TARTU ÜLIKOOL
BIOLOOGIA- JA GEOGRAAFIATEADUSKOND
MOLEKULAAR- JA RAKUBIOLOOGIA INSTITUUT
BIOINFORMAATIKA ÕPPETOOL

Triinu Kõressaar

Metoodika praimerite automaatseks disainimiseks eukarüootsetest genoomidest

Bakalaureusetöö

Juhendaja: Maido Remm, Prof., PhD

TARTU
2004

Sisukord

Lühendid ja mõisted

Aadress – andmete asukoht arvuti mälus või kõvakettal.

Ab initio geeniennustus – ainult DNA nukleotiidsel järjestusel põhinev geeniennustus.

Amplikon – ühe PCR produktina amplifitseeritav DNA piirkond.

bp (***b****ase* ***p****air*) – aluspaar.

BLAST (***B****asic* ***L****ocal* ***A****lignment* ***S****earch* ***T****ool*) – lokaalsel joondamisel põhinev järjestuste võrdlemist teostav algoritm.

Cap – pre-mRNA 5' otsa 7-metüülguaniin-nukleotiid.

cDNA (***c****omplementary* **DNA**) – mRNA-lt pöördtranskriptaasi abil sünteesitud mRNA-ga komplementaarne DNA.

CDS (***cod****ing* ***s****equence*) – geeni kodeeriv ala, mis ei sisalda mittetransleeritavat järjestust.

DUST – programm, mis maskeerib lihtsad kordused nukleotiidses järjestuses.

EST (***E****xpressed* ***S****equence* ***T****ag*) – lühike DNA järjestus, mis on saadud cDNA 3' või 5' otsast.

FASTA (***FAST-A****ll*) – lokaalsel joondamisel põhinev järjestuste võrdlemist teostav algoritm.

Gap – järjestuste joondamisel ühte järjestusse insertsioonide/deletsioonide tõttu tekkiv (tühi) koht.

Gb – 10^9 aluspaari või 10^9 baiti.

Globaalne joondamine – kahe nukleotiidse või aminohappelise järjestuse joondamine kogu nende pikkuses.

Heuristiline algoritm – probleemi lahendamiseks ökonoomne strateegia, kus täpne lahendus on arvutuslikult ebaratsionaalne või reaalselt võimatu.

Isokoor (*isochore*) – suhteliselt homogeensete aluspaariliste koostistega (ühtlase GC protsendiga) suured regioonid eukarüootide genoomses DNA järjestuses.

Joondamine – protsess, kus kohakuti paigutatakse mitte vähem kui kaks järjestust nii, et saavutatakse suurim sarnasus antud järjestuste vahel.

kb – 10^3 aluspaari või 10^3 baiti.

Kodeerivad statistikud (*coding statistic*) – geeni võimaliku lokalisatsiooni indikaator.

Korteež (*tuple*)- kaht või enamat komponenti sisaldav andmeobjekt.

Kromosoomi bänd (*chromosome band*) – kromosoomiala, mis kromosoomi värvimisel naaberaladest selgelt eristub, olles neist heledam või tumedam.

Lokaalne joondamine – kahe nukleotiidse või valgu järjestuse mingi väiksema osa kui lühema järjestuse pikkuse ulatuses joondamine.

Mb – 10^6 aluspaari või 10^6 baiti.

Mitme järjestuse joondamine (*Multiple Sequence Alignment*) – rohkema kui kahe järjestuse üheaegne joondamine, nii et konserveerunud aminohapped või nukleotiidid joondatakse samasse veergu.

Motiiv - lühike konserveerunud regioon nukleotiidses või valgu järjestuses.

Mutatsioon – muutus indiviidi nukleotiidse järjestuse struktuuris (geeni, kromosoomi, genoomi), mis tekivad tänu vigadele DNA replikatsioonil ja DNA parandamisel ning, mis päranduvad tütarrakkudele või ka järglaspõlvkonna indiviididele.

Ortoloogsed geenid – geenid, mis esinevad erinevates liikides, kuid mis pärinevad ühest geenist nende liikide viimases ühises eellases.

PCR (***P****olymerase* ***C****hain* ***R****eaction*) – tehnoloogia uuritavate DNA piirkondade amplifitseerimiseks.

Peidetud Markovi Mudelid (*Hidden Markov Models*) – tõenäosuslikud mudelid, mida bioloogia valdkonnas kasutatakse aminohappeliste ja DNA järjestuste mustrite leidmiseks.

Pointer ehk viide – programmeerimises muutuja, mis hoiab teise muutuja aadressi või muutujate massiivi algusaadressi.

RAM (***R****andom-****A****ccess* ***M****emory*)- arvuti muutmälu, mis on arvuti keskne mäluseade, kuhu saab andmeid kirjutada ja, kust saab neid lugeda.

Sensitiivsus (Sn, *sensitivity*) – mõõt, mis väljendab programmi korrektsete tunnuste ennustamist

Spetsiifilisus (Sp, *specificity*) – mõõt, mis väljendab programmi võimet vältida reaalselt mitteeksisteerivate tunnuste ennustamist.

Sünteensed piirkonnad (*syntenic regions*) – regioonid kahe liigi genoomse DNA vahel, mis on konserveerunud.

TBLASTX – programmi BLAST versioon, kus võrreldakse kuues erinevas lugemisraamis transleeritud uuritavat nukleotiidijärjestust kasutatavas DNA järjestuste andmebaasis leiduvate kuues erinevas lugemisraamis transleeritud järjestustega.

Transitsioon – mutatsioon nukleotiidses järjestuses, kus puriin (A või G nukleotiid) asendub teise puriiniga või pürimidiin (C või T nukleotiid) asendub teise pürimidiiniga.

Transversioon – mutatsioon nukleotiidses järjestuses, kus puriin (A või G) asendub pürimidiiniga (C või T) või vastupidi.

UTR (***Unt****ranslated* ***R****egion*) - geeni mittetransleeritav regioon.

Külgnev piirkond (*flanking region*) – PCR-i amplikoniga külgneva ala, kuhu disainitakse PCR-i praimerid.

Sissejuhatus.

Genoome, mis on juba sekveneeritud ja, mis on veel sekveneerimisjärgus, on palju. Koos genoomide sekveneerimise poolt genereeritavate andmemahtude kasvamisega kasvab ka nõudlus nukleotiidse järjestuse automaatse analüüsi järele. Genoomide suurusi silmas pidades on võimatu neid manuaalselt täismahus analüüsida. Automaatse analüüsi abil on võimalik jõuda jälile paljudele bioloogilistele mudelitele (geeni nukleotiidne struktuur, promooteralade konsensusjärjestused jpm). Suur väljakutse bioinformaatika valdkonnale on kogu genoomi hõlmavate analüüside jaoks efektiivsete ja usaldusväärsete bioloogiliste töövahendite disainimine. Enim kasutatavad tehnoloogiad suuremahulistes genoomiuuringutes on uuritava DNA fragmendi paljundamine PCR-i abil ning huvi pakkuvate nukleotiidsete fragmentide uurimine kiipide peal. PCR-i õnnestumise eelduseks on kvaliteetsete praimerite olemasolu. Praimerite disainimisel tuleb arvestada paljude asjaoludega. Märkimisväärselt keeruline on praimerite disainimine suuremahuliste uuringute jaoks, kus tuleb arvestada genoomis leiduvate kordusjärjestustega ja praimerite sekundaarsete seostumiskohtadega.

Käesoleva töö esimene osa annab ülevaate genoomse DNA sekveneerimise hetkeolukorrast, kodeerivate alade annoteerimisest ning PCR-i praimerite disainimise erinevatest detailidest.

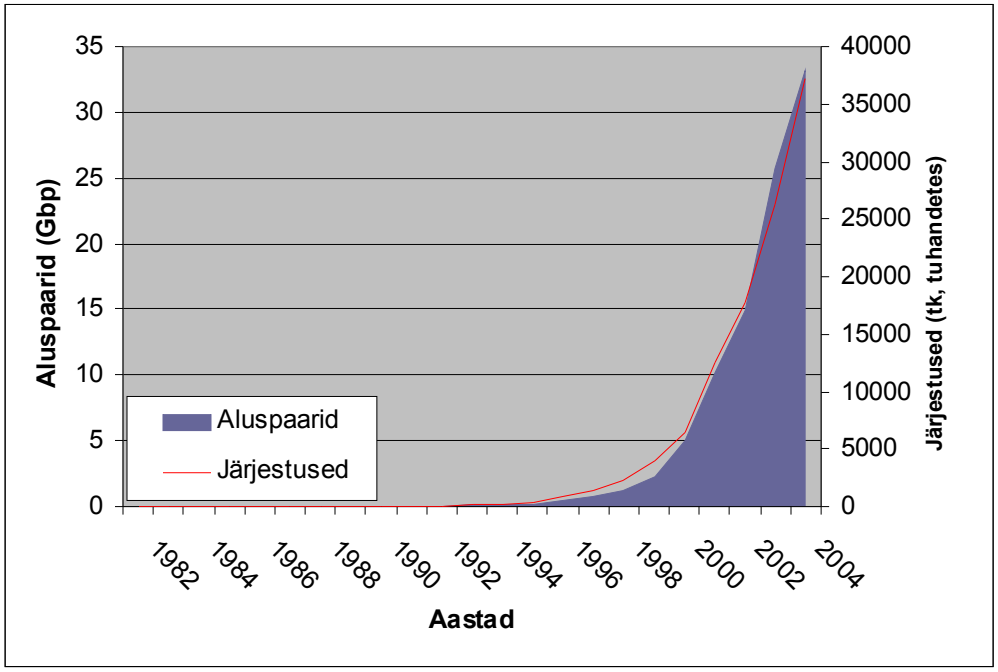
Töö teises ehk praktilises osas antakse ülevaade välja töötatud metoodikast, mis võimaldab disainida spetsiifilisi praimereid suurte eukarüootsete genoomide kodeerivate alade üles-amplifitseerimiseks. Samuti kirjeldatakse arendatud metoodika realiseerimist ning realisatsioonil genereeritud tulemusi.

I Teoreetiline osa

1. Genoomide sekveneerimine, annoteerimine ja andmebaasid

1.1 Sekveneerimise olukord

Genoomide sekveneerimise ajalugu näitab, et nukleotiidse järjestuse info akumuleerumine toimub eksponentsiaalse kiirusega (Galperin, 2004). Viimast illustreerib ka joonis 1, mis näitab andmete akumuleerumise kasvu viimastel aastatel.



Joonis 1. Andmete kasv NCBI GenBank andmebaasis ca 12 aasta pikkuse vahemiku jooksul. Esimese Y-telje väärtused korreleeruvalt X-telje väärtustega kajastavad GenBank andmebaasis sisalduvate järjestuste aluspaaride arvu ja teise Y-telje väärtused vastatavate X-telje väärtustega kajastavad üldist järjestuste arvu antud aasta jaanuari seisuga. Erinevateks järjestusteks loetakse näiteks genoomset DNA-d ja RNA-d, prekursor RNA-d

(preRNA), mRNA-d, cDNA, rRNA-d, tRNA-d, snoRNA, ekspresseeritud järjestusi (EST), STS (***S**equence-**T**agged **S**ite*) jt.

Erinevad avalikud andmebaasid sisaldavad paljude bakterite, nematoodide, pärmi *Saccharomyces cerevisiae*, *Drosophila melanogasteri*, *Arabidopsis thaliana* jt lõpetatud genoomide järjestusi. Lõpetatud genoomse järjestusega organismide hulka kuuluvad ka *Homo sapiens*, kelle ca 3 Gb suurusest genoomist on 92.64% sekveneeritud (*http://www.ensembl.org/Homo_sapiens/stats/status.html*, 17/03/2004), *Mus musculus*, kelle 2.6 Gb suurusest genoomist on järjestatud käesoleval hetkel 93.7% (*http://www.ensembl.org/Mus_musculus/stats/status.html*, 17/03/2004), *Rattus norvegicus*e genoomist on sekveneeritud 91.8% (*http://www.ensembl.org/Rattus_norvegicus/stats*, 17/03/2004). Lõpetatud järjestus tähendab, et eukromatiinse järjestuse nukleotiidse koostise täpsus on 99.99%, st lubatud on vähem kui 1 viga 10,000 aluspaari kohta ja kõik *gap*-id, mida on võimalik teadaolevate metoodikate abil täita, on täidetud. Käesoleval hetkel ei pöörata eraldi tähelepanu heterokromatiini sekveneerimisele (Schmutz et al., 2004). Arvukalt organisme, kelle genoomi järjestus on sekveneerimisjärgus, st DNA primaarsturktuur on vaid osaliselt määratud. Sellisteks organismideks on näiteks *Oryza sativa* (riis), *Gallus gallus* (kana), *Pan troglodytes* (šimpans) ja *Canis familiaris* (koer) jpt liikide esindajad.

