinformaatika õppetoolis.

Märksõnad: DNA sekveneerimine, genoomide kokkupanemine, raamjärjestamine

1. Kirjanduse ülevaade

1.1 Sekvenerimine

Sekvenerimine (*sequencing*) tähendab bioloogilises ja biokeemilises kontekstis meetodit, mis võimaldab määrata biopolümeeride (valkude ja nukleiinhapete) primaarstruktuuri. Sekvenerimise tulemiks on järjestus, mis sisaldab endas kindlatest biopolümeeridest koosnevat jada. Käesoleval ajal pööratakse eraldi tähelepanu eksoomi ehk valke kodeeriva ala järjestuse uurimisele, kuna geenijärjestuste kokkupanek on hõlpsamini teostatav ja lihtsam on uurida nende seoseid bioloogiliste protsesside või haigustega. Sekvenerimine on biomeditsiini jaoks perspektiivne kahel järgneval põhjusel. Esiteks, uued ja efektiivsed sekvenerimismeetodid muutusid laiemale teadlaskonnale kättesaadavaks alates 2004. aasta lõpust. Tänapäeval on võimalik tänu järgmise põlvkonna meetoditele sekvenerida suurte genomikeskuste kõrval ka väiksemates kliinilistes laborites (Moorthie jt., 2011). Teiseks, inimese genoomi resekvenerimine on järjest hoogustumas (Kidd jt., 2008; Wang jt., 2008). Seega on tulevikus võimalik, et seniselt ravisüsteemilt minnakse üle personaalsele meditsiinile. Patsiendi eksoom või genoomne DNA esmalt sekveneritakse ja hiljem analüüsitakse konkreetse inimese nukleotiidset järjestust ning antakse lähtuvalt tulemustest patsiendile ravisoovitusi. Säärane täpsem diagnostika võiks hõlbustada arstidel diagnooside määramist ja olla pikemas väljavaates inimese eluea tõstjaks.

Erinevate biopolümeeride sekvenerimine annab vastuseid erinevatele bioloogilistele küsimustele. DNA sekvenerimine on laboritingimustes rakendatav meetod DNA primaarjärjestuse määramiseks. Organismi nukleotiidilise järjestuse lahtidešifreerimine annab vajalikku informatsiooni geneetilise profiili koostamiseks ja teadmisi, mida saab rakendada meditsiini valdkonnas või ka evolutsiooni uurimisel. DNA sekvenerimine on võrreldes teiste biopolümeeride järjestusmeetoditega ajaliselt kõige vanem. 1977. avaldasid F. Sanger, S. Nicklen ja A. R. Coulson artikli DNA sekvenerimisest ahela terminatsiooni abil (Sanger jt., 1977). RNA uurimisel saadakse informatsiooni ekspresseeritavate geenide kohta. Selle uurimismeetodi jaoks kasutatakse viiruslikku päritolu ensüümi pöördtranskriptaasi, mis sünteesib eelnevalt ülejäänud RNA-st eraldatud mRNA alusel temale vastava komplementaarse DNA (cDNA). Edasiselt toimitakse cDNA järjestamise puhul samamoodi nagu DNA sekvenerimisel. Valkude sekvenerimine aitab tuvastada valgu struktuuri ning funktsiooni, mis on vajalik rakuliste protsesside mõistmiseks ning teades metabolismiradu, on

võimalik tõhusam ravimite väljatöötamine. Valkude sekveneerimise meetodika on nukleiinhapete omast erinev. Tänapäeval saab peptiidjärjestusi tuvastada automatiseeritud Edmani degradatsioonil, kus sammhaavalisel valgumolekuli N-terminuse degradeerimisel saadakse informatsiooni aminohappejääkide kohta (Niall, 1973). Teiseks valkude analüüsimeetodiks on massispektromeetria (MS). Teades nii erinevate aminohappe kombinatsioonide kogumasse kui ka huvipakkuva valgu massi, on võimalik nende alusel välja arvutada uuritava valgu järjestus.

Täna sel päeval on sekveneerimine jagunenud kahte suunda – esimene (Sangeri ahela terminatsioon) ja järgmine põlvkond (NGS). Nõudlus efektiivsemate tehnoloogiate järele tekkis juba „Inimese genoomi projekti“ kestel (Zhang jt., 2011). Sangeri meetod on kahtlemata oluline saavutus bioteaduste vallas, kuid perspektiivitu kogu genoomi sekveneerimisel – meetod nõuab mahukaid ajalisi ja rahalisi ressursse (Wu jt., 2007). Nüüdseks on arendatud Sangeri meetodist järgmise põlvkonna sekveneerimismeetodid. NGSi ehk massiivse paralleelse sekveneerimise esimeseks etapiks on genoomse DNA fragmenteerimine väiksemateks lõikudeks, millest seejärel koostatakse raamatukogu. Raamatukogu lõigud kantakse kandjale, kus toimub DNA fragmentide seondumine kindlate proovidega ja kaksikahela denaturatsioon üheaahelaliseks. Edasiselt sünteesitakse uuritavale üheaahelalisele DNA järjestusele vastasahel. Iga üksiku nukleotiidi lülitamisel sünteesitavasse ahelasse saadakse valgussignaali, mis registreeritakse arvuti abiga. Sellisel viisil toimub sekveneerimine ja DNA süntees üheaegselt. Sekveneerimisprotsess on muudetud kiiremaks, odavamaks ja täpsemaks (Gritsenko jt., 2012). Üha kasvav andmete hulk, mida suudetakse sekveneerimistehnoloogiaga kiirenevas tempos luua, nõuab omakorda rohkem võimsust genoomi kokkupanemise tarkvaradelt ja ruumi järjestuste hoiustamiseks.

Järgnevalt tutvustatakse lähemalt DNA sekveneerimismeetodeid, mis on tänapäeval laialdasemas kasutuses.

1.1.1 Esimese põlvkonna sekveneerimismeetod

Esimese põlvkonna sekveneerimismeetod on eelkõige seostatav Sangeri klassikalise sekveneerimistehnoloogiaga. Aastal 1977 avaldasid F. Sanger, S. Nicklen ja A. R. Coulson artikli, milles tutvustasid uut meetodit nukleotiidide järjestuste määramiseks. Uus meetod kasutas DNA polümeraasi ja teiste nukleotiidide suhtes täpselt määratud vahekorras inhibiitornukleotiide (ddNTP), nii et tagatud oleks ddNTP-de statistiline lülitamine sünteesitavasse ahelasse. Terminaatornukleotiidi ahelasse lisamine lõpetab uue ahela sünteesi

– seetõttu kasutatakse sünonüümina ka mõistet Sangeri ensümaatiline terminaatori meetod (Sanger jt., 1977). Sangeri tehnoloogia kasutab nelja reaktsiooni läbiviimiseks üheaheelalist DNA-d, DNA polümeraasi, DNA praimerit, nelja fluorestseerivalt (algupärane meetod kasutas radioaktiivmäärgist) määrgistatud trifosfaadi analoogi (ddATP, ddGTP, ddCTP, ddTTP) ja desoksüribonukleotiide (dNTP). Fluorestsentsmäärgistus võimaldab eri nukleotiididel vahet teha. Meetod põhineb DNA *in vitro* sünteesil, mis katkestatakse didesoksüribonukleotiidi (ddNTP) sisselülitamisel sünteesitavasse ahelasse. Resultaadiks on fluorestseerivalt määrgistatud fragmentide segu (vastavalt nukleotiididele - A, G, C, T), mis on (iga nukleotiiditüübi puhul eraldi) katkestatud erinevatel positsioonidel. Asjaolu, mis võimaldab ahela katkemist, on vaba 3'OH-rühma puudumine. DNA vajab ahela pikendamiseks vaba 3'OH-rühma, kuid didesoksüribonukleotiidil (ddNTP) esineb vaid 3'H rühm. Pärast sünteesi toimub reaktsioonisegu denaturatsioon (Sanger jt., 1977). Iga nukleotiidi juures lõpeb süntees erineval kaugusel, mistõttu on ahelad erineva pikkusega ning liiguvad geelis erineva kiirusega ja see teadmine võimaldabki nukleotiidide teineteisest eristada. Enne aastat 1987 oli DNA sekveneerimine valdavalt laboripõhine ja vajab radioaktiivseid materjale. Nimelt tutvustas aastal 1987 Applied Biosystems esimest automaatset sekvenaatorit, mis põhineb kapillaarelektroforeesil (elektriliselt laetud osakeste lahutusmeetod) (Liu jt., 2012). Varasem meetodika kasutas iga nukleotiidi puhul eraldiseisvat nelja geelirada, kuid tänasel päeval on võimalik eri lainepikkusel fluorestseeruva ddNTP kaudu ühes rajas reaktsioonisegu jooksutada (Morozova ja Marra, 2008).

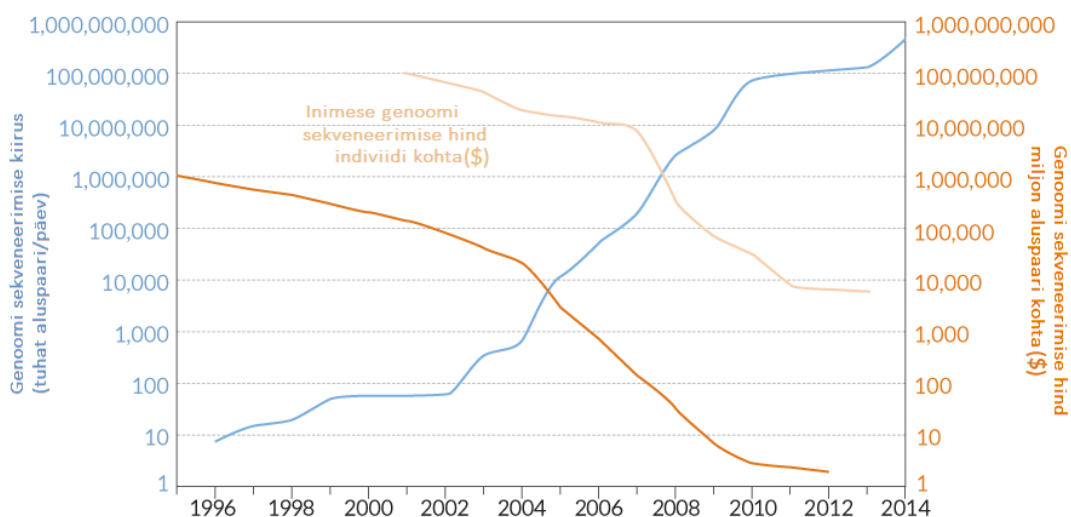
Sangeri tehnoloogiat kasutati „Inimese genoomi projektis“ – ehk inimgenoomi kaardistamisel. Projekt vältas pikalt (13 aastat) ja oli kulukas (~ 3 miljardit dollarit) (International Human Genome Sequencing Consortium, 2004). Sellest lähtuvalt tekkis vajadus odavamate, kiiremate ja suurema läbilaskevõimega tehnoloogiate järele (Liu jt., 2012).

1.1.2 Järgmise põlvkonna sekveneerimismeetodid

Uue põlvkonna sekveneerimistehnoloogiaid (454 pürosekveneerimine, Solexa/Illumina, SOLiD System) on kasutatud erinevates genoomika uurimisvaldkondades, nagu näiteks kogu genoomi ja transkriptoomi sekveneerimine, transkriptsioonifaktorite seondumissaitide avastamine, mittekodeeriva RNA ekspressiooniprofiili määramine ja suunatud sekveneerimine ehk huvi pakkuvate alade valikuline sekveneerimine (Morozova ja Marra, 2008). Uute sekveneerimistehnoloogiate kasutuselevõtt on aidanud nii biolooge

evolutsiooniliste suhete väljaselgitamisel kui ka tervishoiutöötajaid haigustekitajate geenide tuvastamisel (Liu jt., 2012).

Ideaalset sekveneerimistehnoloogiat peaks iseloomustama kiirus, täpsus, odavus ja kõrge läbilaskevõime. Järgmise põlvkonna meetodid vastavad neis enamusele, kuid lühikeste lugemite arvelt. Sekveneerimise kuldstandardiks peetakse Sangeri tehnoloogia saavutatud lugemite pikkuseks 900 bp, sellal kui Illumina sekvenaatorid väljastavad tunduvalt lühemaid kuni 150 aluspaarilisi järjestusi. Lühikesed lugemid on uuemate meetodite pudelikaelaks: raskendavad edasist andmetöötlust ja sellest tulenevat tulemuste interpretatsiooni. Esimese põlvkonna sekveneerimise tundlikuks kitsaskohaks on sekveneerimise hind, mis ajendas leidma odavamaid järjestamismooduseid (joonis 1).



Joonis 1. Sekveneerimise hind genoomi kohta.¹ Kümne aastaga (2001-2011) on toimunud 10 000 kordne hinnalangus. Aastad 2001-2007 peegeldavad Sangeri tehnoloogiaga sekveneeritud genoomi hinda. Eriti järsu pöörde langemise suunas teeb graafik 2007. aasta juures, kui teise põlvkonna sekvenaatorid jõudsid turule.

Sangeri meetodiga kulub miljon aluspaari sekveneerimiseks 2400 dollarit, kuid sama tulemuse saavutamiseks ei kulu teise põlvkonna meetodi rakendamisel üle kümne dollari. Mitte ainult hind ei ole Sangeri puuduseks, vaid ka teiste meetoditega võrreldes märkimisväärselt madalam läbilaskevõime (kuni 84 kbp ühe jooksu kohta). Järgmise

¹ <https://www.sciencenews.org/article/gene-sequencing-future-here>

põlvkonna tehnoloogiad suudavad toota kuni 600 Gbp jooksu kohta ehk võimaldavad massiivset paralleelset DNA järjestamist (Liu jt., 2012). Seega just järgmise põlvkonna sekveneerimine on aidanud viia bioloogilised ja biomeditsiinilised uuringud uuele tasemele tehes järjestamise laialdasemaks ja igapäevasemaks erinevatele uurimisrühmadele (Shendure ja Ji, 2008). Tabelis 1 on võrdluseks välja toodud andmed kolme enamlevinud järgmise põlvkonna sekveneerimismeetodite kohta.

Tabel 1. Kolme järgmise põlvkonna sekveneerimismeetodi võrdlus.^{2, 3}

Platvorm	454	SOLiD	Solexa/Illumina
Metodoloogia	Pürosekveneerimine	Sekveneerimine paralleelselt ligeerimisega	„Pööratav“ terminatsioon
Läheneemisviis DNA amplifikatsioonile	Emulsiooni PCR	Emulsiooni PCR	Sildamplifikatsioon
Sekveneerimisensüüm	DNA polümeaas	DNA ligaas	DNA polümeraas
Hind miljon aluse kohta	\$10	\$0.13	\$0.07
Maksimaalne lugemite pikkus	700 bp	50 bp (ÜL), 101 bp (PL)	150 bp
Väljundandmeid jooksutuse kohta	0.7 Gbp	120 Gbp	600 Gbp
Aeg jooksutuse kohta	24 tundi	7 päeva (ÜL), 14 päeva (PL)	3-10 päeva

1.1.2.1 454 pürosekveneerimine

Biotehnoloogia firma 454 Life Science 454 sekveneerimismeetod (sekvenaator GS 20) oli aastal 2005 esimene järgmise põlvkonna sekveneerimistehnoloogia, mis jõudis turule. See tehnoloogia kasutab emulsiooni PCR-i kloonalse amplifikatsiooni läbiviimise jaoks (joonis 2). 454 meetodi puhul, erinevalt Sangerist, on ajaliselt vähendatud kloonimise etappi, mis on väga töömahukas ja aeganõudev (Rothberg ja Leamon, 2008). 454 edukust tõestab asjaolu, et see on järjekorras teine meetod, millega on suudetud inimese genoomi sekveneerida (Wheeler jt., 2008).