1.2 Annotatsioonikeskused ja andmebaasid

Paralleelselt sekveneerimisandmetega kumuleeruvad ka genoomide annoteerimisandmed. Selleks, et genereeritud bioloogilistest andmetest ka kasu oleks, tuleb neid korrektselt struktureerida ja hooldada (Galperin, 2004). Just sellega tegelevad erinevad instituudid ja annotatsioonikeskused, mille ülesandeks on lisaks genoomide sekveneerimisele, assambleerimisele ja annoteerimisele ka erinevate bioloogiliste tööriistade välja töötamine ning mahukate bioloogiliste andmete säilitamine. Molekulaarbioloogia andmebaaside kollektsioon sisaldab endas rohkesti erinevaid

andmebaase, näiteks on baasid DNA nukleotiidsete järjestuste, RNA ja valgu järjestuste, metaboolsete ensüümide ja vastavate radade, signaaliradade, haiguste jpmt jaoks (Galperin, 2004). Käsitletavate organismide hulga, genoomi andmete mahu, järjestusi analüüsivate vahendite arvu ning annotatsiooniandmete genereerimise hulga poolest suurimad annotatsioonikeskused on USA-s NCBI (*National Center for Biotechnology Information*, *http://www.ncbi.nlm.nih.gov/*) ja Euroopas *The Wellcome Trust Sanger* Instituut (*http://www.sanger.ac.uk/*), mille ühisprojektina EMBL-EBI-ga (EMBL- ***E**uropean **M**olecular **B**iology **L**aboratory*, EBI - *The **E**uropean **B**ioinformatics **I**nstitute*) on valminud ENSEMBL genoomi projekt. Samuti on oluline mainida Stanfordi Ülikooli (*http://genome-www.stanford.edu/*), mille alla kuulub lisaks paljudele teistele erinevate projektide käigus genereeritud andmebaasidele ka *Saccharomyces cerevisiae* genoomi andmebaas (SGD - ***S**accharomyces **G**enome **D**atabase*). Rääkides DNA ja valgu järjestustest on viimaste aastatega oluliselt oma tähtsust tõstnud ka DDBJ (**D**NA ***D**ata **B**ank of **J**apan*, *http://www.ddbj.nig.ac.jp*) andmebaas Jaapanis. Nimetatud keskuste andmebaasid on valdavalt veebist tasuta kättesaadavad ja neid täiendatakse pidevalt (Galperin, 2004). Tuntud kui manuaalset genoomi annotatsiooni läbiviiv *Sanger* Instituudi alla kuuluv selgroogsete genoomide annotatsioonikeskus VEGA (*The **VE**rtebrate **G**enome **A**nnotation, http://vega.sanger.ac.uk/*) kätkeb endas käesoleva töö kirjutamise hetkel kolme organismi annotatsiooniandmeid– *Homo sapiens* (8 kromosoomi), *Mus musculus* (kromosoomi 13 Del36H regioon) ja *Danio rerio* (*zebrafish*). Kiiret, usaldusväärset ja tasuta kättesaadavat inimese jt osaliselt või täielikult sekveneeritud organismide erinevat tüüpi genoomi andmeid vaatamiseks ning hankimiseks pakub California Ülikooli alla kuuluv Santa Cruzi genoomi brauser (UCSC, http://genome.ucsc.edu) (Kent et al., 2002). Genoomide nukleotiidse järjestuse andmed, mida UCSC brauser pakub, on saadud, kas NCBI andmebaasist või antud organismi (rahvusvahelise) genoomi konsortsiumi poolt (näiteks *Saccharomyces cerevisiae,* so pagaripärmi järjestus on saadud *Saccharomycese* genoomi andmebaasist SGD-st, *http://www.yeastgenome.org/*) (Karolchik et al., 2004).

1.2.1 NCBI ja andmebaasid

NCBI andmebaas, mis pakub avalikult DNA ja valkude järjestusi, on GenBank. GenBank sisaldab endas töö kirjutamise hetkel rohkema kui 140,000 organismi DNA nukleotiidset järjestust. GenBank on kolme osalise - Ühendatud Kuningriikide (UK, ***U****nited* ***K****ingdom*), Ameerika Ühendriikide (USA, ***U****nited* ***S****tates of* ***A****merica*) ja Jaapani rahvusvahelise koostöö tulemusena valminud järjestuste andmebaas. Valdav osa andmetest pärinevad individuaalsetest laboritest või suuremahulistest sekveneerimisprojektidest. Järjestuse andmeid vahetatakse iga päev kolme suurema annotatsioonikeskuse vahel, milleks on EMBL andmebaas UK-s ja NCBI USA-s ning DDBJ andmebaas Jaapanis. Kõne all oleva andmebaasi uus versioon lastakse välja iga kahe kuu tagant. (Benson et al., 2003).

1.2.2 EMBL-EBI ja andmebaasid

Euroopa Bioinformaatika Instituut (EBI) koordineerib 8 andmebaasi funktsioneerimist – EMBL Nukleotiidse järjestuse andmebaas, valguandmebaasid SwissProt, UniProt, InterPro, TrEMBL, Makromolekulide Struktuuri andmebaas E-MSD, geeni ekspressiooni andmebaas ArrayExpress ja ENSEMBL-i automaatse genoomi annotatsiooni andmebaas (Kulikova et al., 2004).

Euroopa suurimasse andmebaasi EMBL (*http://www.ebi.ac.uk/embl/*) inkorporeerib, organiseerib ja distribueerib andmeid avalikest allikatest EBI (http://www.ebi.ac.uk). EMBL andmebaas on üks osa kolmeliikmelisest rahvusvahelisest koostööst GenBanki (USA) ja DDBJ-ga (Jaapan). Andmebaasi uus versioon antakse välja iga kvartali järel. EMBL andmebaasis on esindatud ca 150,000 erinevat organismi. (Kulikova et al., 2004)

ENSEMBL andmebaas on valminud koostöös EBI ja *Wellcome Trust Sanger* Instituudiga (Brooksbank et al., 2003). ENSEMBL andmebaas on kasutatav läbi võrguühenduse graafilise kasutajaliidese abil (*http://www.ensembl.org*), sisaldab 9 organismi genoomi andmeid: viie selgroogse – inimene, hiir, rott, kerakala (*fugu*) ja sebrakala (*zebrafish*), kahe ümarussi - *Caenorhabditis briggsae* ja *Caenorhabditis elegans*

ning kahe putuka - äädikakärbse *Drosophila melanogaster* ja sääse *Anopheles gambiae* (Birney et al., 2004).

2. Geenide ennustamine

Olenemata sellest, et käesolevaks hetkeks teada juba paljude organismide genoomse DNA nukleotiidne järjestus, on arusaam sellest, mis printsiibi alusel on geenid genoomile organiseeritud, kaugel täiuslikkusest (Burge, 2001). Sõltumata sellest, et tänaseks on töö geeniennustus-algoritmidega kestnud peaaegu 30 aastat (Fickett et al., 1992), on teha veel palju.

Üks peamisi põhjuseid genoomi sekveneerimisel on määrata liigi täielik geenide kogum. Organismi fundamentaalne tundmine hõlmab antud genoomi geenide arvu, struktuuri ning lokalisatsiooni ennustamist (Fickett et al., 1993).

Täpsete geenipiirkondade ennustamine on genoomikas väljakutset pakkuv arvutuslik ülesanne (Brendel, 2004). Siiani on parimad tulemused geenide ennustamisel saadud täispika cDNA ja paljude ülekattuvate EST-ide joondamisel sekveneeritud genoomsele DNA-le, kuid ükski metoodika ei suuda 100%-liselt katta kõikide geenide struktuure määravaid aspekte (Jones, 2002).

2.1 Geenide üldiseloomustus.

Geenide ennustamise protsess on eukarüootsete genoomide puhul keerulisem kui prokarüootsete korral. Viimane tuleneb eukarüootsete organismide genoomide suuremast keerukusastmest ja pikkusest võrreldes prokarüootidega. Suurte genoomidega (suurusjärk Gb) organismide korral hõlmavad geenid genoomist suhteliselt väikese osa, näiteks *Homo sapiensi* puhul ainult 27% genoomist moodustavad geenid ja kõigest 3% genoomist kodeerivad alad (Duret et al., 1995).

Eukarüootsetes organismides on geenid, kas üksteisega külgnevad (*contiguous*) või mittekülgnevad, st on või ei ole üksteisest eraldatud pikkade intergeensete DNA aladega. Viimasel juhul võib üks geen asetseda teise geeni intronis (*nested genes;* Dunham et al., 1999) või on geenid osaliselt kattuvad (*overlapping genes;* Ashburner et al., 1999). Eukarüootsetele geenidele lisab omakorda veel kompleksust geenide pidevus (*continuous*), st kodeerivate eksonite vaheldumine mittekodeerivate intronitega (Rogic et al., 2001). Eksonid võib tinglikult jaotada neljaks klassiks: geeni 5` eksonid, geeni sisemised eksonid, 3` eksonid ja introniteta geenid ehk ühe eksoniga geenid (joonis 2). Kuna enamus eukarüootseid geene sisaldavad paljusid eksoneid ja introneid, siis ka paljud geenide ennustamise programmid arvestavad ennustamisel kõigi eksonite klassidega. Osad programme on geeniennustamisalgoritmi tööpõhmõtte fokuseerinud ühele kindlale eelpool mainitud eksonite alamklassile või geenide cis-elementide (erinevad regulaatorelemendid, nt TATA-boks) ning erinevate signaalide (nt splaissaidid) ennustamisele (Zhang, 2002).

Joonis 2. Eksonite klassifikatsioon. TSS – transkriptsiooni alguskoht (***t**ranscription **s**tart **s**ite*), GT/AG – splaissaidid, Polü(A) – 3' polü(A)-saba

Rääkides geeniennustus programmidest, on oluline märkida terminite ekson ja CDS (*coding sequence*) tähendusest antud kontekstis. CDS kujutab endast vaid transleeritavat DNA järjestust, kuid mõiste ekson sisaldab endas ka mittetransleeritavat järjestust e UTR regiooni. Näiteks on UTR polü(A)-saba. Antud mõistete algne definistsioon on muutunud häguseks rääkides geeniennustusprogrammidest. Tihti räägitakse eksonitest ja CDS-idest sünonüümselt, mõeldes tegelikult CDS-e (Zhang, 2002).

Genoomilt transkribeeritakse geenid pre-mRNA molekulideks, mis seejärel läbivad keerulise posttranskriptsioonilise protsessingu ehk RNA protsessingu. Selle käigus eemaldatakse mittekodeerivad intronid ja liidetakse eksonid üheks mRNA molekuliks – toimub pre-mRNA splaissimine (Rogic et al., 2001). RNA protsessing kätkeb endas ka

pre-mRNA molekulile 5’ otsa *cap*-i ja 3’ otsa polü(A)-saba lisamist. Oluline on märkida, et eksonite ühendamisel võib olla mitu võimalust (alternatiivne splaissimine), näiteks inimese puhul arvatakse, et 35% kõikidest geenidest (Mironov et al., 1999) ja 75% mitme-eksonilistest geenidest (Garcia-Blanco et al., 2004) teevad läbi alternatiivse splaissimise, roti puhul arvatakse 20% geenide läbimist alternatiivse splaissingu protsessist (Gibbs et al., 2004). Samuti teeb universaalsete geeniennustamis-algoritmide leidmise raskemaks asjaolu, et leidub geene, mis on duplitseerunud või väga sarnase järjestusega ning pseudogeenid, mille DNA järjestus on sarnane valku kodeerivate geenide omale, kuid tegelikult funktsionaalsust ei oma (Rogic et al., 2001).

Väga keeruline küsimus on geenide regulatoorsete elementide ennustamine. Regulatoorsed elemendid omavad geeni ekspressiooni juures märkimisväärset rolli. Nende identifitseerimine on oluline mõistmaks geenide funktsiooni ja rolli rakulistes protsessides. Regulatoorsete elementide lokalisatsioon geeni suhtes pole üheselt määratav. Näiteks väga levinud cis-elemendid TATA- ja CAT-boks esinevad geeni transkriptsiooni alguskoha suhtes tavaliselt 5’ suunas, kuid geenide transkriptsiooni võimendajad ja vaigistajad võivad asuda geeni transkriptsiooni alguskohast nii ees- kui ka tagapool, samuti varieerub oluliselt viimati mainitud elementide kaugus transkriptsiooni alguskohast (Rogic et al., 2001).

2.2 Meetodid geenide avastamiseks.

Geenide identifitseerimine eksperimentaalselt on usaldusväärne, kuid katsete suur ajakulu ja kõrge hind on loonud vajaduse automaatseks geenide ennustamiseks. Isegi kui üheks päevaks suudetakse kõik geenid ühes organismis katseliselt määrata, on ikkagi oluline mõista, mis printsiibi alusel on võimalik DNA nukleotiidset järjestust, mis määrab geeni, eristada nukleotiidsest järjestusest, mis geeni ei määra (Zhang, 2002). Võib eristada kahte peamist lähenemist automaatsele geeniennustamisele - homoloogia kaudu ennustamine ja *ab initio* geeniennustamine. Paljud geeniennustus programmid kasutavad mõlemat lähenemist saavutamaks geeniennustusel maksimaalne kvaliteet.

2.2.1 Homoloogia kaudu ennustamine

Homoloogia kaudu ennustamine baseerub kahe järjestuse omavahelisel võrdlemisel. Kahe järjestuse vahelisel sarnasusel põhinevad meetodid eeldavad, et kaks geeni, mille produktid omavad ühist funktsiooni, on ka järjestuse poolest konserveerunud. Uuritavat järjestust võrreldakse juba iseloomustatud avalikus andmebaasis leiduva järjestusega. Märkimisväärne sarnasus kahe organismi järjestuse vahel annab põhjust oletada, et nad on homoloogsed, so neil on ühine evolutsiooniline päritolu. Uuritavat järjestust võidakse võrrelda genoomse DNA, valgu, EST-ide, cDNA-de või teada olevate järjestuste motiivide (regulatoorsed elemendid) vastu. Kui uuritavale järjestusele leitakse piisavalt sarnane juba annoteeritud järjestus (DNA, proteiin), siis saadakse kasutada infot, mis käib juba karakteriseeritud järjestusel leiduva geenijärjestuse või geeni poolt kodeeritava valgu funktsiooni kohta (Rogic et al., 2001) .

2.2.1.1 Ennustamine EST-ide ja cDNA baasil

EST-id ehk ekspresseeritud järjestuste märgised (***E****xpressed* ***S****equence* ***T****ag*) saadakse, kui transkriptide (mRNA) pealt sünteesitakse cDNA ja seejärel sekveneeritakse, kas cDNA 5’ või 3’ ots (enamasti siiski 3’ ots). Saadud järjestust kutsutakse ekspresseeritud järjestuse märgiseks ehk EST-iks. EST-ide pikkus jääb 200 - 600 nukleotiidi vahemikku (Nadershahi et al., 2004). Kui EST-id omavad üksteisega piisavat ülekattumist, st katavad kogu geeni, siis nende joondamise abil genoomsele DNA järjestusele on võimalik teada saada geenide asukohta ja struktuuri. Samuti annab geeniennustamisel häid tulemusi täispikkade cDNA järjestuste kasutamine (Brendel et al., 2004).

EST-ide võrdlus genoomiga annab järjestuse kohta infot vaid juhul, kui uuritavat järjestust transkribeeritakse, so ta sisaldab ekspresseeritavaid geene. EST-ide abil saadud info on mittetäielik ning annab vaid vihjeid kogu geeni struktuuri või funktsiooni kohta (Rogic et al., 2001). Kuna erinevate EST-ide ning cDNA-de saamine rakust sõltub tugevalt ekspressiooni tasemest, siis on ka enamuste liikide vastavad kollektsioonid ebatäiuslikud. Rakus kõrgelt ekspresseeritud geenid on tihti esindatud sadu kuni tuhandeid

kordi EST-ide raamatukogus, samal ajal kui geenid, mida ekspresseeritakse madalal tasemel või vähestes kudedes, võivad puududa (Burge, 2001). Samuti iseloomustab EST-e väga suur redundantsus (Mathé et al., 2002).

Tuntumad programmid, mis kasutavad geenide leidmiseks EST-ide järjestusi on näiteks EbEST (Jiang et al., 1998), TAP (Kan et al., 2001) ja Est2genome (Mott, 1997). Programm EbEST püüab lahendada kahte probleemi, mis on seotud EST-ide kasutamisega geeniennustamisel – EST-ide redundantsus ja EST-i järjestuse ebakorrektsus. Redundantsuse probleemi lahendamiseks klasterdavad nad EST-id gruppideks, mis ei tohi sisaldada üksteisega ülekattuvaid EST-e ning valivad igast sellisest grupist välja kõige informatiivsema esindaja. EST-ide genoomile asetamiseks kasutatakse Smith-Watermani algoritmi, mis võimaldab ka vigaste järjestustega EST-ide joondamist genoomsele järjestusele (lubab palju nukleotiidide mittevastavusi ehk *mismatch*'e).

Probleemi, mis on seotud EST-ide ja cDNA-de kollektsioonide ebatäiuslikkusega, püüab lahendada programmi GeneSeqer algoritm. Algoritmi loojad leiavad, et paljude tänapäeval kasutuses olevate joondamisprogrammide ebaadekvaatsus seisneb nende liiga ranges EST-i ja märklaudjärjestuse sarnasuse nõudes. Antud programm on võimeline joondama tuhandeid EST-e genoomsele DNA-le mõistliku aja jooksul lubades suhteliselt paljusid nukleotiidide mittesobivusi (ehk *gap'*e), insertsioone ja deletsioone (ehk *mismatch*'e) EST-i järjestuses märklaud (genoomse) DNA suhtes. Viimane nähtus lubab mitte samast liigist pärinevate EST-ide, see hõlmab EST-e suguluses olevate liikide duplitseeritud ja homoloogsetest geenidest, kasutamist geeniennustusel (Brendel et al., 2004).

Programmid, mis kasutavad geenide leidmiseks cDNA järjestusi, on näiteks SIM4 (Florea et al., 1998) ja Spidey (Wheelan et al., 2001). Mõlemad programmid sooritavad genoomse DNA järjestuse joonduse cDNA-de järjestuste vastu.

2.2.1.2 Genoomsel sarnasusel põhinev ennustus

Mitmed meetodid kasutavad kahe eri organismi genoomi omavahelisel sarnasusel põhinevat ennustust. Sellise lähenemise juures eeldatakse, et genoomi kodeerivad alad on

evolutsiooniliselt konserveerunumad kui mittekodeerivad regioonid. Seetõttu järeldatakse, et erinevate liikide vahelised homoloogsed piirkonnad on kodeerivad alad. Kõrgelt konserveerunud alad võivad viidata geeni regulatoorsetele elementidele või signaalidele (splaissaidid). Kahe liigivaheline konserveerumistase sõltub oluliselt antud organismide evolutsioonilisest kaugusest.

Genoom:genoom võrdlust kasutavad näiteks meetodid ExoFish ja GLASS/ROSETTA (Burge et al., 2001). ExoFish kasutab TBLASTX homoloogial põhinevat võrdlust organismi *Tetraodon nigroviridis* (*pufferfish,* teatud kerakala) ja inimese genoomijärjestuse vahel, et leida konserveerunud kodeerivaid regioone (*ecore, **e**volutionary **co**nserved **re**gion*) (Roest Crollius H et al., 2000). GLASS/ROSETTA kombineerib inimese ja hiire genoomse järjestuse globaalse joondamise dünaamilise programmeerimisega, et saada selgust õigetest geenistruktuuridest mõlemas organismis. Meetod GLASS joondab kahe liigi ortoloogsed regioonid (sünteensed alad) mõlemale genoomile. Programm ROSETTA, eeldades ortoloogsete regioonide konserveeritust, püüab leida eksoneid iseloomustavate tunnuste abil joondatud ortoloogsetest piirkondadest geene nii hiirest kui inimesest (Batzoglou et al., 2000).

2.2.1.3 Ennustamine valkude baasil

Hinnatakse, et valkude baasil on võimalik identifitseerida peaaegu 50% genoomis leiduvatest geenidest (Mathé et al., 2002). Võimalike kodeerivate alade identifitseerimiseks kasutatakse lokaalse joondamise algoritmil põhinevaid programme. Sellisel juhul transleeritakse uuritav järjestus kõigis lugemisraamides kontseptuaalseks valguks ning võrreldakse teadaolevate valgujärjestustega. Kirjeldatud põhimõtet kasutab näiteks programm BLASTX. BLASTX-stiilis sarnasusotsingut on kombineeritud globaalse joondamisega programmi GeneWise algoritmis (Birney et al., 2000). Programmi algoritm kasutab Peidetud Markovi mudeleid (HMM, *Hidden Markov Models*), et panna kokku terviklik geen BLASTX poolt leitud geeniosadest. Programmi töös sooritatakse kõige tõenäosuslikumad geeniennustamised genoomsest järjstusest ja võrreldakse siis saadud tulemusi valkude profiili-HMM-iga (*protein profile-HMM*). GeneWise programmi

kasutab ka oma geeniennustuses ENSEMBL genoomiandmebaasi projekt (Zhang et al., 2002).

2.2.2 *Ab initio* geeniennustus

2.2.2.1 *Ab initio* geeniennustuse ülevaade

Kuigi homoloogial põhinevaid meetodeid loetakse efektiivseteks, on näidanud 22. kromosoomi reannoteerimine, et vaid 50% ennustatud valkudest omavad homoloogiat seniajani teadaolevate valkudega (Dunham et al., 1999). Homoloogia kaudu geenide ennustamine on limiteeritud ka asjaoluga, et siiski on suhteliselt vähe organisme, kelle genoom on täielikult või piisaval määral sekveneeritud ja ka annoteeritud (Rogic et al., 2002). Seega on väga oluline homoloogiast sõltumatute meetodite juurutamine. *Ab initio* geeniennustus põhineb peamiselt genoomi DNA järjestusel. Oluline eeldus korrektseks ennustuseks on see, et genoomijärjestus oleks nukleotiidi tasemel täpselt määratud. Need meetodid pole otseselt sõltuvad geenide ekspressiooni tasemest ja suudavad efektiivsemalt ennustada harva ekspresseeruvaid geene (Burge et al., 2001). *Ab initio* geeniennustus loob ka üldisi printsiipe, kuidas geenid on genoomile organiseeritud. Suureks puuduseks kaasajal olemasolevate *ab initio* geeniennustus-programmide juures on see, et nad on väga sensitiivsed (ennustavad palju geene), kuid nende spetsiifilisus on madal (ennustavad geene, mida reaalselt ei eksisteeri). Seepärast on hakatud kasutama kahe lähenemise, homoloogia kaudu ja *ab initio* geeniennustamine, kombinatsiooni. Näiteks programm SGP2, mis kasutab peale *ab initio* geeniennustamise ka TBLASTX otsingut hiire ja inimese genoomide vahel, et tagada spetsiifiline ja sensitiivne geenide leidmine (Parra et al., 2003).

2.2.2.2 Tunnused, mis aitavad detekteerida geene

Peamiselt kasutatakse kahte erinevat suurust geenide leidmiseks genoomsest järjestusest. Esimeseks suuruseks on *content sensorid*. Eristatakse väliseid ja sisemisi

content sensoreid. Esimene baseerub uuritava genoomse DNA piirkondade sarnasuse otsimisel teadaoleva(te) järjestuse(te)ga. Välise *content sensori* mõõdul põhinevad eelpool käsitletud kahe järjestuse vahelisel homoloogial põhinevad meetodid. Sisemine *content sensor* (nimetatakse ka kodeerivaks statistikuks, *coding statistic*) jagab analüüsitava järjestuse kaheks – regioonid, mis kodeerivad valke ja, mida transleeritakse ning mitte-transleeritavad (intra- ja intergeensed alad). Kodeerivad statistikud on funktsioonid, mis arvutavad igale aknale uuritavas järjestuses väärtuse, mis peegeldab antud ala tõenäosust olla kodeerivaks regiooniks. Kodeerivaid statistikuid on kasutatud juba üle kümne aasta (Fickett et al., 1993). Tuntuimaks ja ka tõhusaimaks statistikuks loetakse spetsiifiliste heksameeride esinemist teatud lugemisraamides. Arvatakse, et lugemisraami spetsiifilised heksameerid kätkevad endas infot koodoni eelistuse (*codon-bias*), koodon-koodon korrelatsioonide ja splaissaitide eelistuse kohta. Just viimane statistik ongi üks olulisemaid ekson-intron struktuuri identifitseerijaid (Zhang et al., 2002). Heksameeride mõõtu kasutavad oma algoritmides sellised programmid nagu GeneMark.hmm (Lukashin et al., 1998), Genescan (Burge 1997) ja HMMGene (Krogh 1997). Olulised statistikud on ka GC sisalduse protsent järjestuses (intronid on AT-nukleotiidide rikkamad), erinevate koodonite kasutamise tõenäosust peegeldav mõõt (*codon usage measure*), avatud lugemisraami mõõt (näitab kas lugemisraamis esineb translatsiooni stopkoodon), 20 aminohappe ja stopkoodoni esinemise tõenäousust väljendav mõõt, iga nukleotiidi esinemise tõenäousust igas koodoni positsioonis näitav kompositsiooni mõõt jm suurused (Fickett et al., 1992).