² Liu jt., 2012

³ Moorthise jt., 2011

Adaptoreid kasutades on iga DNA fragment ligeeritud streptavidiinist pärlite külge. Seejärel viiakse iga fragment pikotiiterplaadi erinevatesse emulsioonitilkadesse. Tilkades toimub kõrge tootlikkusega (*ca* 10^7 referents-DNA kloonikoopiat ühe kerakese (*bead*) kohta) klonaalne amplifikatsioon (Margulies jt., 2005).

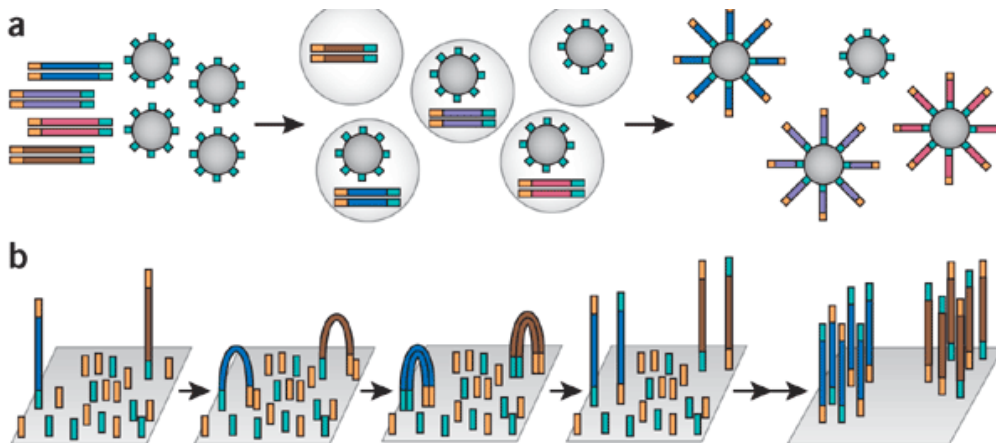
454 meetodi puhul on tegemist paralleelse massiivse sekveneerimisega (sekveneerimine toimub ajaliselt paralleelselt DNA sünteesiga) (Margulies jt., 2005), täpsustavalt pürosekveneerimisega (Rothberg ja Leamon, 2008). Sihtmärk-DNA (*template-DNA*) on mobiliseeritud ning lahuseid, mis sisaldavad dNTP-d, lisatakse ükshaaval. Iga kord, kui lisatakse reaktsioonisegusse mall-DNA-ga komplementaarne nukleotiid (antud juhul dNTP), toimub pürofosfaadi (PPi) eraldumine. Pürofosfaadi eraldumist saab detekteerida samuti reaktsioonisegus leiduva valgustundliku ensüümi lutsiferaasi abil. Ensüümi aktiivsuse, mis väljendub valguse tootmises, tuvastab kaamera ning pürogrammi (*pyrogram*) abil määratakse DNA järjestus (Morozova ja Marra, 2008).

454 meetod on võrreldes Sangeri meetodiga küll kiirem ja odavam, kuid 454 puudused on samal ajal ka Sangeri tehnoloogia tugevused. Pürosekveneerimise tulemiks on lühemad lugemid (keskmiselt 300 bp), mis teeb järgmise sammu järjestusanalüüsis – genoomi kokkupanemise – korduvate DNA motiivide tõttu keerukaks. Roche 454 pürosekveneerimise tõsiseks puuduseks on suur vigade hulk homopolümeersetes piirkondades. Homopolümeeri pikkus sõltub luminesentsist, mille käigus pürofosfaadid vabanevad. Vigade tekke põhjus peitub metodoloogias – variatsioonid valguse intensiivsustes, mille tulemusena 15% lugemitest on vigased (Gomez-Alvarez jt., 2009). (Morozova ja Marra, 2008)

1.1.2.2 Solexa/Illumina

Viimastel aastatel on järgmise põlvkonna sekveneerimistehnoloogiatest kõige laialdasemalt kasutusel olnud Illumina/Solexa meetod (Quail jt., 2012). Selle lähenemisviisi arendajaks oli Solexa, kuid turustajaks Illumina sekvenaatorite Genome Analyser ja HiSeq kaudu (Moorthie jt., 2011). Illumina tööstrateegia võimaldab tunduvalt tõhusamat sünteesi abil sekveneerimist kui Sangeri kapillaarsekvenaatorid. Esmalt fragmenteeritakse proovi DNA juhuslikest punktidest ning tulemuseks saadakse fragmendid mall-DNA-st, mille külge on ligeeritud lõpuspetsiifilised adapterid. Sellisel viisil konstrueeritakse Illumina raamatukogu, mis järgnevalt denatureeritakse üksikahelalisteks DNA-deks. Saadud ssDNA kinnitatakse tahkele kandjale, kus leiab aset nii amplifitseerimine kui sekveneerimine. DNA paljundamine toimub

PCR „silla“ abil (joonis 2b). Paindliku linkeri abil kinnitatakse tahkele kandjale nii päripidine kui ka äraspidine praimer, mis tagab, et iga amplifitseeritud DNA molekul oleks immobiliseeritud ja koondunud kindlale füüsilisele asukohale kiibil. Amplifikatsioonil ühinevad üksikahelalised DNA-d komplementaarsete adapteritega tekitades „silla“. Nii võib amplifitseerimise tulemusena saada mitu miljonit klastrit, milles igaüks koosneb umbes 1000 amplikonist.



Joonis 2. Teise põlvkonna sekveneerimistehnoloogiate DNA amplifitseerimismeetodid.

(a) Mõlema tehnoloogia (454 ja SOLiD) DNA paljundamine põhineb PCRi emulsioonil, mis leiab aset pärlite peal. (b) Illumina/Solexa tehnoloogia baseerub silla amplifikatsioonil, mille protsessi tulemusena saadakse klastrid. Raamatukogu iga üksiku liikme kohta saadakse amplifikatsioonil tuhat koopiat ehk üks klaster sisaldab 1000 amplikoni. (Shendure ja Ji., 2008)

Sekvenerimisreaktsioon algatatakse, kui lisatakse universaalne sekvenerimispraimer, mis hübridiseerub lõuspetsiifilisele adapterile. Ahela laiendamiseks kasutatakse DNA polümeraasi ja nelja modifitseeritud nukleotiidi ehk pöördterminaatorit, mille abil peatatakse DNA süntees 3'OH otsast. Iga modifitseeritud nukleotiid on märgistatud erineva fluorestseeruva värviga. Paardumata nukleotiidid pestakse välja. Kui algselt saavutati Illumina tehnoloogiaga lugemite pikkuseks kuni 36 aluspaari, siis nüüdseks on võimalikus saanud ka pikemad 100 aluspaari pikkused (Zhang jt., 2011). (Shendure ja Ji., 2008)

1.1.2.3 SOLid System

Applied Biosystem täiustas algselt George Churchi laboris väljatöötatud meetodi ja tõi ligeerimisel põhineva sekvenaatori turule aastal 2007 (Voelkerding jt., 2009). Esialgse süsteemi 2.1 platvormist on jõutud välja arendada SOLiD 4 analüsaator, mis saavutab

lugemite pikkuseks kuni 50 ap ja suudab genereerida väljundandmeid jooksu kohta 80-100 Gbp (Zhang jt., 2011).

See ligeerimisel (*sequencing-by-ligation*) põhinev sekveneerimismeetod kasutab DNA paljundamise jaoks analoogselt 454 pürosekveneerimisega PCRi emulsiooni (joonis 2a). Kasutatakse oligonukleotiidseid adaptereid, millele on ligeeritud DNA fragmendid ja amplifikatsioon leiab aset ühemikromeetriliste kerakeste peal. Seejärel kinnitatakse kerakesed sekvenaatoris asuvale spetsiifiliselt töödeldud läbivoolutavale klaaspinnale, millel DNA järjestamine aset leiab. (Voelkerding jt., 2009)

Sekveneerimise alustamiseks lisatakse universaalne sekveneerimise praimer, mis on komplementaarne SOLiDi raamatukogu fragmentide spetsiifiliste adapteritega. Samuti vajatakse DNA ligaasi ja piiratud hulgal poolenisti degradeeritud oligonukleotiididest koosnevaid lühikesi järjestusi. Huvipakkuvad proovid koosnevad kaheksast oligonukleotiidist – oktameeridest. Oktameeri kaks esimest alust on proovispetsiifilised ja nende moodustamiseks võimalusi on 16 (AA, TT, AT jne). Ülejäänud kuus aluspaari, mille 5' otsas on ka üks neljast fluorestsentsmarkerist, on juhuslikud. Juhul kui oktameer vastab komplementaarsusprintsibi alusel DNA fragmendile, leiab aset hübridisatsioon universaalse praimeri 5' fosfaatrühmale. Pärast ligeerimist pestakse seondumata nukleotiidid välja. Fluorestsentssignaal dokumenteeritakse enne oktameeri viimase kolme nukleotiidi eemaldamist. Kolmanda nukleotiidi 5' ots fosforüleeritakse ning lisatakse uus uuritav proov. Praimeri laiendamine leiab aset seitsme ligeerimistsükli, mis moodustab ühe seeria. Hilisemalt praimeri produkt sihtmärk-DNA-lt (*template-DNA*) denatureeritakse ja teine sekveneerimisseeria võib alata n-1 praimeriga. Kokku viiakse läbi viis seeriat viie erineva (*off-set*) praimeriga. (Voelkerding jt., 2009)

Kirjeldatud viisil suudab sekvenaator kuue päevaga genereerida 35 ap lugemeid. Lisades sekvenaatorisse kaks klaaspinda, suudetakse ühe jooksutamisega saavutada 4 Gbp mahus andmeid (Voelkerding jt., 2009).

SOLiD sekveneerimine võimaldab läbi viia kahealuselist kodeerimist st. neljas ja viies alus on tähistatud spetsiifilise fluorestsentsmärgisega (Mardis, 2008; Zhang jt., 2011). See lisakontroll tõstab lugemite täpsust. Oktameeride järjestus sisaldab teadaolevaid fikseeritud nukleotiide, mille alusel saab tuvastada nukleotiidide ebakõlasid edasisel andmeanalüüsil. SOLiD4 platvorm pakub küll väga head andmekvaliteeti, kuid sekveneerimisele eelnev etapp ehk DNA raamatukogu valmistamine võib olla tülikas ja liigselt aeganõudev (Zhang jt., 2011).

1.2 Genoomide kokkupanemine

Pärast geenide sekveneerimist on järgmiseks etapiks lugemitest genoomide kokkupanemine. Kui geenide sekveneerimine andis informatsiooni nukleotiidses järjestuses kohta, siis arvutiprogrammid proovivad rekonstrueerida erinevaid pikemaid genoomseid järjestusi kasutades selleks sekveneeritud lugemite joendamist mitmesuguste algoritmide alusel. Sekveneerimistehnoloogiatel on olnud oluline mõju (tabel 2) genoomide kokkupanemise programmide arengule (Pop, 2009).

Tabel 2. Kokkuvõtte teise põlvkonna sekveneerimisandmete iseärasustest ja nende mõjust genoomi kokkupanemise programmidele.⁴

Sekveneerimistehnoloogia	Mõjud genoomi kokkupanemise programmidele
vead	
Lühikesed lugemid	Muudab keeruliseks kordusjärjestusalade kokkupanemise
Puuduvad paarislugemid	Paarislugemite puudumine muudab kordusjärjestusalade kokkupanemise keeruliseks
Uued veatüübid ⁵	Vajadus täiendada olemasolevaid programme ja/või lisada veaspetsiifilisi algoritme lahendamiseks
Suur andmehulk (lugemite arv ja lisainformatsiooni suurus)	Efektiiivsusküsimused vajavad lahendust paralleelsete rakenduste abil või spetsiaalset suurtele genoomidele kohandatud tarkvarade arendamist

Genoome saab kokku panna kahel erineval viisil. Vastavalt sellele, kas on olemas referentsjärjestused jagatakse genoomid referentsi alusel koostatuks või *de novo*-ks („uus“). *De novo* lähenemisviisi juures on tegemist organismi genoomi kokkupanemisega, kelle enda ja lähisugulaste genoomi ei ole sekveneeritud. Teine lähenemisviis põhineb võrdlusel, mille puhul kasutatakse lähisugulase sekveneeritud genoomi kokkupanemisprotsessi alusena. Sellise meetodiga on lihtsam genoomi konstrueerida, kuna uue organismi nukleotiidses järjestuses kokkupanekuks on vajalik vaid piisaval arvul lugemeid joondada referentsgenoomile. *De novo* kokku pandud genoomi koostamine on ajaliselt ja rahaliselt kulukam, sest eelduseks on suur paarislugemitega raamatukogu olemasolu (Xue jt., 2013). Lisaks on oluline sügavam sekveneerimine, mis tasakaalustaks teise põlvkonna tehnoloogiaga

⁴ Pop, 2009

⁵ Näiteks 454 tehnoloogia, millega DNA järjestuste määramine põhineb luminesentsil, ei suuda edukalt hinnata homopolümeerseid piirkondi pikkesti ja sellest tulenevalt tekitab sekveneerimisvigu.

loodud lugemite puudused lühiduses ja vearohkuses (Desai jt., 2013). Võimalik on ka mõlema meetodi süntees: piirkondade puhul, mis erinevad tunduvalt referentsgenoomi omast, saab kasutada *de novo* lähenemist (Pop, 2009).

Genoomi kokkupanemise programmid põhinevad intuiitiivselt selgel eeldusel: kui kahel lugemil esineb nukleotiidiline ühisosa, siis pärinevad nad tõenäoliselt genoomi samast kromosoomi piirkonnast. Kui selline kattuvus on tuvastatud, siis joondab programm lugemid vastavalt kontiigideks, mille moodustab komplekt ülekattuvaid lugemeid (Narzisi ja Mishra, 2011).

Genoomid peavad olema täpselt ja terviklikult kokku pandud, kuna sellel baseerub edasine töö: funktsionaalsete elementide ennustamine või järjestuse evolutsioonilise päritolu välja selgitamine (Meader jt., 2010). Siiski puudub kvaliteedi hindamiseks üldtunnustatud ja standardiseeritud meetod. Sobivate hindamiskriteeriumite leidmise teeb raskemaks sekveneerimistehnoloogiate mitmekesistumine (Meader jt., 2010). Laialdasemalt kasutusel olevad meetrikud hindavad kokkupanud genoome kontiigide suuruse, mitte kontiigide kvaliteedi ja täpsuse alusel (Narzisi ja Mishra, 2011; Vezzi jt., 2012 b).