Teine laialt kasutatav suurus *signal sensor,* püüab detekteerida geenile spetsiifilisi (funktsionaalseid) elemente. Signaalid (nt cis-elemendid) on mõne nukleotiidi pikkused järjestused, mis on äratuntavad raku masinavärgi poolt ning, mis initseerivad kindlaid protsesse rakus. Sellisteks signaalideks on näiteks promooteri piirkonnas leiduvad regulatoorsed elemendid, splaissaidid, transkriptsiooni start- ja stopkoodonid, polü(A) signaalid, translatsiooni initsiatsiooni koodon jt. Kuna DNA järjestuse signaalid on degeneratiivsed ja ebaspetsiifilised, siis nad sisaldavad vähe informatsiooni, sest peaaegu võimatu on eristada õigeid signaale nendest, mis pole funktsionaalsed. Seepärast pole ka ainult signaalide detekteerimine piisav geeni identifitseerimiseks. Sellest tulenevalt on välja

töötatud algoritmid, mis kombineerivad nii signaalide kui ka kodeerivate statistikute leidmise. Olenemata sellest ei suuda siiski enamus programme veel täielikult ennustada regulaatorelemente ja polü(A) signaale, samuti ei eristata 5´ ja 3´ mittetransleeritud alasid, ei arvesta alternatiivsete splaissvariantidega ning ei detekteerita ülekattuvaid ja põimunud geene (Rogic et al., 2001). Tuntumad programmid, mis kasutavad nii signaalide leidmist kui ka kodeerivaid statistikuid on näiteks GeneParser3 (Snyder et al., 1995), Procrustes (Gelfand et al., 1996) ja AAT (Huang et al., 1997).

Signaalide ja kodeerivate statistikute leidmist teostavate algoritmide treenimiseks kasutatakse andmeid (*training set*), mis sisaldavad geene teadaolevate signaalidega ja statistikutega. Selliseid andmeid nimetatakse ka positiivseteks andmeteks. Signaalid ehk konsensusjärjestused leitakse teadaolevate funktsionaalselt seotud järjestuste mitme järjestuse joondamisel või positsioonilisi kaalumaatrikseid kasutades (***p****ositional* ***w****eight* ***m****atrices,* PWM). Positsioonilised kaalumaatriksid on mudelid, mis vaatavad iga nukleotiidi esinemise tõenäosust signaalis (Mathé et al., 2002). Treenimiseks kasutatakse ka negatiivseid andmeid, mis sisaldavad objekte, mida kindlasti ei soovita leida. Viimasteks on näiteks pseudogeenid ja juhuslikud järjestused, mis ei sisalda geene (Zhang et al., 2002). Geenide identifitseerimiseks konstrueeritakse leitud signaalide abil kokku ühe geeni struktuur (Rogic et al., 2001).

2.2.2.3 Signaalide leidmine *ab initio* geeniennustuses

Ab initio geeniennustuse võib jagada olulisemateks etappideks, seejuures iga etapi läbimiseks on erinevad algoritmid, mis olenevalt ennustatavast geeni osast või suurusest panevad rõhku erinevatele momentidele. Enamus tänapäeval olemasolevatest programmidest on fokuseerunud eksonite ennustamisele ja seejärel nende assambleerimisele geeniks.

Geenistruktuuri kokku panemisel kasutatakse, kas dünaamilist programmeerimist (DP), kus erinevatele ennustatud geeniosadele antakse skoorid, mis peegeldavad antud järjestuse olulisust olla ennustatava geeni osaks, nt programm Fgenes (Solovyev et al., 1995) või tõenäosuslikke mudeleid (näiteks HMM, MM, PWM, logistiline regressioon),

kus erinevatele geeniosadele antakse tõenäosuslikud skoorid. Viimast lähenemist kasutavad nt programmid Genscan (Burge et al., 1997) ja HMMgene (Krogh, 1997). Kuna dünaamilisel programmeerimisel ei kasutata tõenäosuslikke skoore, siis on vaja enne skooride andmist välja selgitada sobilik arvuline väärtus igale võimalikule situatsioonile (üks võimalik situatsioonide komplekt on näiteks kombinatsioon võimalikest omavahelistest paigutustest detekteeritud kolme eksoni ja kahe introni vahel). Optimaalse skooride kaalu välja töötamine on keeruline probleem. Kirjeldatud intsidenti on viimasel ajal kergendanud tõenäosuslike mudelite kasutamine (Zhang, 2002). HMM-mudeli korral toimub üleminek (situatsioon) erinevate seisundite vahel. Seisundid on erinevad geenidega assotseeruvad järjestused (nagu promooterid, mittetransleeritavad regioonid, intergeensed regioonid, polü(A) signaal, intronid, eksonid). Viimaseid on võimalik lisada jooksvalt antud mudelisse juurde. Transitsioon erinevate seisundite vahel lubab ennustada ka osalisi geene, introniteta geene, geene mõlemal ahelal. Mõlemalt ahelalt samaaegselt geenide ennustamine on oluline, kuna sel viisil on võimalik vältida erinevatel ahelatel esinevate kattuvate geenide ennustamist kaheks erinevaks geeniks. Mõlemalt ahelalt simultaanselt geenide ennustamise võimaluse võttis esimesena HMM-mudelite töös kasutusse programm Genscan, mida kasutab geenide ennustamiseks ja assambleerimiseks ka ENSEMBL genoomiandmebaasi projekt (Burge et al., 1997).

Tuntumad geeniennustus-programmid, mis baseeruvad nii *ab initio* kui ka homoloogia otsingutel, on toodud tabelis 1.

Tabel 1. Tuntumad homoloogial ja *ab initio* geeniennustusel baseeruvad geeniennustusprogrammid.

MEETOD	PROGRAMM	KASUTATAVAD ANDMED	TÖÖPÕHIMÕTE
Homoloogial baseeruv ennustus	EbEST	EST	EST-ide klasterdamine, Smith-Waterman algoritm
	TAP	EST	BLAST
	Est2genome	EST, cDNA	modifitseeritud Smith-Waterman ja Needleman-Wunch algoritm
	GeneSeqer	EST, cDNA	joondamiseks suffiks-tabelid assambleerimiseks HMM
	SIM4	cDNA, mRNA	BLAST
	Spidey	cDNA, mRNA	BLAST
	ExoFish	genoom/genoom	BLAST
	GLASS/ROSETTA	genoom/genoom	DP
	GeneWise	valk, HMM-profiil	DP
Ab initio geeniennustus	SGP2	kahe organismi genoomi andmed	HMM, BLAST
	GeneMark.hmm	kodeerivad statistikud, signaalsed sensorid	HMM, DP
	Genscan	kodeerivad statistikud, signaalsed sensorid	HMM
	HMMgene	kodeerivad statistikud, signaalsed sensorid	HMM
	GeneParser3	kodeerivad statistikud, signaalsed sensorid	DP
	AAT	signaalid, cDNA	BLAST, DP
	Fgenes	kodeerivad statistikud, signaalsed sensorid	DP

DP – dünaamiline programmeerimine; HMM – Peidetud Markovi mudelid

2.3 Geeniennustust teostavate programmide ja meetodite täpsus

Kuigi käesoleval ajal on palju töötatud erinevate geeniennustus-algoritmide kallal, pole siiski suudetud välja mõelda meetodit, mis 100 %-lise täpsusega ennustaks kõik antud organismis olevate geenide õiged struktuurid. Keeruline on efektiivse geeniennustusprogrammi loomine suurte genoomidega organismide jaoks, kus kodeeriva osa suhe mittekodeerivasse järjestusse on võrreldes väiksemate genoomidega madal, lisaks paljudele proteiine kodeerivatele geenidele leidub genoomis veel rohkesti RNA-geene, pseudogeene, osalisi geene (*partial genes*), mis omavad küll järjestuse sarnasust vastava cDNA, EST-i või valgu järjestusega, kuid ei vasta täielikult teistele geene määravatele kriteeriumitele (Collins et al., 2003).

Programmide töö korrektsuse hindamiseks võetakse kasutusele suurused spetsiifilisus (Sp, *specificity*) ja sensitiivsus ehk tundlikkus (Sn, sensitivity). Sensitiivsus on suurus, mis näitab algoritmi võimet leida objekte, mis on tõesti olemas ehk 'õigete' positiivsete leidmise võimet. Spetsiifilisus on suurus, mis näitab algoritmi ettevaatlikust 'vale' positiivsete leidmisel, st mida suurem on Sp väärtus, seda vähem leiab programm geene, mis reaalselt antud kohas genoomis ei asu (olematuid geene). Spetsiifilisust on kutsutud ka "valehäire määraks" (Baldi et al., 2001). Nii sensitiivsus kui spetsiifilisus on tõenäosuslikud suurused, mille väärtused kõiguvad vahemikus [0,1]. Mida enam lähenevad suurused väärtusele üks, seda täpsemad nad on.

$$Sp = \frac{TP}{TP + FP} \qquad Sn = \frac{TP}{TP + FN}$$

- Õiged positiivsed (TP, *true positives*) – ennustatud objektid, mis reaalselt asuvad ennustatud kohas
- Valed negatiivsed (FN, *false negatives*) – objektid, mida ei ennustatud antud regiooni, kuigi reaalselt eksisteerivad selles regioonis.
- Valed positiivsed (FP, *false positives)* – objektid, mis ennustati teatud regiooni, kuigi nad seal reaalselt ei asu.

- Õiged negatiivsed (TN, *true negatives*) – objektid, mille olemasolu ennustus-algoritm ei ennusta ning reaalselt pole ka need objektid ennustus-algorimti sihtmärgiks.

Programmide kvaliteedi hindamine võib toimuda erinevatel tasemetel. Programmi kvaliteeti võib hinnata nukleotiidi, eksoni, terve geeni, erinevate signaalide ja kodeerivate statistikute tasemel. Vastavalt on siis objektideks, kas nukleotiid, ekson, geen, signaal või statistik. Enam kasutatakse kvaliteedi hinnangutes nukleotiidi ja eksoni taset, sest terve geeni tasemel hindamine ei ole geeniennustus-programmi kvaliteedi hindamisel usaldusväärne (Rogic et al., 2001).

Jõudmaks selgusele, milline programm on parem kui teine, viiakse läbi erinevate *ab initio* geeniennustusprogrammide testimisi. Programmide testimisel on oluline, et ei kasutataks test-andmetena samu andmeid, millega programmi treeniti. Paljude sõltumatute testide tulemused on lugenud kvaliteetseteks programmideks Genscan ja HMMgene geeniennustus-programmid. Usaldusväärseks ja põhjendatuks loetakse nende programmide töös realiseeritavate tõenäosuslike mudelite erinevate seisundite vahelistele transitsioonidele määratud skoore. Genscan sensitiivsuseks nukleotiidi tasemel on arvutatud 0.95 ja spetsiifilisuseks nukleotiidi tasemel 0.98 nukleotiidi tasemel ning HMMgene vastavad suurused on 0.93 ja 0.99 (Zhang CT, Zhang R., 2002).