Esimese ja teise põlvkonna sekveneerimistehnoloogiate loodud lugemid erinevad nii pikkuselt kui ka vigade omadustelt (Meader jt., 2010). Sagedasti raskendavad genoomide kokkupanemist uue põlvkonna lühikesed ja suure arvukusega lugemid ja nende erinevad veaprofiilid (Dohm jt., 2007; Pop, 2009; Zhang jt., 2011). Seega on kokkupanud genoomide genereerimisalgoritmid nendel põhjustel keerukamad ja isegi kasutades väga võimsaid arvuteid, tekib vastavatel tarkvararakendustel probleeme suure andmemassiivi haldamisega (Dohm jt., 2007).

Sekveneerimistehnoloogiate arenguga proovivad kaasas käia genoomi kokkupanemise programmid. Pärast seda, kui Sangeri tehnoloogia ei olnud enam ainus meetod DNA järjestamiseks, on välja arendatud erinevaid lähenemisviise (tabel 3). Need püüavad, hoolimata uute tehnoloogiatega toodetud lugemite spetsiifilistest probleemidest, leida lahendusi DNA esialgse järjestuse leidmise probleemile. Programmiga Phusion (Mullikin ja Ning, 2003) on kokku pandud suur ja korduselementide rikas hiire genoom. Lisaks hiirele, on koostatud sama programmiga ka nematoodi *Caenorhabditis briggsae* genoomne järjestus (Mullikin ja Ning, 2003). Phusion konstrueerib lugemitest graafi (OLC), kus sõlm vastab lugemi järjestusele ja sõlmi ühendav kaar tuvastatud lugemite vahelisele katvusele (joonis 3). Selline lähenemisviis ei oleks rakendatav teise põlvkonna tehnoloogiaga toodetud lugemitele, kuna arvutuslikult muutuks graaf liiga suureks. Uuemad meetodid nagu ALLPATHS (Butler jt., 2008), SSAKE (Warren jt., 2007), ABySS (Simpson jt., 2009), SGA (Simpson ja Durbin,

2011) ja 454 lugemitele spetsialiseerunud Newbler (Roche 454) on vaid üksikud näited programmidest, mis sisendina kasutavad lühikesi (≤ 100 bp) lugemeid.

Tabel 3. Nimekiri levinud genoomi kokkupanemise programmidest

Genoomi kokkupanemise programm	Lugemitüüp	Toetatavad tehnoloogiad	Autor
Phusion	pikemad lugemid	Sanger	Mullikin ja Ning, 2003
SSAKE	lühikesed lugemid	Illumina/Solexa	Warren jt., 2007
SHARCGS	lühikesed lugemid	Illumina/Solexa	Dohm jt., 2007
Velvet	lühikesed lugemid	Illumina/Solexa, 454	Zerbino ja Birney, 2008
ALLPATHS	lühikesed lugemid	Illumina/Solexa, SOLid System	Butler jt., 2008
ABYSS	lühikesed lugemid	Illumina/Solexa, SOLid System	Simpson jt., 2009
Newbler	lühikesed lugemid	454	http://454.com/products/analysis-software/
SGA	lühikesed lugemid	Illumina/Solexa	Simpson ja Durbin, 2011
SOAPdenovo2	lühikesed lugemid	Illumina/Solexa	Luo jt., 2012

1.3 Genoomide kokkupanemise programmid

1.3.1 SSAKE

SSAKE on genoomi kokkupanemiseks loodud tarkvararakendus, mis prefikspuu abil otsib sekveneerimisandmete seast kahe lugemi vahelist pikimat ülekattvust. Lühikeste järjestustega töötamiseks arendatud programm on sobilik kuni 10 kb suurusega genoomide kokkupanekuks (nt. Phi X174) ja miljonitest identsetest lühijärjestustest klastrite loomiseks metagenoomsete uuringute vallast. SSAKEga on suudetud ühe üksiku kontiigina genome kokku panna Phi 174 ja SARSiga assotsieeritud koronaviiruse jaoks. Viirustega võrreldes komplitseeritumatest eluvormidest on moodustatud unikaalseid kontiige nii *H. influenzae* genoomi puhul kui ka Sargasso mere metagenoomi projekti raames. (Warren jt., 2007)

SSAKE töös talletatakse sekveneerimisandmed paisktabeli andmestruktuurina, mis räsifunktsiooni abil viib vastavusse võtme (unikaalsed järjestused) ja väärtuste (järjestuste esinemissagedused andmestikus) paarid. Prefikspuu organiseerib korrastatud tabelis lugemeid

k-mer'ide (kindlaksmääratud pikkusega alamjärjestused lugemist) alusel ja laiendab neid sobival juhul 3' otsast. Lugemid reastatakse esinemissageduse alusel vähenevas järjekorras, mis peegeldab katvust ja väldib vigu sisaldavate järjestuste kasutamist laienduseks. Esialgselt lugemist genereeritakse ja kasutatakse prefikspuu otsingul kõige pikem 3' lugem, kui see ei ületa kasutaja poolt defineeritud minimaalset sõne (tähestiku sümbolite järjend, antud juhul lugem) pikkust või leiab aset teise lugemi 5' otsaga ideaalne komplementaarne paardumine. Säärase toimingu korral laiendatakse esialgne lugem paardumata 3' otsast kontiigiks ja teine lugem, millega leidis aset paardumine, jäetakse tabelist ja puust välja. Laiendusprotsess toimub tsüklikena lühemate 3' *k-mer*'ide kasutamise suunas. DNA prefikspuud kasutatakse otsinguruumide vähendamise kaudu võimalike lugemi jadade tõhusamaks ja suunatuks leidmiseks. SSAKE programmil on võimalik kontrollida laiendusprotsessi. Ekstensioon lõpetatakse juba ühe sobiva lugemi paardumise korral või kui *k-mer* on väiksem kasutaja poolt kindlaksmääratud minimaalsest lävendväärtusest. Esimese tulemuseks on küll lühemad kontiigid, kuid see vähendab valekokkupaneku tõenäosust. (Warren jt., 2007)

Teiste programmidega võrdlused on näidanud SSAKE puudust, milleks on tundlikkus vigade esinemisel sisendandmetes. Isegi minimaalsel lävendväärtusel põhineva filtersüsteemi kasutamine ei välista genoomi väära kokkupanekut. Nendel põhjustel ei ole see kõige sobilikum rakendus *de novo* genoomide konstrueerimisel. (Dohm jt., 2007)

1.3.2 SHARCGS

SHARCGS ehk *SH*ort-read *A*ssembler based on *R*obust *C*ontig extension for *G*enome *S*equencing on lühikestele lugemitele spetsialiseerunud genoomi kokkupanemise programm, mis põhineb robustsel kontiigi laiendamisel. SHARCGSi algoritmi tööpõhimõtte saab liigendada kolmeks etapiks: 1) filtreerimine, mis kindlustab vaid selliste lugemite kasutamist, mis täidavad kahte tingimust: piisav esinemissagedus ja küllaldasel määral ülekattuvate lugemite olemasolu; 2) kontiigide moodustamine ja 3) kontiigide kvaliteedi hindamine. Tarkvararakenduse algoritmi tuumikut jooksutatakse automaatselt kasutades kolme erinevat filtreerimisparameetrit (nõrk, keskmine, tugev). SHARCGS võimaldab kiiret ja kõrge täpsusega *de novo* genoomide koostamist. Konstrueeritud kahe pärmi kromosoomi ja kahe bakteriaalse genoomi analüüsil selgus, et vaid ühte 949,974-st kontiigist pikkusega üle 50 bp ei suudetud õigesti joondada referentsjärjestusele. Ülejäänute puhul ei tekkinud probleeme

võimalike tühimikega (*gaps*) või mittevastavustega. Ka oli kokkupanud genoomide järjestuste katvus referentsjärjestusele kõrge (ületas 97%). (Dohm jt., 2007)

Kokkupanud genoomides on üheks probleemiks piirkonnad, kus lugemite katvus on väga väike, mis vähendab võimalusi nii kontiigide pikendamiseks kui ka kordusi sisaldavate alade korrektset konstrueerimist. SHARCGSi lähenemisviis (ja samal ajal teiseks filtreerimisetapiks) sellele probleemile on järgnev: iga lugemi puhul, mida kasutatakse kontiigiks konstrueerimisel, uurib programm konkreetse lugemi mõlemat ahelat mitteunikaalsete piirkondade tuvastamiseks. Lugemid kasutatakse kontiigi laiendamiseks vaid sellisel juhul, kui teised lugemid sobivad kontiigiga lühikeste ülekattuvuste kaudu nii, et ei tekita sellega mitteunikaalsust – minimaalse kattuvuse parameeter. Sobivaks loetakse lugemid, kui tema mõlemale otsale leidub vähemalt üks sobiv partner ehk esineks vähemalt kahekordne kattuvus. Esimesel filtreerimisetapil eristatakse lugemid kõrgeima minimaalse kvaliteediväärtuse alusel. SHARCGSi sellised lähenemisviisid aitavad selekteerida sisendandmetest vigased lugemid õigest. Optimaalseks peetaksegi kolmekordset filtreerimist, kuna kahekordne ei ole vigadest vabanemiseks piisavalt efektiivne ja samas neljakordse filtri tulemusena on kokkupanemiseks olemasolevaid lugemeid liiga vähe alles. Leebete filtreerimiskriteeriumite korral on takistatud pikemate kontiigide moodustamine, sest lugemid sisaldavad sekveneerimisvigu, mis omakorda põhjustavad vääraid mitteunikaalsusi. Sama tulemuse võib anda ka liiga range filtri rakendamine: kontiigid muutuvad liiga lühikeseks, kuna ühenduseks sobilikud külgnevad lugemid puuduvad. (Dohm jt., 2007)

Kokkupanud genoomi kvaliteedi hindamiseks kasutatakse indikaatorina N50 suurust, mida määratletakse kui kontiigi vähimat pikkust, millest võrdsed või pikemad kontiigid katavad 50% kogu genoomsest järjestusest. SHARCGS programmiga koostatud *Arabidopsis* ja *Drosophila* BAC järjestuste N50 väärtuseks saadi >20 kbp. (Dohm jt., 2007)

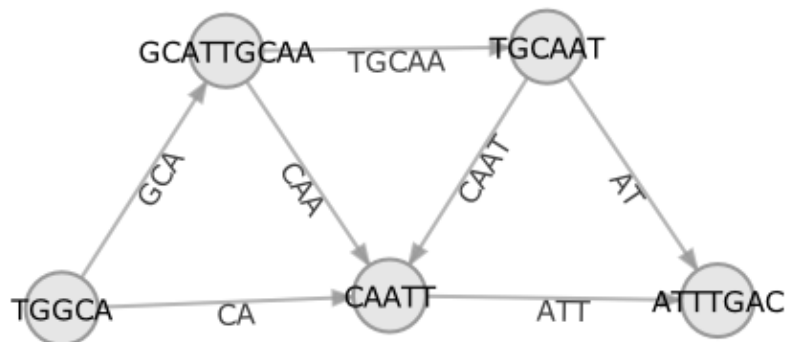
Teine kuni 25 aluseliste lugemite spetsialiseerunud programm SSAKE ei ole nii efektiivne võrreldes SHARCGSiga – tulemuseks on vigaselt kokkupanud genoomid, mida põhjustab kontiigide pikendamine kordusjärjestuste piiridel. Realistliku simulatsiooni korral, mis hõlmas nii puuduvaid lugemeid kui sekveneerimisvigu, ei suudetud 25% saavutatud kontiigidest referentsjärjestusele joondada. Samas joondati kõik SHARCGS programmi poolt kokkupanud kontiigid referentsile. Kui SSAKE saavutas ühe BACi katvuseks 75%, siis SHARCGSil küündis tulemus 93%-ni. Seega on SSAKE palju vastuvõtlikum vigade suhtes, mis esinevad sisendandmetes võrreldes SHARCGSiga. Kuigi SSAKE-l on samuti filtersüsteem, mis peaks selliseid valeühendusi vältima, ei taga see piisavat efektiivsust. See

asjaolu piirab SSAKE rakendatavust lühikeste lugemitega *de novo* sekveneerimisprojektide tarbeks. (Dohm jt., 2007)

1.3.3 Velvet

Velvet on sobivalt kohandatud *de Bruijn* graafil põhinev genoomi kokkupanemise programm, mis on arendatud spetsiaalselt lühikeste lugemite jaoks. Velvetil on võime lahendada kahte ülesannet eraldi: see suudab nii vigu kõrvaldada, kui leida lahenduse korduste jaoks. Velveti tööprotsessi võib jagada nelja etappi: 1) lugemite lõikumine *k-meri*-deks; 2) graafi moodustamine; 3) vigade korrigeerimine; 4) korduste lahendamine paarislugemite kasutamisel. (Zerbino ja Birney, 2008)

Velveti lahendusviis vastandub traditsioonilisele OLC ülekatvuse leidmise meetodile (joonis 3). Varasem lähenemisviis on sobilikum Sangeri ensümaatilise terminaatori tehnoloogiaga toodetud pikemate lugemite jaoks. Seda ei saa väita lühemate lugemite kohta. Lühikesed lugemid muudab atraktiivseks võrreldes pikematega nende sünteesimise odavus, kuid nende edasiseks rakendamiseks on vaja suurt kvantitatiivset kogust. Illumina tehnoloogiaga toodetud miljon lugemit (ehk graafi jaoks miljon sõlme) muudab OLC graafi liiga suureks ja pikaks, et tänapäeva arvutitega neid lahendada saaks. (Compeau jt., 2011)

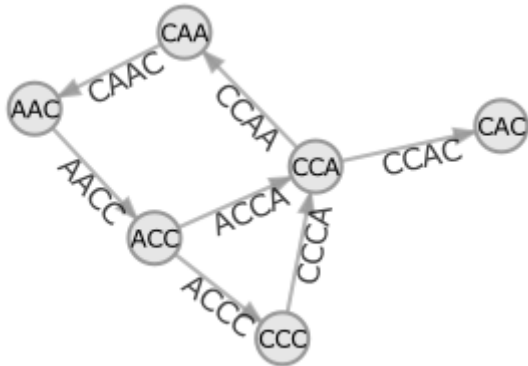


Joonis 3. OLC graaf.⁶ Sõlmedele vastavad terviklikud lugemid ja neid siduvad ühendused lugemite vahelisele ülekatvusele.

De Bruijn graafi moodustavad defineeritud pikkusega *k-mer*'id (joonis 4). Kindlaksmääratud pikkuse juures väljendab *k* tasakaalu tundlikkuse ja spetsiifika vahel, mis on vajalik edasiseks katvuste tuvastamiseks, *k-mer*'ide ühendamiseks ja sõlmede konstrueerimiseks. Graafi

⁶ <http://gcat.davidson.edu/phast/olc.html>

konstrueerimiseks luuakse andmebaas, mis sisaldab informatsiooni, näiteks *k-mer*'i asukohta lugemis või millised *k-mer*'ide järjestused moodustavad sõlme.



Joonis 4. De Bruijn graaf.⁷ *K-mer*'i pikkuseks on antud joonisel neli ($k=4$). Sõlmedeks on kattuvad *k-mer*'ide osajärjestused. Graaf koosneb *k-mer*'idest, mis ühisosa ($k-1$) esinemise korral teisega moodustavad sõlme. Nt. *k-mer*'ide CAAC ja AACC ühisosaks (ja ühtlasi sõlmeks) on ACC.