2.4 Geenide annoteerimise süsteem suuremates annotatsioonikeskustes.

ENSEMBL-i geeniennustuse süsteem võimaldab eukarüootsete organismide genoomide kiiret automaatset annotatsiooni. Geenide annoteerimisel kombineeritakse omavahel nii homoloogial põhinevaid kui ka *ab initio* geeniennustusmeetodeid. ENSEMBL-i geeniennustuse süsteem põhineb eelkõige valkude ja cDNA järjestuse homoloogsusel genoomi järjestusega. Valkude ja cDNA kasutamine paralleelselt võimaldab detekteerida mitte-transleeritavaid järjestusi geenide struktuuris. ENSEMBL-i geeniennsutuse süsteem hõlmab endas mitmeid järjestikuseid etappe. Esimese etapina asetatakse liigi-spetsiifilised valgud genoomile programmi PMATCH (Durbin, unpublished)

abil. PMATCH on programm, mis joondab valgu järjestused robustselt (ei kasuta splais-saitide leidmise mudeleid), kas teise valgu või DNA järjestusega. Kuna ta on suhteliselt kiire, siis teda kasutatakse ligikaudseks transkriptide lokalisatsiooni määramiseks genoomis, et vähendada suurte genoomide edasistesse (joondamis-) etappidesse minevat genoomset järjestust. Lõpliku valgu joondamise viib läbi programm Genewise, mis erinevalt eelmisest programmist kasutab splaissaitide ja raaminihke leidmise mudeleid, kuid on programmist PMATCH oluliselt aeglasem. Genewise programmi suure ajakulu tõttu teostatakse enne selle programmi kasutamist veel individuaalsete eksonite, mis on leitud genoomist PMATCH programmi poolt, joondamist BLAST-iga valgu järjestuse vastu. BLAST-iga leitud homoloogseid piirkondi suurendatakse mõlemalt poolt 200 bp võrra, et saada splaissaitidega külgnevat ala. Sellised ühe-eksonilised järjestused liidetakse seejärel üheks minijärjestuseks. Sellise protseduuriga vähendatakse nt 50 kb geeni pikkus 2 kb peale. Seejärel kasutatakse eelmise kolme etapi tulemusena leidmata jäänud transkriptide otsimine teistest liikidest pärinevate valkudega BLAST-i joonduse ja Genewise programmi abil. Paralleelselt valkude joondamisega genoomile joondatakse genoomile ka cDNA järjestused programmi Exonerate (Slater, unpublished) abil. Valgu ja cDNA homoloogial põhinevate meetodite abil saadud transkriptidest moodustatakse kokku üks konsensusjärjestus, mis sisaldab 5’ mittetransleeritavat ala, kodeerivat regiooni ning 3’ UTR-i. Viimaks kasutatakse programmi GeneBuilder, et klasterdada kõik genereeritud transkriptid ülekattumise alusel (Curwen et al., 2004).

Viimasel ajal on ENSEMBL geeniennustuse süsteemis kasutatud vähem *ab initio* geeniennustust, sest vastavad pogrammid vähe spetsiifilised ja väga sensitiivsed. Käesoleval ajal pakub annotatsioonikeskus ENSEMBL *ab initio* geeniennustusel baseeruvaid programmi Genscan poolt ennustatud geene. ENSEMBL ennustab ka geene, mis baseeruvad EST-ide homoloogial. Kuna EST-ide järjetused sisaldavad palju vigu ning on oma lühikese pikkuse tõttu ebaspetsiifilised, siis ei loeta nende baasil saadud geene väga usaldusväärseteks ning sarnaselt *ab initio* geeniennustusegagi neid ENSEMBL geeniennustuse süsteemi kaasata, vaid pakutakse sel meetodil ennustatud geene eraldi (Curwen et al., 2004).

Geenide annotatsiooniandmete hoidmiseks on NCBI-s kaks peamist andmebaasi – Refseq ja LocusLink (Pruitt et al., 2000). Esimene pakub mRNA-del baseeruvaid geeniennustuse andmeid, nn Refseq geene (***Ref****erence* ***seq****uence*), mida iseloomustab vähene redundantsus. Teine andmebaas sisaldab andmeid, mis kirjeldavad üheselt ära antud lookuse (*locus-to-sequence*). Lookust kirjeldavateks andmeteks on järjestus, standard-nomenklatuur, geeni ning talle vastava valgu kirjeldused. Viimaste andmete saamiseks kasutatakse erinevaid NCBI andmebaase, mille hulka kuulub ka Refseq (Pruitt et al., 2001).

Manuaalset annotatsiooni läbiviiv *The Welcome Trust Sanger* Instituudi alla kuuluv HAVANA grupp kasutab geenide ennustamiseks mitme-etapilist süsteemi. Genoomset järjestust analüüsitakse kloonide abil ning teostatakse homoloogia otsinguid DNA ja valgu andmebaaside vastu, samuti kasutatakse tulemuste kvaliteedi hindamiseks *ab initio* geenieennustuse programme (Genscan, Fgenes). Uudsete geenide leidmiseks kasutatakse teistes organismides sisalduvate geenidega seotud andmeid (cDNA, evolutsiooniliselt konserveerunud regioonid) (*http://vega.sanger.ac.uk/*).

UCSC genoomi brauseris pakutavad annotatsiooniandmetest on pooled saadud välistest allikatest ning pool on arvutatud UCSC poolt. Näiteks kasutatakse NCBI GenBank andmebaasist võetud mRNA ja EST-ide järjestusi BLASTi-sarnase joondamise (BLAT) abil annotatsiooniandmete genereerimiseks. Välistest allikatest kasutavad näiteks geeniennustus programmidega Fgenesh++, Genie, Genscan genereeritud geeni-ennustuse andmeid ning ENSEMBL geeni-ennustuse süsteemi kaudu genereeritud andmeid (Karolchik et al., 2003).

3. PCR parameetrite valimine

Polümeraasi ahelreaktsioon (***p****olymerase* ***c****hain* ***r****eaction,* PCR) võimaldab huvipakkuva regiooni (*target*) amplifitseerimist DNA järjestuselt (*template*). Konventsionaalses PCR-i protsessis kasutatakse kahte oligonukleotiidi ehk praimerit, mis seonduvad erinevatele ahelatele. Reaktsioon baseerub tsüklite kordamises mitmeid kordi.

Igas tsüklis toimub uuritava DNA denaturatsioon, praimerite seondumine komplementaarsetesse kohtadesse uuritavas DNA-s ja praimerite ekstensioon (Rubin et al., 1996).

PCR-i praimerite disain on essentsiaalne ja aeganõudev samm paljudes eksperimentides. Praimerid, mis on disainitud kindlale sihtmärkjärjestusele võivad reakstioonis anda ootamatuid tulemusi. Tihti amplifitseerub ka mittesihtmärkjärjestusi. Viimast just olukorras, kus uuritav DNA on suur ja keeruline (Rubin et al., 1996). Praimerite disainimisel tuleb arvestada paljude kriteeriumitega. Korrektselt töötav praimer peab moodustama stabiilse dupleksi õige sihtmärk DNA-ga, ei tohi seostuda iseendaga, moodustada teiste praimeritega ega vale kohaga sihtmärkjärjestuses interaktsioone. PCR-i kvaliteeti püütakse tagada kvaliteetsete praimerite disainiga (Li et al., 1997).

3.1 PCR-i kasutamine

PCR-i kasutatakse palju mutatsioone ja DNA markereid (näiteks ühenukleotiidiline polümorfism, lühikeste järjestuste märgised STS, mikrosatelliidid) sisaldavate genoomsete DNA regioonide üles-amplifitseerimiseks. Üles-amplifitseeritud piirkondi (PCR produkte) saab kasutada mutatsioonide resekveneerimisel (uute mutatsioonide avastamine), mutatsioonanalüüsis (insertsioonid, deletsioonid jt) ja genotüpiseerimises. Erinevate geneetiliste haiguste uurimise tasemel on oluline, et oleks võimalik genoomsest DNA-st üles amplifitseerida kodeerivaid piirkondi, samuti on oluline indiviidide vaheliste polümorfismide uurimine (Albertson et al., 2003).

Geenide uurimiseks vajaliku kodeeriva piirkonna spetsiifiline üles-amplifitseerimine genoomselt DNA-lt on keeruline probleem. Geenide amplifitseerimiseks kogu genoomist järgitakse praimerite disainimisel üldtuntud praimerite disainimise põhimõtteid. Lisaks sellele välditakse praimerite disianimist genoomis leiduvatesse korduvatesse aladesse ning jälgitakse, et disainitud praimerid ei omaks genoomis sekundaarseid seondumiskohti Genoomselt DNA-lt eelnevalt üles-amplifitseeritud kodeerivaid piirkondi on võimalik

analüüsida kasutades mitmeid erinevaid genotüpiseerimise ja mutatsioonide detekteerimise tehnoloogiad. Viimasteks on näiteks ühenukleotiidiliste mutatsioonide genotüpiseerimiseks alleel-spetsiifiline PCR ehk ARMS (Montanaro et al., 2002, Brightwell et al., 2002), reaal-aja PCR ehk RT-PCR (Wilhelm et al., 2003, Syvanen AC., 2001, Mhlanga et al., 2001) mutatsioonide detekteerimiseks ja genotüpiseerimiseks, translokatsioonide ja deletsioonide uurimiseks pööratud PCR ehk IPCR (Williams et al., 2002) jpt.

3.2 Sulamistemperatuur ja GC nukleotiidide sisaldus.

Sulamistemperatuur (Tm, *melting temperature*) iseloomustab praimeri ja tema sihtmärkjärjestuse dupleksi stabiilsust. Sulamistemperatuur on temperatuur, mille juures pooled nukleotiidide molekulid (praimerid ja nende sihtmärkjärjestused) on kaheahelalised ja pooled üheahelalised. Sulamistemperatuur on oluline, kuna määrab praimerite seondumistemperatuuri PCR-i reaktsioonis (Ta, *annealing temperature*). Kui Ta võtta liiga madal, siis amplifitseeritakse üles ka mittespetsiifilisi DNA fragmente, vastupidisel korral saadakse mitte rahuldavas koguses soovitud PCR-i produkti (Rychlik et al., 1990). Kuigi palju on uuritud Tm-i ja Ta vahelisi seoseid, on täpne Tm-i ja Ta üksteisest sõltuvus ebaselge. Seondumistemperatuuri arvutamiseks on analüüsitud erinevates katsetes kasutatud praimereid ja nende produkte ning jõutud järgmise seondumistemperatuuri arvutava valemini:

$$T_a^{OPT} = 0.3T_m^{praimer} + 0.7T_m^{produkt} - 14.9 \quad (i)$$

Valemis (i) $T_m^{praimer}$ - praimeri sulamistemperatuur, $T_m^{produkt}$ – PCR produkti sulamistemperatuur.

Kui praimeri sulamistemperatuur arvutatakse enamasti *Nearest-Neighbor* mudelit (vt allpool) kasutades, siis PCR produkti sulamistemperatuuri arvutamiseks sama valemit kasutada ei saa (valem ei tööta pikkade DNA fragmentide korral), vaid kasutatakse modifitseeritud T_m (*nearest-neighbor* meetod) arvutamise valemit (Baldino et al., 1989):

$$T_m^{produkt} = 0.41(\%G + \%C) + 16.6*\log[K^+] - 675/l \text{ (ii)}$$

Valemis (ii) %G ja %C on vastavate nukleotiidide esinemise osakaal protsentides antud PCR-i produktis, K^+-ioonide molaarsus (mM), log – logaritm alusel 10, l on produkti pikkus nukleotiidides.

Oluline on, et kahe praimeri (pluss ja miinus ahela praimeri, *reverse, forward primers*) sulamistemperatuuri oleks sarnane, kuna PCR-i tingimused on ühes ja samas füüsilises katses samad.

Erinevaid lähenemisi sulamistemperatuuri arvutamiseks on mitmeid. Kõige lihtsam ja seega ka üldisem viis Tm-i arvutamiseks on järgmine valem:

$$T_m^{praimer} = 4(*\#G + \#C) + 2*(\#A + \#T) \text{ (iii)}$$

Valemis (iii) #C, #G, #A, #T on vastavate nukleotiidide esinemisarvud disainitavas praimeris, T_m ühikuks on ^oC.

Antud valem on empiiriliselt tuletatud uurides paljusid praimereid. Arvestades oma primitiivsust töötab ta suhteliselt hästi praimerite korral, mille pikkus jääb nukleotiidide vahemikku [16-28] (Kämpke et al., 2001).

Sulamistemperatuuri mõjutavad rohkem G ja C nukleotiidid, kuna nende nukleotiidide vahel on kolm vesiniksidet samal ajal kui A ja T nukleotiidide vahel on kaks vesiniksidet. Praimerite G/C sisaldus tuleb valida nii, et taotletud spetsiifiline seondumine oleks stabiilne, kuid samas saab toimuda ka efektiivselt vajalik praimeri 'sulamine' sihtmärkjärjestuselt.