Pärast graafi loomist on võimalik vigade korrektuur (tahtmatute ühenduste ehk väärade sõlmede kõrvaldamise läbi) või lihtsustamine. Silmas peab pidama, et vigade päritolu võib olla bioloogiline (polümorfismid) või hoopis metodoloogiline (tekkinud sekveneerimisprotsessi ajal). Velvet tuvastab vigu lähtuvalt graafi topoloogiast. (Zerbino ja Birney, 2008)

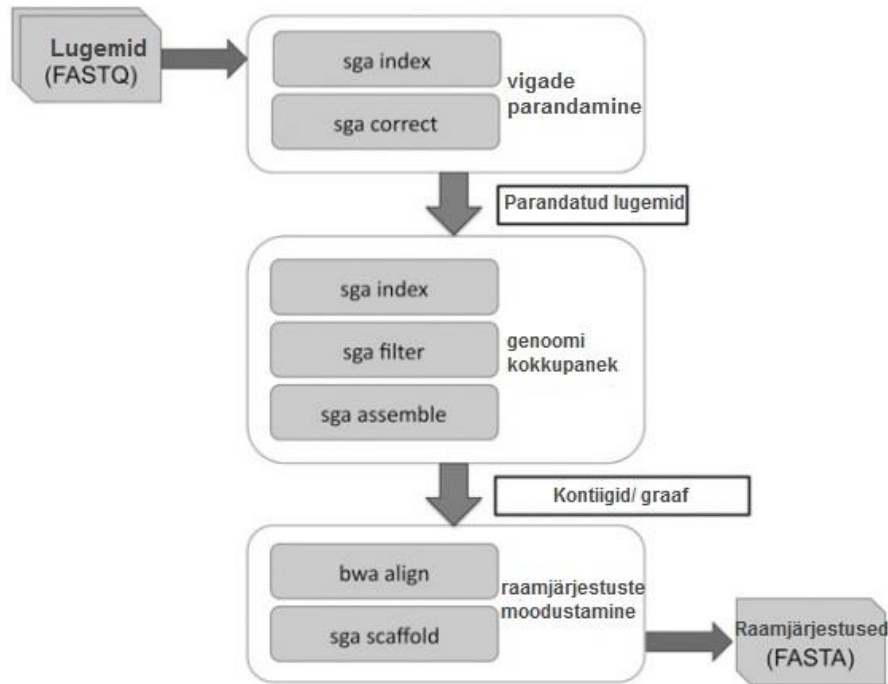
Kõige ressursinõudlikum protsess Velveti tegutsemis skeemi juures on graafi konstrueerimine. Võrreldes SSAKE-ga on Velvet mälunõudlikum programm, kuid suudab moodustada veavabalt pikemaid kontiige lühema aja jooksul. (Zerbino ja Birney, 2008)

1.3.4 SGA

SGA (*String Graph Assembler*), erinevalt mitmetest teistest *de Bruijn* graafi kasutatavatest programmidest, rakendab arvutimälu säästlikumat meetodit genoomi kokkupanemiseks (joonis 5). *De Bruijn* graafi puuduseks on lugemite jupitamise käigus kaduma läinud informatsiooni taastamine keeruliste algoritmide abil. SGA korral tekitab algoritm graafi ülekattuvate terviklike lugemite alusel. Alternatiivne lähenemisviis lubab efektiivsemat andmetöötlust sadadesse giga-aluspaaridesse ulatuvate imetajate genoomide lugemite korral. Tõhustatud algoritm kasutab ära lugemite liiasuse esinemist (ühe positsiooni kohta

⁷ <http://gcat.davidson.edu/phast/debruijn.html>

mitmekordne sama järjestusega lugemi leidumine) ja seetõttu saab andmestruktuuride kokkusurumisega genoomi kokku panna väiksema mälu kasutusega. (Simpson ja Durbin, 2011)



Joonis 5. SGA tarkvararakenduse tööjärjekorra skemaatiline esitus.

Genoomi kokkupanemise programmi käsuahel koosneb kolmest etapist: vigade parandamine, kontiigide moodustamine ja raamjärjestamine. Veaparandusetapis varustatakse lugemid FM-indeksiga (*sga index*), mis võimaldab teostada tõhustatud otsingut kokkusurutud andmestikus. Seejärel viiakse läbi defektsete lugemitega kahest meetodist koosnev korrigeerimisprotseduur, mille käigus loetakse lugem usaldusväärseks, kui igale konkreetsele alusepositsioonile vastab piisav arv *k-mer*'e. Teine meetod põhineb lugemite ebatäpsel ülekattuvusel. Järgmine etapp võtab sisendiks eelnevalt parandatud lugemid, taasindekseerib need ja heidab kõrvale duplikaatsed ja madala kvaliteediga lugemid. Protsessi lõpuks ehitatakse graaf, mille alusel genereeritakse kontiigid. Viimaseks etapiks on raamjärjestuste koostamine kontiigide ja paarislugemite põhjal. (Simpson ja Durbin, 2011)

Assemblathon 1 projekti tulemuste järgi saavutas SGA parima raamjärjestuste N50 väärtuse ja tegi ühtlasi kõige vähem asendusvigu (Earl jt., 2011). Võrdlusel kolme programmiga (AbySS, SOAPdenovo ja Velvet) edestas SGA neid programme täpsuses (madalaim valepaardumiste

määr) ja kokkupanud genoomi terviklikkuses. Velvet saavutas parima kontiigide N50 väärtuse (18.4 kbp), kuigi tulemustevahe teisel kohal asuva SGA-ga ei ole märkimisväärselt suur (16.8 kbp). (Simpson ja Durbin, 2011)

1.3.5 Soapdenovo2

Soapdenovo2 on oma eelkäijaga (Soapdenovo1) võrreldes täiustatud versioon moodulite paketist, mis on loodud eesmärgiga lugemitest *de novo* genoome kokku panna. Algupärast versiooni on kasutatud edukalt mitmete avaldatud genoomide kokkupanemiseks, kuid esineb olulisi parandamist vajavaid puudusi nii kvaliteedis kui ka kvantitatiivsetes näitajates. (Luo jt., 2012)

Soapdenovo2 programm tervikuna koosneb kuuest erineva ülesandega moodulist. Esimeseks on sekveneerimisandmete vigade paranduse etapp, mis põhineb lühemate *k-mer*'ide kasutusel. Sellele järgneb *k-mer*'idel põhineva lihtsustatuma *de Bruijn* graafi (joonis 4) koostamine: lugemid lõigatakse lühemateks järjestusteks ehk *k-mer*'ideks ja grupeeritakse sobivateks rühmadeks. Pärast esmase graafi koostamist genereeritakse uus, kuid seekord kaardistades pikemad *k-mer*'id esialgsele, et oleks võimalik kordusjärjestustega alade lahendamine. Soapdenovo2 kasutab seega efektiivselt *k-mer*'ide pikkusest tulenevaid omadusi erinevate ülesannete lahendamiseks. Pikemaid *k-mer*'e kasutatakse kordusjärjestusi sisaldavate piirkondadega töötamiseks, lühemaid madala katvusega ja sekveneerimisvigu sisaldavate regioonide juures. (Luo jt., 2012)

Raamjärjestuste konstrueerimise põhiraskusteks on heterosügootsete kontiigipaaride, kimäärsete järjestuste ja kontiigide vaheliste tühimike esinemine. Õigel positsioonil asuv kontiig eristatakse väärtast suurima katvussügavuse alusel välja. Kimäärsed raamjärjestused, mis on ekslikult üles ehitatud väiksemate kloonide alusel, heidetakse edasisest analüüsis kõrvale kasutades suure insertiga raamatukogusid. Programm suudab topoloogial põhineva meetodi abiga valida edasise töötamise jaoks kontiige, millel on piisavalt paarisotsaliste lugemite (PL) informatsiooni. Tühimike sulgemise tõhustamiseks kasutab Soapdenovo2 meetodit, mis hõlmab ka eelnevates tsüklites joondatud lugemite kasutamist. (Luo jt., 2012)

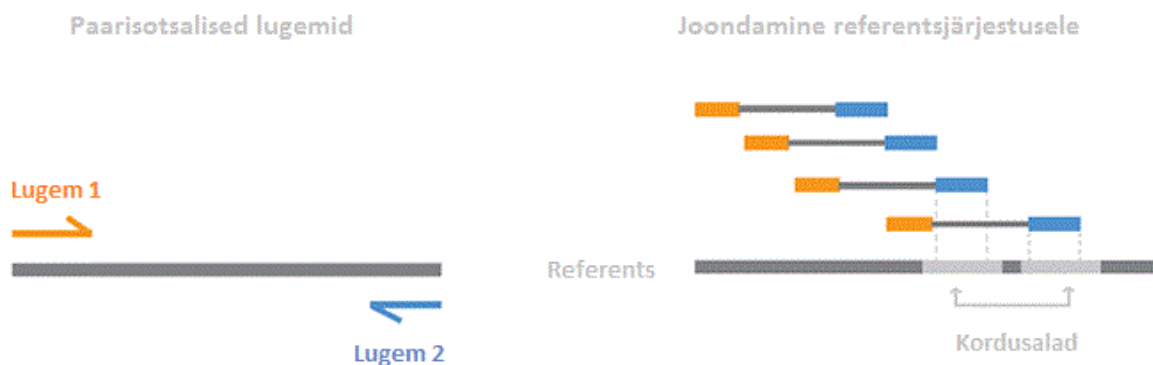
Kokkuvõtteks sisaldab Soapdenovo2 järgmisi muudatusi: 1) võimendatud vigade parandusalgoritm; 2) lihtsustatum ja seetõttu vähem mälunõudlikum graafi konstrueerimine; 3) pikkade kordusjärjestuste lahendamine *k-mer*'idega; 4) täiustatum tühimike sulgemise meetod. Nende ümberkorralduste tõttu on uus programm oma eellasest efektiivsem ja

kasulikum bioinformaatika tööriist, mida tõendas ka kokkupandud uus asiadist inimese genoom (suurem kontiigide ja raamjärjestuse N50 väärtus), suurem genoomikatvusprotsent ja kolmveerandi võrra väiksem mälukasutus. (Luo jt., 2012)

1.4 Raamjärjestamine

Kokkupandud genoomid hõlmavad endas kollektsiooni pidevatest pikematest DNA järjestustest ehk kontiigidest, mille asukoht genoomis ei ole defineeritud (Vezi jt., 2012b; Pop jt., 2004). Kontiigi määratlemisel on oluline, et aluspaarid oleksid kõrge usaldusnivooga (kvaliteetsed). Samuti on teada lugemite orientatsioon ehk kummast ahelast on nad rekonstrueeritud.

Asukoha määratlemiseks viiakse kontiigidega läbi protseduur, mida nimetatakse raamjärjestamiseks. See protseduur hõlmab endas kontiigide järjestamist ja õige suuna andmist paarilugemitest pärinevat info abil. DNA fragmenteerimisel saadakse järjestused, mille otsad sekveneeritakse ja genereeritakse lugemid (joonis 6). Lugemite vaheline piirkond jääb tavaliselt sekveneerimata.



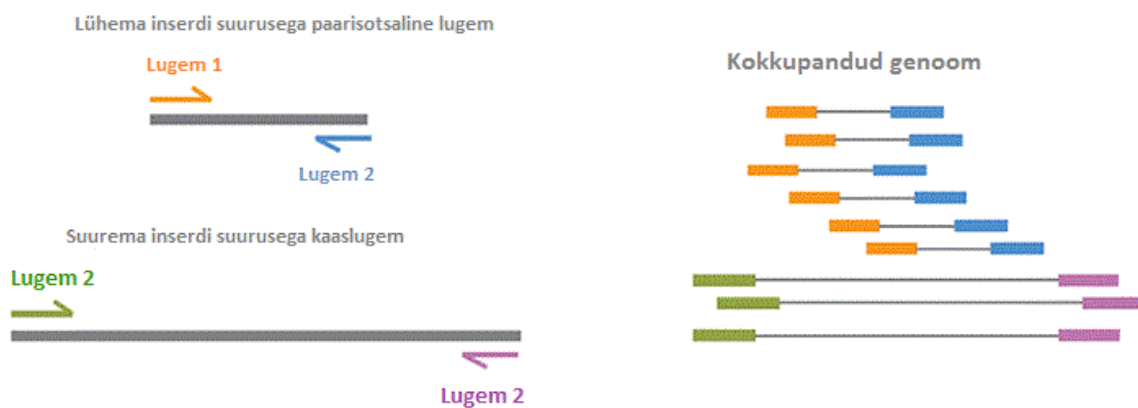
Joonis 6. Paarisotsaline sekveneerimistehnoloogia.⁸ Teades sekveneeritud otste vaheala pikkust on võimalik selliseid lugemeid kasutada raamjärjestamisel kontiigide liitmiseks ka probleemsete genoomsete alade (nt. kordusalade) puhul.

Raamjärjestus koosneb kontiigidest ja kontiigide vahele jäävatest tühimikest (*gaps*). Paralleelselt termini paarisotsaline lugem (PL) kõrval kasutatakse ka kaaslugem (mõlema ühendnimetusena paarilugemid), mis täidavad mõlemad sama eesmärgi (joonis 7) ehk annavad informatsiooni kahe lugemi vahelisest füüsilisest kaugusest (Gritsenko jt., 2012).

⁸ http://www.illumina.com/technology/paired_end_sequencing_assay.ilmm

Erinevus nende vahel seisneb raamatukogu tegemise metodoloogias. DNA raamatukogu moodustavadki fragmendid, mis on enamvähem ühepikkused ja on insertseeritud klonereimisvektorisse. Nii kontiigi moodustavate lugemite orientatsiooni kui ka fragmentide keskmise pikkuse teadmine võimaldab eeldada kahe kontiigi vahelise tühimiku pikkust. Taolise protsessi tulemuseks on koostatud raamjärjestused, mille abil luuakse täielikum genoomne järjestus.

Siiski esineb ka genoomi kokkupanemise programme, mis sisaldavad endas juba raamjärjestamise moodulit. Kuigi mõlemaid komponente hõlmav programm pakub kasutajamugavust, ei ole see alati kõige õigem viis genoomi järjestuse lõpetamiseks, sest universaalset, kõikidele genoomidele sobivat meetodit ei ole veel leitud. Sellise programmi rakendamisel on kasutajatel vähe kontrolli raamjärjestamise protsessi üle ja informatsiooni analüüsimise suunamist pole võimalik teostada. Arendatud on ka selliseid programme, mis on paindlikumad ja võimaldavad kasutajatel detailsemalt parameetreid muuta (Pop jt., 2004; Salmela jt., 2011). Järgnevas peatükis kirjeldatakse neid lähemalt.



Joonis 7. Paarisotsaliste lugemite ja kaaslugemite kasutamine kokkupandud genoomi koostamiseks.⁹ Mõlema lugemitüüpe pikkuse erinevuste iseärasuste kasutamine tõhustab *de novo* genoomi kokkupanemist (Boetzer jt., 2011). Pikema inserdiga kaaslugemeid kasutatakse põhiliselt kotiigidest raamjärjestuste kokkupanemiseks ja selle abil on võimalik liita keerukamaid kordusjärjestusi sisaldavaid piirkondi pikemateks kontiigideks suuremate vahemaade tagant. Väiksema inserdi pikkusega paarisotsaliste lugemite kasutamine täiendab kaaslugemeid võimaldades tühimike täitmist.