Täpsema sulamistemperatuuri arvutamiseks lühikestele oligotele (praimeritele) on laialdaselt kasutusel *Nearest-Neighbor* meetod (Borer et al., 1974).

$$T_m^{praimer} = \Delta H/[\Delta S + R*\ln(c/4)] - 273.15^\circ C + 16.6*\log_{10}[K^+] \text{ (iv)}$$

Valemis (iv) ΔH ja ΔS on vastavalt heeliksi moodustamise entalpia ja entroopia, R - universaalne gaasikonstant (R=8.31J/mol), c – oligonukleotiidide totaalne molaarne konsentratsioon reaktsioonisegus.

Sulamistemperatuuri arvutamise mudel (iv) eeldab, et energia, mida on vaja paardunud ahelas ühe aluspaari vahelise vesiniksideme lõhkumiseks sõltub kõrval olevast aluspaarist. Arvutatakse välja dimeerdupleksite (nt AC/TG) termodünaamilised suurused (ΔH, ΔS). Näiteks järjestuse p = GGAT jaoks arvutatakse entalpia järgmise valemiga: ΔH (GGAT) = ΔH(GG) + ΔH(GA) + ΔH(AT). Termodünaamilised suurused iga dimeerdupleksi jaoks on välja arvutatud mitme grupi poolt, üks esimesi sellel alal oli aga Breslaueri grupp (tabel 2) 1986ndal aastal (Owczarzy et al., 1997).

Tabel 2. *Nearest-Neighbor* termodünaamika

dimeer dupleks	ΔH (kcal/mol)	ΔS (cal/K mol)
AA/TT	9.1	24.0
AA/TA	8.6	23.9
TA/AT	6.0	16.9
CA/GT	5.8	12.9
GT/CA	6.5	17.3
CT/GA	7.8	20.8
GA/CT	5.6	13.5
CG/GC	11.9	27.8
GC/CG	11.1	26.7
GG/CC	11.0	26.6

3.3 Praimeri pikkus

Praimeri pikkus püütakse teha minimaalne, et vähendada PCR reaktsiooni hinda. Pikkuse määrab ära minimaalne unikaalse ala pikkus sihtmärkjärjestuses, G/C sisaldus, Tm, disainitava praimeri 5’ ja 3’ ots, praimeri sisene nukleotiidne koostis. Konventsionaalse

PCR-i jaoks disainitakse praimerid enamasti pikkusevahemikus [18,30] ning optimaalseks praimeri pikkkuseks loetakse vahemikku [20,25] (Kamel et al., 2003).

Oluline on märkida, et praimeri pikkus, GC nukleotiidide sisaldus ja sulamistemperatuur on üksteisest tihedalt sõltuvad.

3.4 Produkti pikkus

Parimaid tulemusi annab konventsionaalne PCR kui produkti pikkus on alla mõne tuhande aluspaari. PCR tehnoloogiat on modifitseeritud pikkade fragmentide amplifitseerimiseks, seda eelkõige reaktsioonisegusse lisatavate komponentide osas. Erinevad *long*-PCR tehnoloogiad võimaldavad amplifitseerida suhteliselt pikki (kuni 20 kb pikkuseid) nukleotiidseid järjestusi ühe etapina. Näiteks XL-PCR (*long* PCR) abil on võimalik paljundada kogu inimese mtDNA (mitokondriaalne DNA, pikkus ca 16,500 bp) järjestus ühe etapina ja võimaldades seeläbi detekteerida järjestuses aset leidnud mutatsioone või insertsioone järjestuse kogupikkuse varieeruvuse alusel (Kopsidas et al., 2000). *Long* RT-PCR (*long-reverse transcription PCR*) võimaldab amplifitseerida täispikki mRNA järjestusi ühe etapina, mis kindlustab selle, et üks kindel cDNA on saadud ühest kindlast mRNA-st (Hu et al., 1998).

3.5 Praimeri 3’ otsa nukleotiidide sisaldus

On näidatud, et G/C nukleotiidide esinemine praimeri 3’ otsas tõstab praimer/sihtmärkjärjestus dupleksi stabiilsust ja kindlustab efektiivset praimeri ekstensiooni (Sheffield et al., 1989). Osad teadlased väidavad, et väga kleepuv praimeri 3’ ots võib põhjustada praimerite seondumist sekundaarsetele seostumiskohtadele ning soovitavad disainida praimeri nii, et kleepuvam ots oleks praimeri 5’ ots ning 3’ ots oleks A-T rikkam (Andreson, unpublished). Uuritud on ka PCR-i efektiivsust, kui praimeri 3’ otsa on disainitud sihtmärkjärjestusega mittepaarduv nukleotiid. Katsed on näidanud, et üksikud

T/T, T/C või T/G mittepaardumised praimeri 3’ otsa ja sihtmärkjärjestuse vahel ei mõjuta PCR-i kvaliteeti, kuid G/A mittepaardumise korral pole võimalik soovitud PCR-i produkti genereerida. Vältimaks sekundaarseid hübridisatsioone, ei soovitata praimeri 3’ otsa disainida mono-nukleotiidseid kordusi (Simsek et al., 2000).

3.6 Dimeeride ja sekundaarstruktuuride moodustumine

Et tagada PCR-i kvaliteeti, on oluline, et disainitud praimer seostuks vaid ühe spetsiifilise järjestuse ühte kindlasse kohta. PCR-i kvaliteedi all pean silmas kahte asjaolu – reaktsiooni käigus tekiks ainult ühte oodatud produkti ning seda produkti genereeruks piisavas koguses. Tuleb kontrollida iga praimeri (+ ahela praimer A, - ahela praimer B) seondumist iseendaga (*self annealing*), so dimeeride moodustumist nii sama praimeri (AA, BB) kui ka teise praimeriga (AB) ja praimeri 3’ otsa seostumist iseendaga – sekundaarstruktuuride moodustumist (*self-end annealing*) (Hillier et al., 1991).

Kasutusel on erinevaid lähenemisi, kuidas hinnata praimeri seondumist iseendaga, teise praimeriga samast või teisest praimeripaarist ning praimeri 3’ otsa tagasipöördumist iseendale. Kahe sama praimeri (AA, BB) ja praimeripaari kuuluvate praimerite vahelise (AB) seondumise määra arvutamine toimub analoogse põhimõttega. Kasutusele on võetud numbrilist väärtust tagastav funktsiooni S, mis väljendab kahe praimeri järjestuse seondumise ulatust. Kui meil on kaks praimerit, millest esimene on x = ($x_{1,...,}$ x_n) ja teine y = ($y_{1,...}y_m$), kus n ja m on vastavate praimerite pikkused, siis saame me moodustada nende praimerite vahel k = (-n + 1, ..., m –1) ülekattuvat dupleksit. Seega võimalik kahte järjestust üksteisega joondada k = (n + m - 1) erinevale viisil. Kui tegu on kahe sama praimeriga (n = m), siis lihtsustades valemit k saame, et k = (2n – 1). Igale võimalikule dupleksile arvutatakse funktsiooni S(x, y) väärtus (v).

$$S(x, y) = \max_{k=(n+m-1)} \sum_{i=1}^{n} s(x_i, y_{i+k}) \qquad (v)$$

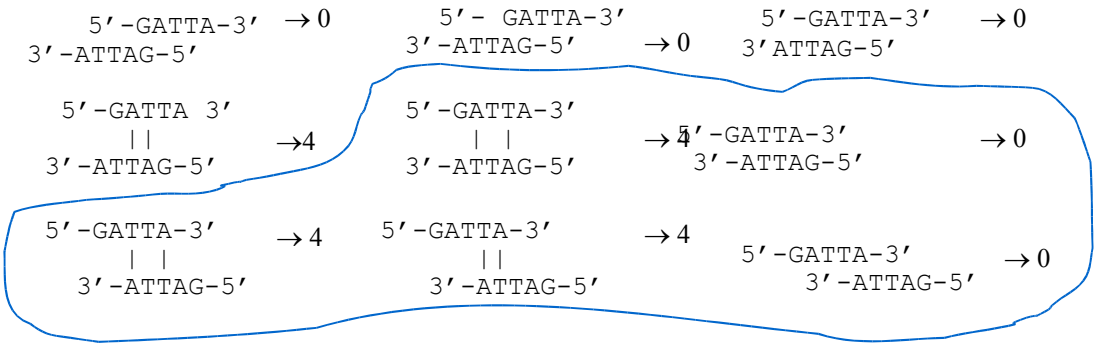
Valemis (v) i on üks dupleks kõikidest võimalikest moodustatud dupleksitest.

Funktsiooni olemus seisneb igas dupleksi positsioonis olevale aluspaarile skoori määramises valemi $s(x_i, y_j)$ (vi) alusel.

$$S(x_i, y_j) = \begin{cases} 2, \text{ kui } \{x_i, y_j\} = \{ A, T \} \\ 4, \text{ kui } \{x_i, y_j\} = \{ C, G \} \\ 0, \text{ kõik teised esinevad situatsioonid} \end{cases} \qquad (vi)$$

Valemis (vi) on i vastava dupleksi esimese praimeri positsioon, mis vastab teise praimeri j-nda positsiooniga. Sama praimeri seostumist väljendava S(x, y) funktsiooni väärtuse arvutamine on toodud näites 1.

NÄIDE 1. Praimer p = GATTA, pikkus n = 5, k = (2*5-1) = 9 ülekattuvat joondust. Toodud on kõigi võimalike ülekattuvate dupleksite kombinatsioonid järjekorras k = -4, ..., 4 ning neile arvutatud dupleksite summaarsed $s(x_i, y_j)$ väärtused.



Praimeri p iseendaga seondumise määr on 4, sest S(GATTA, ATTAG) = max {0,0,0,4,4,0,4,4,0} = 4.

Kirjeldatud joonduse abil on võimalik ka määrata praimeri 3’ otsa tagasipöördumise määra. Kasutatakse valemit (vi), kuid vaadatakse vaid selliseid joondusi kõigi võimalike joonduste hulgast, kus 3’ ots kuulub ülekattuvasse alasse. Näite 1 põhjal praimeri p=GATTA sellisteks joondusteks on joondused k = (0,..., 4,) mis on ümbritsetud joonega. Skoori arvutamiseks vaadatakse ainult praimeri 3’ otsas olevat alamjärjestust, mis sisaldab dupleksi moodustunud järjestuste vahel järjestikku moodustunud vesiniksidemeid. Näite 1 korral saame seega arvestada k = 2 ja k = 3 joondustega. Joondus k = 2 annab summaarseks $s(x_i, y_j)$ väärtuseks 2 ja k = 3 annab vastavaks väärtuseks 2+2 ehk 4. Seega praimeri p iseenda 3’ otsaga seondumise skooriks on $S = \max\{2, 4\} = 4$. (Kämpke et al., 2000).

3.7 Rist-hübridiseerimise vältimine

Saavutamaks ühe unikaalse produkti genereerimist PCR-il tuleb lisaks eelpool käsitletud punktidele kontrollida ka disainitud praimerite sekundaarsete seondumiskohtade olemasolu sihtmärkjärjestust sisaldavas järjestuses. Viimase ülesande lahendamine on eriti oluline ja keeruline juhul, kui soovitakse siht-märkjärjestust üles-amplifitseerida genoomselt järjestuselt. Manuaalselt sellise probleemi lahendamine pole võiamlik.

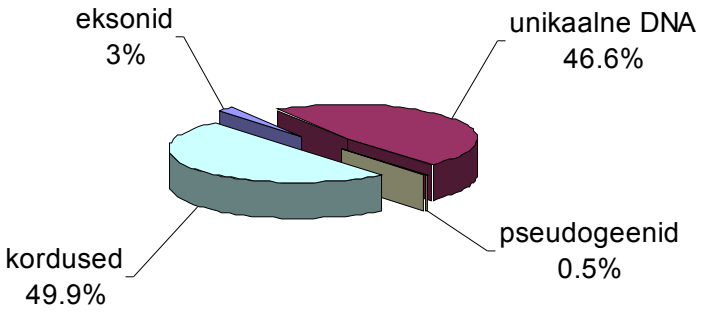
PCR- praimerite sekundaarsete seostumiskohtade vältimiseks on levinud kahe lisa sammu läbi viimine praimeri disaini protsessis. Esimene etapp on järjestuse, millele praimereid disainitakse maskeerimine kohtadelt, mis sisaldavad kordusjärjestusi. Sellega elimineeritakse praimeri võimalike sekundaarsete seondumiskohtade olemasolu genereeritavas PCR-i produktis. Teine etapp viiakse läbi, kui järjestus, mis sisaldab siht-märkjärjestust on pikem kui ala, mida üles tahetakse amplifitseerida. Etapp seisneb valmis disainitud praimerite sekundaarsete seostumiskohtade otsimises järjestusest, mis sisaldab siht-märkjärjestust (nt genoomsest järjestusest). Kahe mainitud ülesande lahendamiseks on loodud mitmeid lahendusi. Esimese ülesande lahendamiseks otsitakse järjestusest N-nukleotiidi pikkused M-korda esinevad alam-järjestused, ning jäetakse meelde nende esinemise asukoht. Teise probleemi lahendamiseks otsitakse etteantud alam-järjestuste

esinemiste arvu järjestusest. Mõlemate ülesannete lahendamiseks kasutatakse samu metoodikaid, millest levinumaid käsitletakse järgmises peatükis (täpsemalt punkt 4.2).

4. Kordusjärjestused

4.1 Kordusjärjestuste iseloom ja hulk

Suurte ja keeruliste genoomide laiaulatuslikul uurimisel puututakse tihti kokku probleemiga, mis on tingitud suurest korduvate järjestuste esinemisest genoomis. Enamustel eukarüootidel moodustavad korduselemendid suure osa genoomist (joonis 3). Prokarüootide puhul pole korduvate järjestuste küsimus nii oluline, kuna nende genoomid on väikesed ja mitte nii kompleksed kui eukarüootidel (Makalowski, 2001).



Joonis 3. Erinevate fraktsioonide osakaal inimese genoomis. 10 % inimese genoomist loetakse kõrgelt korduvaks, 30% mõõdukalt korduvaks ja ülejäänud genoomist esineb, kas ühe koopiana või väga väheste koopiatena.

Kordusjärjestused võib jagada üldiselt kaheks – tandeemselt korduv DNA ja hajusalt üle genoomi korduv DNA. Kordusjärjestused esinevad tüüpiliselt geenidevahelistes alades, kuid võivad asuda ka intrageenses alas. Tandeemselt korduva DNA võib jagada kaheks: satelliit DNA ja makro/mini/mikrosatelliidid. Esimesena mainitud kujutab endast miljonites

kordustes tsentromeerses alas esinevaid väga lihtsaid järjestusi mustri pikkusega 5-200 bp olenevalt eukarüoodist. Makrosatelliidid (pikkus alla 10 bp) esinevad väga pikkade klastritena (sajad kiloaluspaarid). Järjestikuste, peamiselt telomeerses alas ühe blokina esinevate minisatelliitide (nimetatakse ka VNTR, *variable number of tandem repeat*) pikkus genoomis on 1,000 - 15,000. Mikrosatelliidid on lühemad kordusjärjestused (nimetatakse ka STR, *short tandem repeat*) ja levinud üle kogu genoomi. Tüüpiliselt mikrosatelliitide mustri pikkuseks jääb alla 5 nukleotiidi ja kogu klastri pikkus alla 150 bp (Krane et al., 2002). Hajuskordusjärjestused (IRS, *interspersed repeated sequence*) ei esine tandeemselt, vaid üksikult üle genoomi paisatuna. Inimese genoomis levinumad IRS-id on LINE-d (*long interspersed nuclear elements*) ja SINE-d (*short interspersed nuclear elements*). Inimesel kõigel levinum SINE on Alu-element (pikkus ca 280 bp), mida arvatakse olevat genoomis 500,000-900,000 koopia ringis (Makalowski, 2001).

Väga suur (bio)informaatika haru tegeleb mustrite otsimisega (korduvate järjestuste) ja leidmisega (praimerite jt oligonukleotiidide) etteantud stringist (genoomist). Suurte genoomide puhul on oluline, et oleksid olemas efektiivsed (kiired, korrektsed) ja automaatsed vahendid korduste otsimiseks, leidmiseks ning maskeerimiseks. Suuremahulised, kogu genoomi hõlmavad uuringud kasutavad oma töös palju kiipide ja PCR-i tehnoloogiat. Nendes tehnoloogiates kasutatavate oligote (peavad seonduma vaid ühte spetsiifilisse kohta reaktsioonisegus) disainimine on olulise tähtsusega eksperimentide õnnestumise tagamiseks. Oluline on korduste leidmine ja maskeerimine järjestuses enne oligote disaini vältimaks sekundaarseid seostumiskohti omavate oligote disaini. Järgnevalt on toodud mõningad levinumad mustrite (järjestuste) otsimiseks ja leidmiseks kasutatavad algoritmid ja nende realisatsioonid.

4.2 Algoritmid järjestuste (korduste) leidmiseks

4.2.1 Homoloogia otsing

Homoloogia viitab kahe järjestuse ühisele päritolule. Kahe järjestuse vahelist homoloogiat otsitakse läbi kahe järjestuse vahelise sarnasuse. Sarnasus on vähem rangem

kui identsus. Homoloogia otsinguid kasutatakse kahe organismi vahelise põlvnemise leidmiseks, samuti teatud mustri (nt praimeri järjestuse) leidmiseks ette antud järjestusest (nt genoomist).

Meetodid, mis realiseerivad homoloogia otsinguid, kasutavad kahe järjestuse vahelise sarnasuse leidmiseks maatrikseid. Maatriksite abil seatakse kaks järjestust vastavusse ning ühe järjestuse igat positsiooni võrreldakse teise järjestuse kõigi positsioonidega ning igale vastavusele antakse mingi kindel skoor. Viimast lähenemist nimetatakse dünaamiliseks programmeerimiseks ja seda kasutavad erinevad homoloogia otsinguid teostavad algoritmid (Krane et al., 2002).

Tuntumad homoloogia otsingu algoritmid on FASTA (Lipman et al., 1985), BLAST (Altschul et al., 1990), BLAT (Kent, 2002). Kasutatakse neid meetodeid peamiselt kahe järjestuse vahelise homoloogia leidmiseks ja mustrite leidmiseks antud järjestusest. Tegu on lokaalsel joondamisel ja heuristikal põhinevate algoritmidega, mis ei vaata läbi kõiki võimalikke kahe järjestuse vahel esinevaid joonduse kombinatsioone, vaid kõige tõenäosemaid. Viimane asjaolu kiirendab oluliselt algoritmide tööaega. Tuntuim ja enim kasutatud algoritm on BLAST, mis jagab uuritava järjestuse 4-ühiku pikkusteks alam-sõnedeks ning otsib alam-sõnede esinemiskohti andmebaasis olevast (siht-märkjärjestusest) järjestusest. Alam-sõne leidmise korral pikendatakse sõne mõlemalt poolt kuni pikendatava alam-sõne ja andmebaasis esineva järjestuse vahelisele sarnasusele pandud skoor (skoor pannakse vastavalt eelpool kindlaks määratud maatriksile) on väiksem kui enne otsingut kindlaks määratud suurim skoor, mis veel väljendab kahe järjestuse vahelise homoloogsuse olemasolu. BLAST-il on hulk alam-programme, näiteks BLASTN, mis võrdleb nukleotiidset järjestust nukleotiidsega, BLASTP, mis võrdleb valgu järjestust valgu järjestusega jpt. Samuti on BLAST-ist tehtud modifitseeritud versioone, mis kokku moodustavad BLAST-i perekonna, kuhu kuuluvad näiteks PSI-BLAST (Altschul et al., 1997), MEGA-BLAST (Zhang et al., 2000).

4.2.2 *Hashing* meetodid

Programmide, mis põhinevad dünaamilisel programmeerimisel, pikk tööaeg - $O(n^2)$ keerukusega, (loe: suur o n ruudust, $O(n^2)$ on maksimaalne aeg, mis programmil kulub antud ülesande lahendamiseks, n - mustri pikkus nukleotiidides) pole paljude probleemide, mis hõlmavad pikki ja keerulisi järjestusi, lahendamisel rahuldav (Delcher et al., 1999). Kasutusele on võetud *hashing* meetodid, mis organiseerivad DNA järjestuse (sihtmärkjärjestuse, mille vastu mingit järjestust ehk mustrit ehk sõne otsitakse) paisktabeli (*hash table* – paisktabel) andmestruktuuri. Praimeri disaini protsessis kasutatakse h*ashing* meetodeid praimerite sekundaarsete seondumiskohtade leidmiseks genoomist.

Tuntumad paisktabeleid kasutavad programmid on SSAHA (Ning et al., 2001) ja GTESTER (Reppo et al., unpublished). SSAHA (***S**equence **S**earch and **A**lignment by **H**ashing **A**lgorithm*) algoritm jaotab DNA järjestuse alam-järjestusteks ja märgistab iga niisuguse järjestuse ühe positiivse täisarvulise väärtusega ehk indeksiga (viide ehk *pointer*). Ühe DNA järjestuse paisktabel moodustatakse vaid üks kord ning hoitakse arvuti mälus kahe andmestruktuurina – alam-järjestuste positsioone kirjeldavate suuruste (k-korteežid) list L ja jada A alam-järjestuste *pointeritest*. Kasutatakse mõistet k-korteež, mis viitab alam-järjestuse, pikkusega k bp positsioonidele antud DNA järjestuses. Erinevate *pointerite* arv sõltub alam-järjestuste pikkusest, A-l 4^k (4 erinevat nukleotiidi) *ponterit*. Näiteks, kui järjestuse pikkuseks on 2, siis on A pikkuseks 16. Sel viisil moodustatakse paisktabel ning jäetakse meelde konkreetsete k-meeride esinemised DNA järjestuses. Konkreetse järjestuse leidmine tabelist toimub indeksite järgi. Selline indeksite abil otsimine on kiire ning efektiivne ja ei sõltu paisktabeli suurusest. Limitatsiooniks võib ainult kujuneda asjaolu, et paisktabel võtab väga palju RAM mälu (Ning et al., 2001). Programmi GTESTER algoritmi tööpõhimõte on analoogne SSAHA-le. GTESTER-i (***genome TESTER***) erinevuseks SSAHA-st on see, et esimene kasutab alam-järjestuse pikkuseks fikseeritud suurust (vaikimisi autorite poolt soovituslikult 16 bp) ja seetõttu on programm paisktabeli moodustamisel suurusjärgu võrra kiirem kui SSAHA. GTESTER-i korral uuritava järjestuse otsimine paisktabelist on kiirem kui SSAHA, kuna väiksemate andmemahtude korral ei ole tarvis indekseid mällu lugeda (Remm,

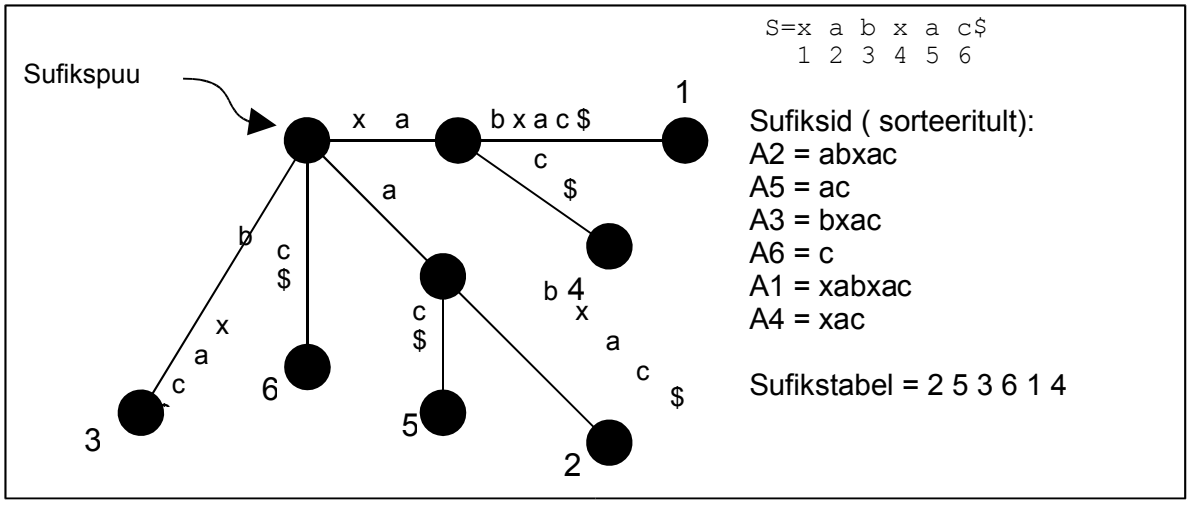
Andreson unpublished). Siiski on sellisest andmemudelist mustri leidmise kiirus O(n*log (n)) (loe: suur o n korda log n-ist), kus O(n*log(n)) on ülim aeg, mis programmi sooritamiseks kulub. Inimese genoomi puhul vajavad indeksid (paisktabel) GTESTER-i korral ca 20 Gb arvuti RAM mälu. SSAHA puhul mälu vajadus kasvab eksponentsiaalselt sõne pikkuse kasvamisega (mäluvajadus suurusjärgus 4^{k+1}+ k-meeride esinemiste arv).