⁹ http://www.illumina.com/technology/mate_pair_sequencing_assay.ilmn

1.5 Raamjärjestamise programmid

1.5.1 Bambus

Bambus on esimese eraldiseisva meetodi, Grouperi, edasiarendus (Fuchs, 1997). See on vabavaraline programm, mis toetab enamike genoomi kokkupanemise programmide väljundandmeid ja tekitab andmete vahelisest seosest graafi. Samuti võimaldab tarkvara parameetrite paindlikku rakendust vastavalt kasutaja soovidele. Lisaks tüüpilistele meetoditele, mis toetuvad eelkõige paarislugemite informatsioonile, võimaldab Bambus kasutada toetavaid lisaandmestikke (Gritsenko jt., 2012). Nii saab kasutada lisaks paarislugemitele ka järjestuste joendusprogrammi Mummeri (Kurtz jt., 2004) andmeid lõpetatud genoomide järjestuste kohta, et neid võtta aluseks lähisugulaste genoomi kokkupanemisel. (Pop jt., 2004)

Bambuse algoritm kasutab heuristilist lähenemisviisi (*greedy*), millega proovib jõuda eesmärgini võimalikult minimaalse arvutusliku hinnaga. Eelnimetatud lähenemisviis tähendab, et raamjärjestamist alustatakse kõige kindlamate seostega kontiigidest ja ülejäänuid kasutatakse, kui nad ei tekita ebakõla olemasolevate kontiigide suhtes. Programm võimaldab parameetreid seadistada konfiguratsioonifailiga, millega saab kasutaja muuta eelistusi, nagu näiteks millistest raamatukogudest esmalt paarislugemeid kasutada soovitakse või defineerida minimaalsete linkide arv, mida on vaja kahe kontiigi ühendamiseks (Pop, 2009). (Pop jt., 2004)

Bambuse tulemiks võivad olla nii unikaalsete kui ka mitteunikaalsete asetustega kontiige sisaldavad raamjärjestused. Alternatiivsed raamjärjestused on kasulikud nii haplotüüpsete piirkondade indikaatoritena kui ka genoomide kokkupanemise lõpetamisele kaasa aitajatena. Kui on vaja unikaalseid raamjärjestusi, siis suudab tarkvara spetsiaalse mooduli abiga need teistest eraldada. Lisaks dokumenteerib tarkvara algoritm kasutatud ühendusi nelja kategooriasse: kehtiv (ühendus on raamjärjestuses kasutus leidnud), kehtetu orientatsioon (väär suundumuse tõttu), kehtetu pikkus (pikkus ei vasta piirangule) ja kasutamata. (Pop jt., 2004)

Bambust on võrreldud Celera Assembleri (CA) (Myers jt., 2000) ja Arachnega (Batzoglou jt., 2002), mis mõlemad on genoomi kokkupanemise programmid ja sisaldavad raamjärjestamise moodulit. Bambus oli CA-st ja Arachnest efektiivsem, kuna saavutas üle poolte juhtude suuremad kontiigid. Raamjärjestuse kvantitatiivse väärtuse tõstmine toimub kvaliteedi arvelt ja Bambus toodabki keskmiselt rohkem vigaseid raamjärjestusi võrreldes kahe teise

programmiga. Vigade suur osakaal tuleneb Bambuse erinevast rõhuasetusest olles eelkõige abivahend genoomi viimistlemiseks ja on seetõttu tundlik väärtalt konstrueeritud kordusjärjestuse suhtes. (Pop jt., 2004)

Bambus on leidnud edukat kasutust võrdleva genoomika projektides nagu *Bacillus anthracis*'e tüvede või *Drosophila* liikide sekveneerimine. Bambuse tugev külg ehk polümorfsete regioonide ülesleidmine on aidanud odasaba (*Limulus polyphemus*) genoomi raamjärjestuste konstrueerimisel ja haplotüüpide identifitseerimisel (Nossa jt., 2013).

Võrreldes teise eraldiseisva raamjärjestamise programmi SSPACE-ga (Boetzer jt., 2011) (kasutab ka *greedy* lähenemist), on Bambusel palju rohkem funktsioone. SSPACE seevastu toetab uuemate sekveneerimistehnoloogiate lugemeid, kuid Bambus on disainitud esimese põlvkonna järjestuste analüüsiks (Salmela jt., 2011). Praeguseks on Bambuse autorid publitseerinud programmi uue versiooni (Bambus 2), mis toetab järgmise põlvkonna sekveneerimise lugemeid ja oskab lisaks analüüsida metagenoomika andmeid (Koren jt., 2011).

1.5.2 SSPACE

SSPACE (*SSAKE-based Scaffolding of Pre-Assembled Contigs after Extension*) on eraldiseisev programm, mis kasutab eelnevalt kokkupandud kontiige raamjärjestuste moodustamiseks. SSPACE on täiendatud versioon SSAKE genoomi kokkupanemise programmist. Selle uueks omaduseks on kontiigide laiendamise võimaldamine enne raamjärjestamist. Kasutajale antakse võimalus valida, milliseks otstarbeks ta soovib paarislugemeid kasutada, kas otse edasiseks töötamiseks või hoopis kontiigide ekstensiooniks. Nagu paljude teiste raamjärjestamise programmide puhul kasutab ka SSPACE paarislugemite informatsiooni. Lisaks on programmi eeliseks välja toodud selle kiirus ja vähene arvuti ressurssinõudlus. (Boetzer jt., 2011)

Esmalt filtreeritakse paarislugemeid ACTG (*Automatic Correspondence of Tags and Genes*) alusel ja need, mis sisaldavad ACTG kaardistatakse joendusprogrammiga Bowtie (Langmead jt., 2009) eelnevalt konstrueeritud kontiigidele. Nii leitakse igale kontiigile asukoht ja orientatsioon. Neid lugemiandmeid, mida ei suudetud kaardistada, kasutatakse kontiigide laiendamiseks. Nii proovitakse maksimaalselt kasutada lugemites sisalduvat informatsiooni. (Boetzer jt., 2011)

Raamjärjestamise protokoll on muudetud SSAKE programmis sellisel viisil, et see võimaldab kasutajal suuremal määral rakenduse tegutsemist kontrollida. Kontiigide ühendamise eelduseks on algoritmi poolt arvutatud ja kasutaja poolt määratud distantsi kooskõla. Samuti on vajalik teatud minimaalne ühendavate lugemite arv. Raamjärjestamist alustatakse suurimast kontiigist, mis toimub hierarhilisel viisil ehk esimesena läheb kasutusse väiksema inserdi suurusega raamatukogu. Lisaks suudab SSPACE hinnata kontiigide alternatiivseid ühendusi. Kõiki alternatiive proovitakse hinnanguliselt õigesse järjekorda paigutada. Kui see ei õnnestu, siis leitakse arvutuslikult kahest alternatiivist parim, mis ületab lävendi. Raamjärjestusi ei moodustata kahel järgneval juhul: kui kontiigil pole ühendusi teistega või alternatiivsete ühenduste määr ületab olulisel määral lävendit. (Boetzer jt., 2011)

Läbiviidud võrdluses kolme programmiga (GRASS, MIP ja OPERA) suutis vaid SSPACE kolme erineva andmestiku (*E.coli*, *P.suwonensis*, *P.syringae*) töötlusel saavutada nii väiksema arvu raamjärjestusi kui ka suure N50 väärtuse, kuid samas on läbivaks moodustatud raamjärjestuste ebatäpsus (Gritseko jt., 2012). Samuti oli SSPACE edukam N50 ja madalate raamjärjestuste arvus võrdluses ABySSiga (Boetzer jt., 2011). Seega saavutab SSPACE häid tulemusi, mis puudutab väikest raamjärjestuste arvu ja N50 väärtust, kuid see toimub vigade tegemise arvelt. Inimese genoomsete järjestuse puhul moodustas SSPACE võrreldes MIP Scaffolder'iga veidi pikemad, kuid siiski vähem täpsemad raamjärjestused (Salmela jt., 2011). Ühe andmestiku alusel ei suudetud võrdlust teostada st. SSPACE ei toeta SOLiD tehnoloogiaga loodud lugemeid (Salmela jt., 2011).

1.5.3 MIP Scaffolder

MIP Scaffolder on raamjärjestamise programm, mis sarnaselt teistele meetoditele kasutab sisendina kontiige ja paarislugemeid. Tänu nendevahelistele tekkinud ühendustele moodustub graaf, kus sõlmed vastavad kontiigidele ja ääred neid kontiige ühendavatele paarislugemitele. MIP Scaffolder kasutab edaspidises töös vaid selliseid paarislugemeid, mille mõlemad otsad on unikaalse asetusega kontiigide kollektsioonis. (Salmela jt., 2011)

MIP Scaffolder kasutab raamjärjestamise probleemi lahendamiseks graafi jagamist väiksemateks osadeks ja valib neist iga puhul eraldi MIP lähenemisviisi (*mixed integer programming*). Selliselt püütakse saavutada eesmärk, kus iga/ülekattuvad kontiig/id saaks raamjärjestuses oma asetuskoha ja suuna. MIP kasutab optimeerimislahendust, milles graafis asuvad kontiigide vahelised ühendused eemaldatakse nii, et kontiigi asetuse oleks

allesjäänutega kooskõlas. Järgneb raamjärjestuste kombineerimine, kui ühendatavad kontiigid on järjestustelt vähemalt 90% ulatuses sarnased. (Salmela jt., 2011)

MIP Scaffolderi tugevaks küljeks on kiirus. Võrdluses programmidega SSPACE ja SOPRA oli ta üle poolte juhtudest kiirem. Samuti suudab MIP Scaffolder genereerida pooltel juhtudel pikimaid raamjärjestusi, aga seejuures kannatab kattuvuse täpsus. Inimese genoomi andmete põhjal toimunud võrdluse alusel oli MIP võrreldes SSPACE-ga raamjärjestamisel täpsem saavutades parema genoomi ja raamjärjestuse kattuvuse. (Salmela jt., 2011)

1.5.4 GRASS

GRASS ehk *GeneRic ASsembly Scaffolder* on raamjärjestuste programm, mis sarnaselt Bambusele suudab oma tööks kasutada lisainformatsiooni erinevatest allikatest (eelnevalt lõpetatud genome lähisuguluses olevate organismide analüüsi jaoks). Kõikide sobilike andmete kasutamine aitab ületada kontiigide pikemaks järjestusteks liitmise probleemi. (Gritsenko jt., 2012)

GRASS kasutab raamjärjestamise probleemi lahendamise jaoks sarnaselt MIP Scaffolder'ile MIP formulatsiooni: graaf jagatakse esmalt väiksemateks osadeks ja lahendatakse eraldi (Salmela jt., 2011). Kontiigide ja raamjärjestuste protsessimine toimub eelnevalt vastavate järjestuste jaoks arvutatud ennustatud tühimiku suuruse alusel. Kui tühimiku suurus ei eelda ülekatvusi, asetatakse kontiigid raamjärjestusse. Kui tühimiku suurus eeldab pikemaid ülekatvusi, poolitatakse raamjärjestus. Alla 50 bp tühimike suuruste korral järgnevad kontiigid üksteisele ilma ülekatvusteta. (Gritsenko jt., 2012)

GRASS oli võrdlusel SSPACE, MIP, OPERA programmidega edukam moodustades võrreldava arvu raamjärjestusi ja tehes selle juures vähem raamjärjestamise vigu tänu erinevatest allikatest pärinevate raamjärjestuse informatsiooni kombineerimisele ja kasutamisele. (Gritsenko jt., 2012)

1.5.5 SCARPA

SCARPA kasutab raamjärjestamiseks mitmete algoritmide kombinatsiooni. Vigaste kontiigide tuvastamise ja nende valimist eraldamise tulemusel pannakse raamjärjestused kokku ainult korrektsetest andmetest. (Donmez ja Brudno, 2013)

Esmalt kasutab SCARPA filtrit, et leida üles ja eemaldada mitteunikaalselt paigutunud lugemid. Igale kasutatavale raamatukogule leitakse paarislugemite vaheline järjestuse keskmine pikkus ja selle hälve. Sarnaselt programmidele MIP Scaffolder ja Bambus, koostab SCARPA graafi. (Donmez ja Brudno, 2013)

SCARPA proovib leida igale kontiigile orientatsiooni, nii et tekiks paarislugemitega võimalikult vähe ebakõlasid. Mitmed raamjärjestamise programmid eemaldavad just ebakõla tekitavaid paarislugemite ühendusi, eeldades, et neid põhjustavad näiteks kimäärsed lugemid. Tegelikult võib probleem peituda valesti kokku pandud kontiigides, mida oleks otstarbekam ühenduse asemel eemaldada. Pärast suuna määramist igale kontiigile proovitakse neid lineaarselt järjestada ebahariliku tsüklilise graafi lahendamise teel. Konfliktid ühendused tekitavad graafi ebahariliku tsüklilisuse. Minimaalse arvu äärte eemaldamisel suudetakse tsüklid lineariseerida. Viimases etapis proovitakse leida igale kontiigile asetus nii, et nendevaheline distant oleks kooskõlas paarislugemites sisalduva infoga. (Donmez ja Brudno, 2013)

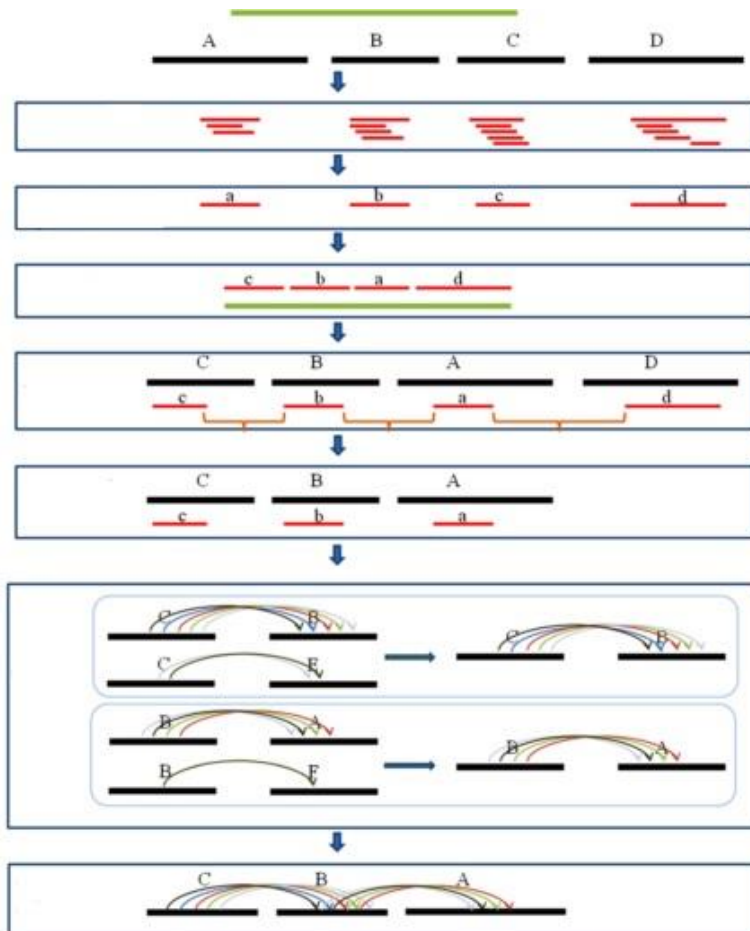
SCARPA-t võrreldi teiste raamjärjestamise programmidega (SSPACE, MIP, SOPRA). SCARPA väljundiks on kõrge täpsusega raamjärjestused, mis on vähemalt sama pikad kui teiste meetoditega saadud järjestused. Eelnimetatud nelja programmi tööaegu võrreldes oli SCARPA-st kiirem vaid SSPACE. SCARPA oluliseks eeliseks teiste meetodite ees on väike mälunõudlus. Siiski, SCARPA ei ole sobilik suurte ja keeruliste genoomide jaoks, kuna see oleks arvutuslikult liiga ressursinõudlik. (Donmez ja Brudno, 2013)

1.5.6 L_RNA_Scaffolder

L_RNA_Scaffolder on esimene meetod, mis kasutab transkriptoomi lugemeid (RNA-seq) kontiigide kombineerimiseks raamjärjestusteks. Kuna antud lugemite puhul on tegemist eksonite järjestustega, saab neid kasutada kontiigide korrektseks liitmiseks (orientatsioon ja järgnevus). (Xue jt., 2013)

Esmaseks ülesandeks on transkriptide kogust dominantse ekspressiooniga alustranskripti leidmine optimaalsete parameetri näitude abil: maksimaalne introni pikkus (MIL), minimaalne kattuvuse pikkus (MLC) ja minimaalne protsentuaalne sarnasus (MPI) (joonis 8). Kasutajal tuleb parameetri väärtused vastavalt seadistada. Juhul, kui muuta parameetreid liiga rangeks, jääb väga vähe transkripte edasiseks kasutuseks. (Xue jt., 2013)

L_RNA_Scaffolder suudab anda pikemaid ja korrektselt kokku pandud raamjärjestusi, millelt on hõlpsam gene tuvastada. Pikendatud transkriptomid suudavad suurendada terve genoomi terviklikku hõlmamist. Raamjärjestatud genoomi üleüldist N50 pikkust suudeti võrreldes esialgsega kahekordistada. (Xue jt., 2013)



Joonis 8. L_RNA Scaffolder'i tööprotsess. Rohelisega tähistatud transkriptid joondatakse mustaga märgitud genoomsele järjestusele. Joondatud järjestused (transkriptid) jaotatakse lugemi alguskohtade alusel erinevatesse blokkidesse a, b, c ja d. Transkriptid, mida ei suudetud eelmises etapis täielikult joondada, valitakse alustranskriptideks (kõige pikemad). Toimub blokkide järjestamine vastavalt neis sisalduvate lugemite koordinaatidele ja nende ümberpaigutamine (c, d, a ja d). Genoomsed järjestused reastatakse blokkide järgi, millele nad eelnevalt esimeses etapis joondati. DNA järjestus kahe kõrvuti asetseva bloki vahel on potentsiaalne intron. Kui intron on võrreldes teistega liiga pikk, filtreerib programm selle välja (a ja d vaheline kaugus). Fragmentide vahele luuakse lugemite abiga sildühendused. Algufragmendiks valitakse järjestus, millel on arvuliselt kõige rohkem toetavaid sidususi. Viimases etapis leitakse raamjärjestusrada, mis koosneb vähemalt kahest fragmendist. (Xue jt., 2013)

Mitmete genoomide lõpetamist takistab kõrge polümorfisuse aste. Tavapärased raamjärjestamise programmid lahutavad säärased piirkonnad ja sellega ei ole geeni järjepidevus tagatud. L_RNA_Scaffolder saavutas häid tulemusi pärlikarbi (*Pinctada fucata*)

puhul ja on sobilik tema geenide ennustusvahendiks. Siiski, selle raamjärjestamise meetodi efektiivsust pärsib suhteliselt kõrge veamäär (inversioonid, relokatsioonid, translokatsioonid). Geenide ennustamiseks on L_RNA Scaffolder võrreldes teiste meetoditega küll parem, aga referentsjärjestustega võrdlus näitas, et üldine kontiigide kokkupanek raamjärjestusteks oli probleemne: üle 18% ühendustest ei olnud omavahel kooskõlas. (Xue jt., 2013)

L_RNA_Scaffolder on üldiselt edukam teistest raamjärjestamise programmidest, kui raamatukogude lugemite inserdi suurus jääb vahemikku kuni 10 kbp. Pikema inserdiga raamatukogude puhul (>35 kbp) saavutavad varasemad meetodid parema N50 väärtuse. Erandi moodustas MIP programm, mis saavutas suurima N50 väärtuse 10 kbp raamatukogu kasutamisel. Kui raamjärjestamise eesmärgiks on saavutada võimalikult suur transkriptide katvus, siis on antud programm senistest meetoditest efektiivsem. (Xue jt., 2013)

1.6 Parameetrid genoomide kokkupanemise kvaliteedi hindamiseks

Parima genoomide kokkupanemise meetodi väljaselgitamiseks on läbi viidud mitmeid mahukaid uurimisi (Bradnam jt., 2013; Earl jt., 2011; Salzberg jt., 2012). Esimeseks nende seas on Assemblathon 1, mille tulemused avalikustati 2011. aasta lõpus (Earl jt., 2011). Sellest projektist võtsid osa mitmed teadusgrupid, kes enda loodud programmide abil proovisid rekonstrueerida korraldajate etteantud lugemite põhjal võimalikult täpse ja tervikliku esialgse genoomi järjestuse. Kui osalejad suutsid oma ülesande täita ettenähtud aja jooksul, hindasid ja võrdlesid organiseerijad nende töötulemusi teiste võistlejatega. Kasutatavateks andmeteks olid simuleeritud Illumina lugemid, mille valikut on kritiseeritud, kuna ükski sünteetiline genoom ei suuda piisaval määral imiteerida tegelikku olukorda (Baker, 2012). Need programmid, mis annavad häid lahendusi simuleeritud andmetega, ei pruugi saavutada samaväärseid tulemusi reaalse genoomidega. Teine projekt, mis samuti kasutas sünteetilisi järjestusi, on dnGASP.¹⁰ Projekti dnGASP elluviimisel polnud rõhuasetus ideaalilähedase programmi leidmine, vaid avastada selle ettevõtmise käigus viise kokkupandud genoomide hindamiseks ja edaspidiseks täiustamiseks. Enne, kui Assemblathon 1 projekt lõpetati, algatati juunis 2011 Assemblathon 2 (Bradnam jt., 2013). Erinevus nende kahe Assemblathon projekti vahel seisneb töödeldavates andmetes. Erinevalt esimesest, kasutati Assemblathon 2 läbiviimisel kolme erineva selgroogse liigi reaalseid genoomseid

¹⁰ <http://cnag.bsc.es/>

järjestusi: viirpapagoi (*Melopsittacus undulatus*), *Maylandia zebra* ja kuningboa (*Boa constrictor constricto*). Sarnase eesmärgiga projekt on GAGE, mis erinevalt teistest ei muutunud võistluseks osalejate vahel, sest GAGE-s rakendasid uurimistöö autorid ise teiste kirjutatud programme (vaikimisi määratud parameetritega) oma andmete peal ja teostasid võrdluse (Salzberg jt., 2012).

Kõik eelpool mainitud projektid jõudsid sarnasele järeldusele: isegi kõige paremad genoomi kokkupanemise programmid teevad arvukaid ja olulisi vigu (Baker, 2012). Tähtis on leida meetrikuid, mis suudaksid juhatada vastavalt eesmärgile ja konkreetsest liigist pärinevatele andmetele sobiva programmi. Sobilike parameetrite valimine on problemaatiline, sest nende vahel esineb lõivsuhe: kontiigide/raamjärjestuste pikkus ja nendes esinevad vead. Meetrikute rohkust näitab asjaolu, et Assemblathon 2 valis saja meetriku hulgast hindamiseks kümme näitajat, mille põhjal prooviti kokkupanud genoomi kvaliteedis selgusele jõuda (Bradnam jt., 2013).

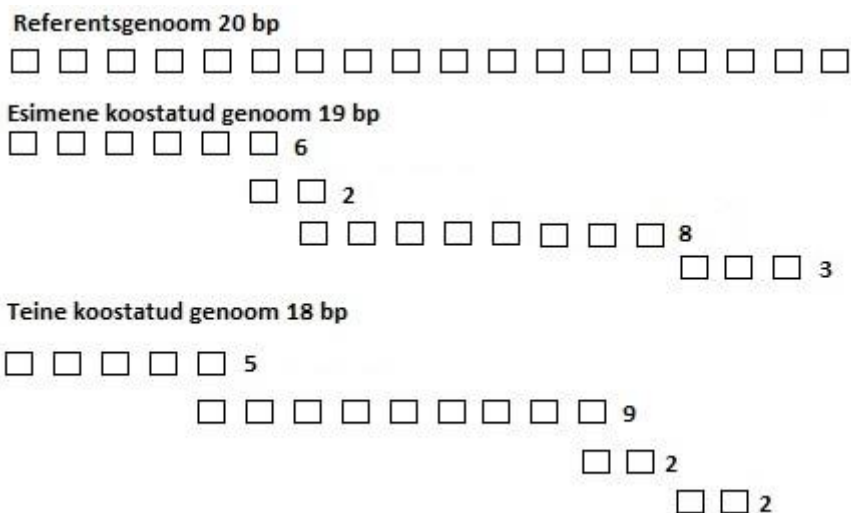
Laialdasemalt kasutusel olevad meetrikud (N50, väikeste kontiigide arv) hindavad kokkupanud genoomi eelkõige kontiigide suuruse, mitte kvaliteedi ja täpsuse alusel, mis kajastaksid paremini nende tegelikku väärtust (Narzisi ja Mishra, 2011; Vezzi jt., 2012b). Kokkupanud genoomide hindamine oleneb sellest, kas on olemas kõrgkvaliteetne referentsgenoom, millega hinnatavat kõrvutada, või mitte. Täiesti uue genoomi puhul referents puudub ja hindamiseks kasutatakse kvantitatiivseid meetrikuid. Parameetreid jagatakse nii kvantitatiivseteks, mis hindavad kokkupanud genoomide arvulisi väärtusi, kui ka kvalitatiivseteks, mis viitavad vigu sisaldavatele piirkondadele. (Baker, 2012)

1.6.1 Kvantitatiivsed parameetrid

Mitmete genoomide kokkupanemise programmide tulemuste kvaliteedi hindamiseks kasutatakse parameetrit N50 (Dohm jt., 2007; Vezzi jt., 2012b; Pop jt., 2004). Tegemist on suuruse arvulise väärtusega, mida määratletakse kui kontiigi vähimat pikkust, millest võrdsed või pikemad kontiigid katavad 50% kogu genoomsest järjestusest (näide 1). N50 arvutamiseks järjestatakse kontiigid pikematest lühemateni. Seejärel liidetakse järjestuste pikkused alustades pikimast kuni liidetavate summa on arvuliselt võrdne poolega kõikide kontiigide pikkuste summaga kokkupanud genoomis. Geenide annotatsiooni (asukoha kirjeldamise) jaoks peavad kontiigid olema piisava pikkusega, et kataksid täispikkuses kõik kodeerivad järjestused ja soovitatavalt ka intronid. Teisalt on pikad raamjärjestused kasutatud, kui nad on valesti kokku pandud. Seega on sellel meetrikul nõrk külg: ta ei pruugi peegeldada

kokkupandud genoomi tegelikku kvaliteeti. Ideaalis peaks genoom olema võimalikult terviklik. Terviklikkuse probleemi olemasolu ilmnes, kui üritati meritupe (*C. intestinalis*) genoomi taas kokku panna. Kui varasema v1.95 kokkupandud genoomi N50 väärtuseks oli 234 500 bp, siis uuema v2.0 versiooni puhul oli see kümme korda suurem (2 571 800 bp). Hoolimata pikematest raamjärjestustest, sisaldas uuem versioon oluliselt vähem kõrgelt konserveerunud gene (Parra jt., 2009). Genoomi terviklikkuse hindamine toetudes ainult N50-le võib anda soovitud vastupidise tulemuse. N50 lähedane mõõdik on NG50, mille mõlema arvutamise lähtealuseks on erinevad kontekstid. N50 arvutamisel peetakse silmas koostatud genoomi suurus, samal ajal kui NG50 väärtuse leidmisel kasutatakse eeldatava genoomi kogusuurust.

Näide 1. Kahele koostatud genoomile N50 väärtuse leidmine.



Esimese koostatud genoomi N50 väärtuse leidmiseks järjestame kontiigid: 8 bp, 6 bp, 3 bp, ja 2 bp. $8 + 6/19 = 74\%$. Järelikult on N50 6 bp. Teise koostatud genoomi N50 on 5 bp. $9/18 = 50\%$. Esimese koostatud genoomi NG50 leidmiseks tuleb samuti kontiigid kahaneva suuruse järjekorras reastada ent leitakse võttes terve kokkupandud genoomi suuruse asemel eeldatava genoomi kogusuurus ehk antud juhul 20 bp. NG50 on 6 bp. $8 + 6/20 = 70\%$. Teise konstrueeritud genoomi NG50 on $9 + 5/20 = 70\%$.

Rämps kontiig on üksik kontiig, mis on lühem kui 200 bp. Selliste lühikeste kontiigide osahulka genoomis väljendatakse protsentuaalselt. Neid ei kasutata genoomsetes analüüsid, kuna nende pikkus ei võimalda edukat geenide annotatsiooni. (Salzberg jt., 2012)

Raamjärjestuste arv hindab, kui täielikult on genoom kokku pandud. Ideaalsel juhul suudab raamjärjestamise programm kontiigidest ühe tervikliku genoomse järjestuse moodustada.

1.6.2 Kvalitatiivsed parameetrid

CEGMA on arvutuslik meetod, mis võimaldab hinnata kokkupandud genoomi terviklikkust konserveerunud geenide leidumise alusel. Kui CEGMA ei leia evolutsiooniliselt konserveerunud järjestusi, võib see viidata puudulikult kokkupandud genoomile (Parra jt., 2007). Kõrgelt konserveerunud geenid eukarüootides on näiteks *hox* geenid, mis vastutavad posterioorse-anterioorse kehatelje väljakujunemise eest varajases embrüonaalses arengus.

FRC (*Feature Response Curve*) on kvalitatiivne meetrik, mis ei vaja kokkupandud genoomi kvaliteedi hindamiseks referentsjärjestuste olemasolu (Vezi jt., 2012a). Antud meetod leiab igale veatüübile, mida kontiigid sisaldavad, arvulised väärtused, kusjuures vigade omavahelisi suhteid arvesse ei võeta. Antud parameetri edasiarenduseks on loodud meetod FRC^{bam} (Vezi jt., 2012b). Paarislugemid paigutatakse hinnatavale koostatud genoomi kontiigidele, mille tulemuseks saadakse BAM formaadis fail (Li jt., 2009). BAM fail on FRC^{bam} -ile sisendiks, mille alusel leitakse igale veatüübile arvulised väärtused (tabel 4), mis aitavad tuvastada probleemseid piirkondi konstrueeritud genoomis. Kontiigid järjestatakse pikimast lühemateni ja ainult pikemaid kontiige, mille kohta arvatud erinevate veatüüpide väärtuste summa jääb alla kindlaksmääratud lävendi, kasutatakse genoomi katvuse arvutamiseks. Selline kontiig tähistab üksikut punkti kõvera graafikus (joonis 9). FRC^{bam} väljundiks on kolm faili: üldine veatüüpide summaarne FRC-kõver, spetsiifiline FRC-kõver iga veatüübi kohta ja detailsem fail, milles sisaldub info igas kontiigis esineva veatüübi kohta. See meetod on paindlik ja pakub kasutajale võimalust vajadusel ka uute veatüüpide lisamist leidmaks teistsuguseid probleemseid piirkondi. FRC^{bam} on võimeline tuvastama valesti kokkupandud piirkondi vaid unikaalsete lugemite joendamisel. Siiski on see ainus meetod, mis võimaldab hinnata kokkupandud genoomi kvaliteeti referentsgenoomi puudumisel. (Vezi jt., 2012b)

Tabelis 4 on välja toodud FRC^{bam} -kõvera abil arvatavad veatüüpide väärtused kasutades selleks nii kaaslugemeid kui paarisotsalisi lugemeid. Kaaslugemeid kasutatakse suuremate insertioonide/deletsioonide tuvastamiseks, kuid paarisotsalisi lokaalsemate sündmuste leidmiseks. Lisaks arvutatakse CE-statistik ehk kokkusuruvus- ja laiendusstatistika, kus kõrged väärtused viitavad laiendusvigadele, mida põhjustavad insertioonid (Zimin jt., 2008). Madalad väärtused omakorda viitavad deletsioonide tõttu lühenenud piirkondadele. (Vezi jt., 2012b)

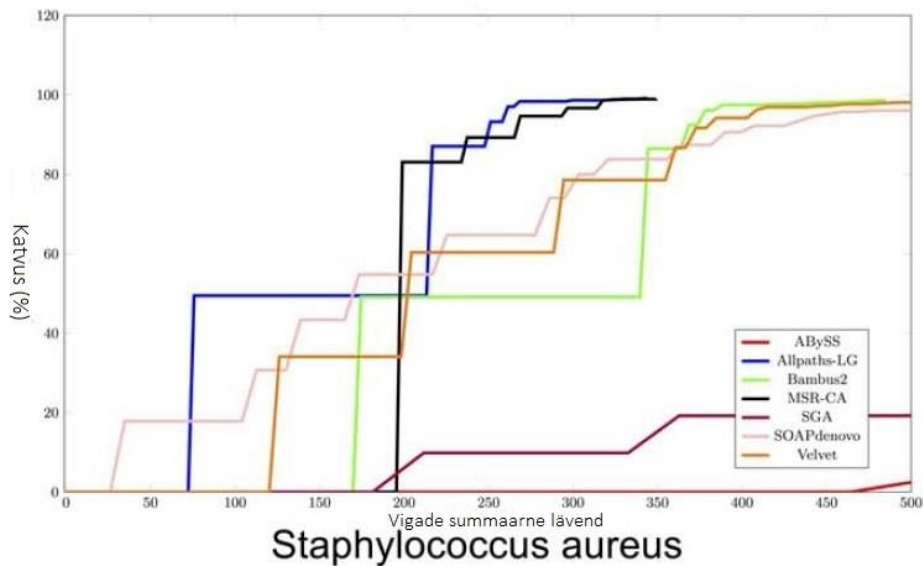
Esmalt leitakse genoomi piirkondades sisalduvad vead nii FCR-kõvera abiga kui ka iseseisvalt programmi *dnadiff*ga (Kurtz jt., 2004). *Dnadiff* on analüüsirakendustarkvara,

millega saab tuvastada detailselt kahe genoomi või järjestuse vahelisi erinevusi. Mõlemate väljundtulemuste võrdluse alusel on võimalik edasiselt arvutada väärtusi kahe parameetri jaoks: tundlikkus ja spetsiifilisus. Tundlikkuse ja spetsiifilisuse alusel suudetakse FCR^{bam}-kõveraga eristada genoomsete piirkondade keerukustaset. (Vezzi jt., 2012b)

Tabel 4. FRC^{bam}-kõveraga arvutatavate veatüüpide kirjeldused.¹¹

Parameetri nimi	Kirjeldus
LOW_COV_PE	Madala katvusega piirkondade arv (kõik joondatud lugemid)
HIGH_COV_PE	Kõrge katvusega piirkondade arv (kõik joondatud lugemid)
LOW_NORM_COV_PE	Madala PL katvusega piirkondade arv (ainult korrektselt paigutatud paarid)
HIGH_NORM_COV_PE	Kõrge PL katvusega piirkondade arv (ainult korrektselt paigutatud paarid)
COMPR_PE	Deletsioonide arv PL vahel (madal CE-statistik)
STRECH_PE	Insertsioonide arv PL vahel (kõrge CE-statistik)
HIGH_SINGLE_PE	Kõrge arvukusega PL lugemid, kus üks paariline on paigutamata
HIGH_SPAN_PE	Kõrge arvukusega PL lugemid, kus paarilised on paigutatud erinevasse kontiigi/raamjärjestusse
HIGH_OUTIE_PE	Kõrge arvukusega PL lugemid, mis on vale orientatsiooniga või liiga suurte vahekaugustega
COMPR_MP	Deletsioonide arv KL vahel (madal CE-statistik)
STRECH_MP	Insertsioonide arv KL vahel (kõrge CE-statistik)
HIGH_SINGLE_MP	Kõrge arvukusega KL lugemid, kus üks paariline on paigutamata
HIGH_SPAN_MP	Kõrge arvukusega KL lugemid, kus paarilised on paigutatud erinevasse kontiigi/raamjärjestusse
HIGH_OUTIE_MP	Kõrge arvukusega KL lugemid, mis on vale orientatsiooniga või liiga suurte vahekaugustega

¹¹ (Vezzi jt., 2012b)



Joonis 9. FRC^{bam}-kõverad analüüsis *S. aureuse* järjestusi. X-telg tähistab kindlaksmääratud vigade summaarset lävendit. Vaid selliseid kontiige kasutatakse genoomi ligikaudse katvuse arvutamiseks (y-telg), mille vigade esinemine ei ületa antud lävendit. Parimad tulemused ja ühtlasi järsemad kurvid on saavutatud MSR-CA ja Allpaths-LG-ga konstrueeritud genoomidega. (Vezzi jt., 2012b)

Tundlikkus on määr, väljendab õige positiivse (*dnadiff* märkis piirkonna valesti kokkupanduks ja märgitud on üks või enam viga) tulemuse suhet kogu tulemusesse koos valenegatiivsega. See parameeter võimaldab FRC^{bam}-kõvera abil tuvastada problemaatilisi piirkondi. (Vezzi jt., 2012b)

Spetsiifilisus on määr, mis väljendab vale negatiivse tulemuse (*dnadiff* märkis piirkonna valesti kokkupanduks, kuigi märgitud pole ühtegi viga) suhet kogu tulemusesse (koos valepositiivsetega). See näidik eristab problemaatilisi piirkondi teistest regioonidest. Valepositiivsed ja –negatiivsed aitab tuvastada võrdlev analüüsiskript *dnadiff*. (Vezzi jt., 2012b)

2. Uurimus

2.1 Töö eesmärgid

Käesoleva töö tegevuskavandi eesmärgiks on anda juhised, mis võimaldavad tulevikus leida sobivate parameetrite kasutamise läbi võimalikult täpne kokkupandud raamjärjestamise protsessi läbinud genoom reaalse te järgestusandmete alusel.

Selle saavutamiseks püstitatakse järgmised ülesanded:

- Võrrelda erinevate raamjärjestamise programmide efektiivsust (nii kiirus kui ka kvaliteet) erinevate genoomi kokkupanemise programmidega konstrueeritud järjestuste alusel genoomi kokkupanemistulemuste parandamiseks.
- Kasutada eri tüüpi reaalseid lugemeid ja kontiige, mitte simuleeritud andmeid, et analüüs vastaks tegelikele probleemidele suuremahulistes sekveneerimisprojektides.
- Anda hinnang parameetritele, mille alusel raamjärjestuste korrektsust hinnatakse.

2.2 Materjal ja metoodika

2.2.1 Kasutatavad raamjärjestamise programmid

Raamjärjestamise programmid, mida töös võrdlustes kasutatakse, on järgmised:

Programm	Ligipääs programmile
Bambus 2.33	http://sourceforge.net/projects/amos/files/bambus/2.33/
SSPACE 2.0	http://www.baseclear.com/lab-products/bioinformatics-tools/sspace-standard/
MIP Scaffold 0.6	http://www.cs.helsinki.fi/u/lmsalmel/mip-scaffold/
GRASS 0.003	https://github.com/AlexeyG/GRASS
SCARPA 0.241	http://compbio.cs.toronto.edu/hapsembler/scarpa.html
L_RNA_Scaffold	http://www.fishbrowser.org/software/L_RNA_scaffold/index.php?action=isin&do=download

Raamjärjestamise programmide lühikirjeldused on esitatud töö kirjanduse ülevaates ja valitud vastavalt aspektidele: a) kättesaadavus; b) kasutaksid sisendina erinevaid andmeid (454 ja/või

Illumina lugemeid, L_RNA_Scaffolder transkriptoomi lugemeid) c) erinevad tööstrateegiad (heuristiline (Bambus, SSPACE), graafi moodustamine (MIP Scaffolder, GRASS, SCARPA) ja eksonite paigutamine kontiigidele (L_RNA_Scaffolder)).

2.2.2 Kasutatavad andmed

Raamjärjestamise programmide võrdluse läbiviimise jaoks kasutatakse reaalseid (mitte simuleeritud) genoomseid kolmest erinevast allikast pärinevaid andmeid, millest kaks on võetud erinevatest uurimustest: GAGE (Salzberg jt., 2012) ja Assemblathon 2 (Bradnam jt., 2013). Töös kasutatakse eelkõige neid genoome ning järjestusi, mille jaoks on referents olemas. See võimaldab pärast raamjärjestuste kokkupanemise korrektsust hinnata.

Kuna antud uurimuses soovitakse hõlmata ka organisme, kellel on olemas transkriptoomi andmed, valisime kolmandaks andmestikuks ühe koonusteo liigi *Conus consors* lugemid ja kontiigid. Lisaks on antud genoomi keskmine lugemite katvus (19x) võrreldes teistega (~40x) madalam ja võimaldab raamjärjestamise programmide võrdlusel ka seda aspekti uurida.

GAGE-st pärinev andmestik:

Lugemid	Kontiigid	Liigid
Illumina ÜL, PL, KL lugemite raamatukogud (FASTQ formaat)	Kokkupandud kontiigid 8 erineva programmi poolt (FASTA formaat)	<i>Staphylococcus aureus</i> , <i>Rhodobacter sphaeroides</i> , <i>Homo sapiens</i> 14. kromosoom

Assemblathon 2-st pärinev andmestik:

Lugemid	Kontiigid	Liik
Illumina ja 454 PL lugemite raamatukogud (FASTQ formaat)	Kokkupandud kontiigid 12 erineva programmi poolt (FASTA formaat)	<i>Boa constrictor constrictor</i>

Koonusteo andmestik:

Lugemid	Kontiigid	Liik
Illumina PL, KL ja 454 ÜL lugemite raamatukogud (FASTQ formaat). RNA-seq PL lugemite raamatukogud kaheksast koest. (FASTQ formaadis).	Newbleriga kokkupandud kontiigid (FASTA formaat)	<i>Conus consors</i>

2.2.3 Kasutatavad parameetrid

Programmide tõhususe hindamise jaoks kasutatakse töös kvalitatiivseid ja kvantitatiivseid parameetreid.

Kvantitatiivsed: N50, NG50, rämpskontiigide osahulk, moodustatud raamjärjestuste arv.

Kvalitatiivsed:

Parameeter	Ligipääs parameetri rakendustarkvarale
CEGMA	http://korflab.ucdavis.edu/Datasets/cegma/#SCT6
FRC ^{bam}	https://github.com/vezzi/FRC_align.git

CEGMA parameetri jooksumiseks on vaja eelnevalt installeerida WU-BLAST,¹² HMMER,¹³ GeneWise,¹⁴ geneid.¹⁵ Kasutatavaks andmebaasiks on COG.¹⁶

2.3 Tulemused ja arutelu

Raamjärjestamise programmide efektiivsuse hindamiseks valiti välja kuus parameetrit, millest 4 on kvantitatiivsed ja 2 kvalitatiivsed näitajad. Nende parameetrite alusel leitakse igale raamjärjestamise programmiga koostatud genoomile kumulatiivne z skoor, mis on iga üksiku parameetri näitaja tulemuste summa. Selle alusel saab leida, milline parameetrite kombinatsioon viib kõige täpsema tulemuseni – referentsjärjestuseni ja millised on parameetrite omavahelised suhted, kui eemaldada z skoorist konkreetse parameetri tulemus. Eesmärgiks ei ole nende tulemuste alusel luua uusi hindamisparameetreid, vaid olemasolevate alusel raamjärjestamise programmide tõhusust hinnata.

Paralleelselt leitakse programmidega konstrueeritud genoomidele kvaliteedihinnang FCR^{bam}-kõver. Antud parameetri ja kumulatiivse z-skoori väärtuste võrdlemisel saab leida, kas ja millisel määral programmidele antud kvaliteedihinnangu osas tulemused erinevad. Programmi tulemuslikkus andmete töötlemisel ja õigete intepretatsioonide leidmisel oleneb genoomi eripäradest (suurus, diploidsusaste ja kordusalade sagedus) ja uurimise eesmärkidest.

¹² <http://blast.wustl.edu>

¹³ <http://hmmmer.janelia.org>

¹⁴ <http://www.ebi.ac.uk/~birney/wise2/>

¹⁵ <http://genome.crg.es/software/geneid>

¹⁶ <http://www.ncbi.nlm.nih.gov/COG>

Hiljuti avaldati esmakordne uurimus (Hunt jt., 2014) raamjärjestamise programmide võrdlusest, kuid käesoleva töö tegevuskavand käsitleb teemat põhjalikumalt ja lähtub erinevatest aspektidest. Hunt jt., 2014 uurimus põhineb paljuski simuleeritud andmestikul, kuid antud töös kasutatakse ainult reaalseid andmeid (lugemid ja kontiigid), mis sisaldavad nii valestipaardumisi kui ka valesti kokkupandud järjestusi. Ka puudub Hunt, jt. 2014 uurimuses võrdlus uuemate meetodite vahel, mis oskavad kasutada RNA-*seq* andmeid ning võrdlus madala katvusega kokkupandud genoomi jaoks raamjärjestuste leidmiseks. Selle jaoks kasutatakse käesolevas töös *Conus consors* genoomi RNA-*seq* andmeid. Erinevalt käesolevast tegevuskavandist pole varasemalt võrreldud eri tüüpi lugemivariantide (Illumina vs 454) kasutamisest tingitud mõju raamjärjestamise programmide efektiivsusele. Täiesti tähelepanuta on jäänud genoomi kokkupanemise metoodika mõju raamjärjestamise protsessile, sest Hunt jt. 2014 kasutasid vaid enda poolt Velveti rakendustarkvaraga konstrueeritud kontiige. Seepärast kasutatakse GAGE ja Assemblathon 2 projektist pärinevaid kokkupandud kontiige, mis on erinevate meetoditega loodud. Selle juures kasutasid Assemblathon 2 projektis osalevad meetodite väljatöötajad ekspertidena optimaalseid parameetreid oma programmide jooksutamiseks, mitte vaikumisi väärtuseid.

Käesoleva töö lähenemisviis raamjärjestamise protsessi probleemile arvestab erinevaid faktoreid alates sisendandmetest kuni programmideni välja ja proovib selgitada, milline kumulatiivne mõju tervikule (kokkupandud genoomi täpsusele) avaldub.

Kokkuvõte

Sekveneerimistehnoloogite areng on tänapäeval olnud murranguline bioteaduste vallas. Sekveneerimine on muutunud ressursside seisukohalt mõeldavaks ka väiksemate uurimisrühmade jaoks. See asjaolu on mõjutanud omakorda mitmete erinevate genoomiandmete töötlusprogrammide arendamist. Siiski ei jõua paljud sekveneerimisprojektid täielikult lõpetatud kokkupandud genoomini, sest ei viida läbi viimast etappi – raamjärjestamist.

Selle töö tegevuskavandi eesmärgiks on anda kasutajale infot, millised raamjärjestamise programmid sobivad kõige paremini pro- ja eukariootide genoomide kokkupanemiseks või millised lugemid (Illumina vs 454) milliste programmidega paremini kokku sobivad. Lisaks näitavad reaalsete andmete peal tehtud katsed ära programmide nõrgad kohad ning annavad soovitusi nende edasiseks arendamiseks. Samuti on kasutaja jaoks oluline saada tagasisidet programmide jõudluse (tööaja ja mälu kasutus) kohta nii suurte kui ka väikeste genoomide raamjärjestamise korral.

Lisaks on oluline kasutatavate hindamiskriteeriumite asjakohasuses, mille abil selgitatakse, kas konstrueeritud genoom on võimalikult täpne. Näidatakse, kas tüüpiliselt genoomi kvaliteedi hindamiseks kasutavad kvantitatiivsed parameetrid (nt. N50) on piisavad optimaalsema raamjärjestamise programmi valimiseks oma projekti tarbeks või on vajalik kasutada uusi (nt. FCR^{bam}) kvalitatiivseid parameetreid.

Summary

The comparison of scaffolding programs using previously assembled genome sequences

The most expensive sequencing project since today is the Human Genome Project. 3.5 billion dollars for a project is far too expensive and this prompted for developing faster, cheaper methods for revealing the order of nucleotides. The rapid growth of the new Next Generation Sequencing methods lead to another problem: what to do with the growing number of data? More data is created than analysed. The aim of sequencing is to recreate possibly accurate genomes for making further interpretations while researching evolution or molecular diagnostics. So most genome projects end before the scaffolding stage with raw assembly: much potential information will never be seen.

The aim of this study is to give information about the aspects which one should consider when choosing a suitable scaffolding program. Also to give answer to questions like which program is optimal for prokaryotic or eukaryotic genomes and what kind of reads match with a specific program. One should keep in mind that testing programs with real not simulated data, will show their weaknesses and it can give useful recommendations for program developers for the future. Another purpose of this thesis is to give information about the running time and memory efficiency of the tools while analysing genomes with different complexity.

An important aspect of evaluating programs are the parameters for estimating scaffolding quality. This work finds out if the widely used quantitative parameters (e.g. N50, number of scaffolds) are sufficient to find the optimal scaffolder or is it necessary to use new qualitative parameters (e.g. counting CEGMA core genes, FCR^{bam}). Parameters will be tested using six different scaffolding programs on real genomic data from five distant species.

Kasutatud kirjandus

- Baker, M. (2012). De novo genome assembly: what every biologist should know. *Nat. Methods* 9, 333–337.
- Batzoglou, S., Jaffe, D.B., Stanley, K., Butler, J., Gnerre, S., *jt.* (2002). ARACHNE: A whole-genome shotgun assembler. *Genome Res.* 12: 177-189.
- Boetzer, M., Henkel, V.C., Jansen, J.H., Butler, D., Pirovano, W. (2011). Scaffolding pre-assembled contigs using SSPACE. *Bioinformatics* 27(4): 578-579.
- Bradnam, K.R., Fass, J.N., Alexandrov, A., Baranay, P., Bechner, *jt.* (2013). Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species. *GigaScience* 2 (1): 10.
- Butler, J., MacCallum, I., Kleber, M., Shlyakhter, I.A., Belmonte, M.K., *jt.* (2008). ALLPATHS: de novo assembly of whole-genome shotgun microread. *Genome Res.* (5): 810-20.
- Compeau, P.E.C., Pevzner, P.A., Tesler, G. (2011). How to apply de Bruijn graphs to genome assembly. *Nat. Biotechnol.* 29, 987-991.
- Desai, A., Marwah, V.S., Yadav, A., Jha, V., Dhaygude, K., *jt.* (2013). Identification of Optimum Sequencing Depth Especially for De Novo Genome Assembly of Small Genomes Using Next Generation Sequencing Data. *PLoS ONE* 8(4): e60204.
- Dohm, J.C., Lottaz, C., Borodina, T., Himmelbauer, H. (2007). SHARCGS, a fast and highly accurate short-read assembly algorithm for de novo genomic sequencing. *Genome Res.* 17 (11): 1697-706.
- Donmez, N., Brudno, M. (2013). SCARPA: scaffolding reads with practical algorithms. *Bioinformatics* (4): 428-34.
- Earl, D., Bradnam, K., St John, J., Darling, A., Lin, D., *jt.* (2011). Assemblathon 1: A competitive assessment of de novo short read assembly methods. *Genome Res.* (12): 2224-41.
- Fuchs, R. (1997). Grouper — creation of marker sets for multiplexed genotyping. *Comput. Appl. Biosci.* 13 (3): 239-41.
- Godson, G.N, Barrell, B.G., Staden, R., Fiddes, J.C. (1978). Nucleotide sequence of bacteriophage G4 DNA. *Nature* 276: 236-247.
- Gomez-Alvarez V., Teal, T.K., Schmidt, T.M. (2009). Systematic artifacts in metagenomes from complex microbial communities. *Isme. J.* 3: 1314–1317.

- Gritsenko, A.A., Nijkamp, J.F., Reinders, M.J.T., de Ridder, D. (2012). GRASS: a generic algorithm for scaffolding next-generation sequencing assemblies. *Bioinformatics* 28 (11): 1429-37.
- Hunt, M., Newbold, C., Berriman, M., Otto, T.D. (2014). A comprehensive evaluation of assembly scaffolding tools. *Genome Biol.* 15: R42.
- International Human Genome Sequencing Consortium. (2004). Finishing the euchromatic sequence of the human genome. *Nature* 431: 931-945.
- Koren S, Treangen T.J., Pop M. (2011). Bambus 2: scaffolding metagenomes. *27* (21): 2964-71.
- Kidd, J.M., Cooper, G.M., Donahue, W.F., Hayden, H.S., Sampas, N., *jt.* (2008). Mapping and sequencing of structural variation from eight human genomes. *Nature* 453: 56–64.
- Kurtz, S., Phillippy, A., Delcher, A.L., Smoot, M., Shumway, M., *jt.* (2004). Versatile and open software for comparing large genomes. *Genome Biol.* 5: R12.
- Langmead, B., Trapnell, C., Pop, M., Salzberg, S.L. (2009). Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.* 10: R25.
- Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., *jt.* (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics* 25 (16): 2078-2079.
- Liu, L., Li, T., Li, S., Hu, N., He, Y., *jt.* (2012). Comparison of Next-Generation Sequencing Systems. *J. Biomed. Sci.* Article ID 251364.
- Luo, R., Liu, B., Xie, Y., Li, Z., Huang, W., *jt.* (2012). SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. *GigaScience* 2012, 1:18.
- Mardis, E. R. Next-Generation DNA Sequencing Methods. (2008). *Annu. Rev. Genomics Hum. Genet.* 9: 387-402.
- Margulies, M., Egholm, M., Altman, W.E., Attiya, S., Bader, J.S., *jt.* (2005). Genome sequencing in microfabricated high-density picolitre reactors. *Nature* 437 (7057): 376-80.
- Myers, E.W., Sutton, G.G., Delcher, A.L., Dew, I.M., Fasulo, *jt.* (2000). A whole-genome assembly of *Drosophila*. *Science* 287: 2196-2204.
- Meador, S., Hillier, L.W., Locke, D., Ponting, C.P., Lunter, G. (2010). Genome assembly quality: Assessment and improvement using the neutral indel model. *Genome Res.* 20: 675-684.
- Moorthie, S., Mattocks, C.J., Wright, C.F. (2011). Review of massively parallel DNA sequencing technologies. *HUGO J.* 5: 1-12.
- Morozova, O., Marra, M.A. (2008). Applications of next-generation sequencing technologies in functional genomics. *Genomics* 92 (5): 255-64.

- Mullikin, J.C., Ning, Z. (2003). The phusion assembler. *Genome Res.* 13(1): 81-90.
- Narzisi G., Mishra, B. (2011). Comparing de novo genome assembly: the long and short of it. *PLoS One* 6 (4).
- Niall, H.D. (1973). Automated Edman degradation: the protein sequenator. *Method. Enzymol.* 27: 942–1010.
- Nossa C., Havlak, P., Yue, J.-X., Lv, J., Vincent, K., *jt.* (2013). Joint assembly and genetic mapping of the Atlantic horseshoe crab genome reveals ancient whole genome duplication. <http://arxiv.org/abs/1309.7402>
- Parra, G., Bradnam, K., Korf, I. (2007). CEGMA: a pipeline to accurately annotate core genes in eukaryotic genomes. *Bioinformatics* 23 (9): 1061-7.
- Parra, G., Bradnam, K., Ning, Z., Keane, T., Korf, I. (2009). Assessing the gene space in draft genomes. *Nucleic Acids Res.* 37 (1): 289-297.
- Pop, M. (2009). Genome assembly reborn: recent computational challenges. *Briefings in Bioinformatics* 10 (4): 354-366.
- Pop, M., Kosack, D.S., Salzberg, S.L. (2004). Hierarchical scaffolding with Bambus. *Genome Res.* 14 (1): 149-159.
- Quail, A.M., Smith, M., Coupland, P., Otto, D.T., Harris, R.S., *jt.* (2012). A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers. *BMC Genomics*, 13: 341.
- Rothberg, J. M., Leamon, J.H. (2008). The development and impact of 454 sequencing. *Nat. Biotechnol.* 26: 1117-1124.
- Salzberg, S.L., Phillippy, A.M., Zimin, A., Puiu, D., Magoc, T., *jt.* (2012). GAGE: A critical evaluation of genome assemblies and assembly algorithms. *Genome Res.* 22 (3): 557-67.
- Salmela, L., Mäkinen, V., Välimäki, N., Ylinen, J., Ukkonen, E. (2011). Fast scaffolding with small independent mixed integer programs. *Bioinformatics* 27: 3259–3265.
- Sanger, F., Nicklen, S., Coulson, A.R. (1977). DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. U.S.A.* 74 (12): 5463–7.
- Shendure, J., Ji, H. (2008). Next-generation DNA sequencing. *Nat. Biotechnol.* 26, 1135 – 1145.
- Simpson, J.T., Durbin, R. (2012). Efficient de novo assembly of large genomes using compressed data structures. *Genome Res.* 22 (3): 549-56.
- Simpson, J.T., Wong, K., Jackman, S.D., Schein, J.E., Jones, S.J.M., Birol, I. (2009). ABySS: A parallel assembler for short read sequence data. *Genome Res.* 19(6): 1117–1123.

- Vezi, F., Narzisi, G., Mishra, B. (2012a). Feature-by-Feature – Evaluating De Novo Sequence Assembly. *PLoS ONE* 7 (2): e31002.
- Vezi, F., Narzisi, G., Mishra, B. (2012b). Reevaluating Assembly Evaluations with Feature Response Curves: GAGE and Assemblathon. *PLoS ONE* 7 (12): e52210.
- Voelkerding, K.V., Dames, A.S., Durtschi, J.D. (2009). Next-generation sequencing: from basic research to diagnostics. *Clin. Chem.* 55 (4): 641–658.
- Wang, J., Wang, W., Li, R., Li, Y., Tian, G., *jt.* (2008). The diploid genome sequence of an Asian individual. *Nature* 456: 60–65.
- Warren, R.L., Sutton, G.G., Jones, S.J.M, Holt, R.A. (2007). Assembling millions of short DNA sequences using SSAKE. *Bioinformatics* 23 (4): 500-501.
- Wheeler, D.A., Srinivasan, M., Egholm, M., Shen, Y., Chen, L., *jt.* (2008). The complete genome of an individual by massively parallel DNA sequencing. *Nature* 452: 872-876.
- Wu, W., Stupi, B.P., Litosh, V.A., Mansouri, D., Farley, D., *jt.* (2007). Termination of DNA synthesis by N^6 -alkylated, not 3'-*O*-alkylated, photocleavable 2'-deoxyadenosine triphosphates. *Nucleic Acids Res.* 35 (19): 6339-6349.
- Xue, W., Li, J.-T., Zhu, Y.-P., Hou, G.-Y., Kong, X.-F., *jt.* (2013). L_RNA_Scaffolder: scaffolding genomes with transcripts. *BMC Genomics* 14: 604.
- Zerbino, D.R., Birney, E. (2008). Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res.* 18: 821-829.
- Zhang, J., Chiodini, R., Badr, A., Zhang, G. (2011). The impact of next-generation sequencing on genomics. *J. Genet. Genomics* 38 (3): 95-109.
- Zimin, A.V., Smith, D.R., Sutton, G., Yorke, J.A. (2008). Assembly reconciliation. *Bioinformatics* 24: 42–5.

Kasutatud veebiaadressid

<https://www.sciencenews.org/article/gene-sequencing-future-here>

<http://gcat.davidson.edu/phast/olc.html>

<http://gcat.davidson.edu/phast/debruijn.html>

http://www.illumina.com/technology/paired_end_sequencing_assay.ilmn

http://www.illumina.com/technology/mate_pair_sequencing_assay.ilmn

<http://cnag.bsc.es/>

<http://blast.wustl.edu>

<http://hmmer.janelia.org>

<http://www.ebi.ac.uk/~birney/wise2/>

<http://genome.crg.es/software/geneid>

<http://www.ncbi.nlm.nih.gov/COG>

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina _____ Triin Edula _____

(*autori nimi*)

(sünnikuupäev: 6. märts 1991 _____)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose
Raamjärjestamise programmide võrdlus eelnevalt kokkupandud genoomijärjestuste alusel
(*lõputöö pealkiri*)

mille juhendaja on Reidar Andreson,
(*juhendaja nimi*)

1.1.reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2.üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace´i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, 23.05.2014 (*kuupäev*)