Kui SSAHA on mõeldud eelkõige erinevate uuritavate järjestuste sarnasuse otsimiseks siht-märkjärjestusest, siis GTESTER spetsiaalselt PCR-i praimerite sekundaarsete seostumiskohtade testimiseks (Reppo et al., unpublished). PCR-i praimerite potentsiaalsete seostumiskohtade testimiseks genoomis on välja töötatud samuti *hashing* meetodit kasutav programm PRIMEX (***PRI****mer* ***M****atch* ***EX****tractor*), mida saab kasutada ka kiipidel kasutatavate oligonukleotiidide valimiseks ja genoomide joondamiseks. Genoomsest järjestusest paisktabeli moodustamiseks liigutakse n-sõnapikkuse aknaga genoomi esimesest nukleotiidist alates üle kogu järjestuse ning jäetakse meelde kõik akendes esinenud järjestused tabelina. Hiljem on võimalik leida uuritava järjestuse (PCR-i praimeri) esinemised moodustatud tabelist ehk genoomist. Võimalik on lubada ka uuritava ja siht-märkjärjestuse nukleotiidide vahel mitte-paardumisi (Lexa et al., 2003).

4.2.3 Sufikspuud

Sarnaselt *hashing*-meetoditega töötlevad ka sufikspuudel baseeruvad mustrite (tundmatute, uuritavate järjestuste) otsimise ja leidmise algoritmid eelnevalt DNA järjestust (genoomset DNA-d), kust erinevaid mustreid tahetakse leida või otsima hakata. Igast sõnest (DNA järjestusest) pikkusega m on võimalik genereerida maksimaalselt m sufiksit Sufikseid on võimalik hoida sufiksipuuna (joonis 4). Selline puu on võimalik valmis teha lineaarse ajaga. Puu valmistamise aeg sõltub sõne pikkusest ning kindla mustri otsimine sellisest puust sõltub lineaarselt otsitava mustri pikkusest (Gusfield, 1997). Sufikspuust uuritava järjestuse leidmine on võrreldes eelpool käsitletud lähenemistega kiirem (parimal juhul O(m + log n) keerukusega, kus m on uuritava järjestuse ja n siht-märkjärjestuse pikkus), kuid samas on sufikspuu andmestruktuur ka kõige mälu

nõudlikum. Sõltuvalt sufikspuu realisatsioonist vajavad sufikspuud erineval hulgal mälu. Ökonoomsemate puude korral vajatakse halvimal juhul algse sõne iga tähe kohta ca 20 baiti, keskmiselt vajatakse aga baiti tähe jaoks. Seega sufikspuu vajab 10-20 korda rohkem mälu kui algne sõne ise (DNA järjestus, millest sufikspuu koostatakse) (Kurtz, 1999). Inimese genoomi kohta (~3 giga-aluspaari, üks aluspaar võtab ühe baidi arvuti mälu) vajatakse keskmiselt ca 30 Gb (giga-baiti), kuid halvimal juhul ca 60 Gb.



Joonis 4. Näide sõne S esitamisest sufikspuuna ja sufikstabelina. Antud sõnel pikkusega 6 (karakterit, tähte) on 6 sufiksit. $ tähistab sufiksi lõppu.

Tuntumad programmid bioinformaatika valdkonnas, mis kasutavad oma töö teostamiseks sufikspuid on genoomide joondamiseks kasutatavad MUMmer (Delcher et al., 1999), edasiarendus sellest MUMmer2 (Delcher et al., 2002) ja MGA (*Multiple Genome Aligner,* Höhl et al., 2002) ning korduste leidmiseks genoomist REPuter (Kutz et al., 1999). Viimane programm võimaldab leida genoomist kõik seal esinevad kordused. REPuter-i töö aeg ja mäluvajadus sõltuvad lineaarselt genoomi pikkusest ja seal sisalduvatest kordusjärjestuste hulgast (Kutz et al., 1999).

4.2.4 Burrows-Wheeleri transformatsioon (BWT)

Esmakordselt avalikustati algoritm 1994 aastal ja on kasutusel mitmetes suurte andmemahtudega kokkupuutuvates valdkondades. Algoritmi kasutatakse andmete kompressiooniks. BWT (*Burrows-Wheeler Transform*) algoritm võtab andmed (näiteks DNA järjestuse) ja kasutades sorteerimist muudab andmete struktuuri ümber. Transformatsioon on pööratav, st sorteeritud andmetest on võimalik taastada täpne algne andmete struktuur (Nelson, 1996). Nagu eelpool käsitletud algoritmidest näha, on nii *hashing* meetodites kui ka sufikspuudes kasutatav andmemudel mälunõudlik ning võib seega muutuda paljusid probleeme lahendades limiteerivaks asjaoluks. Healy koos kaasautoritega kasutasid BWT algoritmi, et leida antud mustri (uuritava järjestuse) esinemiste arvu antud siht-märkjärjestuses (genoomis). Realiseerides oma programmis BWT algoritmi suudavad nad inimese genoomi kokku pakkida nii, et ca 3 Gb (inimese genoomi suurus aluspaarides võtab arvuliselt sama palju mälubaite) mälu asemel vajatakse ca 1 Gb mälu. Programmi töös on elimineeritud ajakulukas andmete kõvakettale kirjutamine ning genereeritakse andmetüüp, millest on lihtne andmeid pärida ja, mis võib täies mahus asuda arvuti muutmälus ehk RAM-is. Siht-märkjärjestusest (pikkusega n) transformatsiooni moodustamise aeg on maksimaalselt O(n * log n) keerukusega. Kõigi 24-meeride esinemiste arvu arvutamine tervest inimese genoomist koos tulemuste kirjutamisega kettale võtab 1 minuti mega-aluspaari kohta aega (Healy et al., 2003).

II Praktiline osa

Töö eesmärgid

Arvestades eukarüootsete genoomide suurt mahtu ja teadlaste vajadust genoomsest DNA-st kodeerivaid regioone üles-amplifitseerida (genotüpiseerimiseks jt mutatsioon-analüüsideks), on vajadus automaatse PCR-i praimerite disainimise järele möödapääsmatu. Tänapäeval on saadaval paljud PCR-i praimerite disainimise programmid, seda nii veebiserverites kui ka personaalarvutitel töötavate programmidena. Sellised programmid ei arvesta korduvate järjestustega ja praimerite sekundaarsete seostumiskohtadega kogu genoomis ning pole seega sobivad praimerite disainimiseks kogu genoomi hõlmavate uuringute jaoks. Samuti ei ole veebiserveritel töötavad programmid sobivad suuremahuliste uuringute jaoks vajaminevate praimerite genereerimiseks, sest tihti on serveri poolselt seatud piirangud genereeritavate andmete mälumahu ja kettavajaduse osas.

Käesoleva töö peamiseks eesmärgiks oli välja töötada meetod, mis võimaldab eukarüootsete genoomide kõikide kodeerivate alade amplifitseerimiseks vajalike PCR-i praimerite automaatset disaini. Ülesandeks oli leida eukarüootse genoomi võimalikult paljudele geenidele võimalikult kvaliteetsed PCR-i praimerid optimaalse tööajaga. Töö hõlmab ka nimetatud meetodi abil eukarüootsetele organismidele praimerite disainimist ja vastava andembaasi loomist. Samuti luuakse graafiline kasutajaliides disainitud praimerite võrguvahendusel kättesaamiseks.

Meetodid

1. Andmete päritolu ja struktuur

Liike, kellele me praimerid disainisime on 5. Valitud liigid on inimene *Homo sapiens,* kaks närilist – *Mus musculus, Rattus norvegicus* ja kaks pärmi *Saccharomyces cerevisiae, Schizosaccharomyces pombe.* Liikide valimisel arvestasime sellega, milliste liikidega enam teadusmaailmas tegeletakse ning, milliste liikide kohta on saadaval rohkem usaldusväärset infot.

Kolme organismi – inimene (24 kromosoomi), hiir (21 kromosoomi) ja rott (21 kromosoomi) genoomi nukleotiidse järjestuse saime avalikust ENSEMBL MySQL serverist (*kaka.sanger.ac.uk*). Inimese korral kasutasime genoomi versiooni 34, hiire ja roti puhul vastavalt 32. ja 3. genoomiversiooni. Annotatsiooniandmete saamiseks kasutasime kõigi kõneall oleva kolme liigi korral kolmest erinevast annotastioonikeskusest saadud andmeid – ENSEMBL, NCBI, VEGA (viimase puhul puuduvad roti genoomi andmed). Andmete hankimise hetkel oli VEGA annotatsioonikeskusest võimalik saada infot inimese 6., 7., 13., 14., 20., 22. ja hiire 13. (Del36H regioon) kromosoomis leiduvate geenide kohta. Pärmide *S. cerevisiae* (16 kromosoomi) ja *S. pombe* (3 kromosoomi) kohta saime nii järjestuse info kui ka annotatsiooniandmed vastavalt Stanford ülikooli alla kuuluvast SGD-andmebaasist ja *The Wellcome Trust Sanger* instituudi ftp-serverist.

Kasutatud annotatsiooniandmed kujutavad endast eksonite algus- ja lõppkoordinaati, vastavas annotatsioonikeskuses geeni annoteerimisel talle pandud süstemaatilist nime, geeninimede sünonüüme ja aliasi (võimalikult palju erinevaid). Geeninimede aliasi on meil vaja selleks, et hiljem genereeritavas graafilises kasutajaliideses (GUI, **G***raphical* ***U****ser* ***I****nterface*) saaksid kasutajad võimalikult mugavalt neid huvitavat

geeni üles-amplifitseeritavad praimerid kätte. Sellest, milliseid geeninimesid me ühe geeni kohta pakume, annavad ülevaate tabel 3 ja tabel 4. Tabel 3-s kujutab ENSEMBL ID endast annotatsioonikeskuse ENSEMBL süstemaatilist geeninime (nt ENSG00000186891), NCBI ID on NCBI poolt pandud RefSeq geeni nimele lisanimi (võivad olla redundantsed; näide nimele TNFRSF18 vastavad RefSeq geeninimed NM_148902, NM_148901, NM_004195), HUGO ID (tuntud ka kui HGNC ID – ***H**ugo **G**ene **N**omenclature **C**ommittee ID*) on Inimese Genoomi Organisatsiooni (*The Human Genome Organisation*'i) poolt geenile pandud geeninimi (nt 11914). RefSeq ID-d (***REF**erence **SEQ**uence ID*) on mRNA-dele vastavad geeninimed, LocusLink ID (nt 8784) seab igale mRNA-le vastavusse ühe lookuse (Pruitt et al., 2000; Pruitt KD et al., 2001). Geneetiliste haigustega seotud geeninimi MIM ID (tuntud ka kui OMIM – *Online Mendelian Inheritance in Man*, nt 603905), VEGA ID (nt OTTHUMG00022002632) on manuaalselt annoteeritud geenide süstemaatiline nimi. SwisProt ID (nt Q9Y5U5) on lookusele vastava proteiini nimi.

Tabel 3. ENSEMBL, NCBI ja VEGA annotatsioonikeskustest saadud geeninimed vastavalt liikidele.

Geeni-nimi / Liik	ENSEMBL ID	NCBI ID	HUGO ID	RefSeq ID	LocusLink ID	MIM ID	VEGA ID	SWIS-SPROT ID
H. sapiens	ens	ncbi	ens/vega	ens/ncbi/vega	ens/ncbi/vega	ncbi	vega	
M. musculus	ens	ncbi	ens	ens/ncbi	ens/ncbi	ens/ncbi	vega	
R. norvegicus	ens	ncbi		ens/ncbi	ens/ncbi	ncbi		ens

ens – ENSEMBL, ncbi – NCBI, vega –VEGA

Tabel 4. Kahe pärmi *S.cerevisiae* ja *S.pombe* geenide nimed ja sünonüümid

geeni-nimi / Liik	SGD süstemaatiline ID	SGD standard geeninimi	SGD alias	*S.pombe* süstemaatiline	*S.pombe* standard geeninimi	*S.pombe* alias
S. cerevisiae	+	+	+			
S. pombe				+	+	+

SGD – *S. cerevisiae* genoomi andmebaas

2. Kasutatud riistvara

Praimeridisaini programmide kirjutamiseks ja käivitamiseks kasutasime riistvarana arvutit MicroLink Novator 5000HG. Antud arvuti protsessori tüüp on Intel Xeon, arvuti on kaheprotsessoriline, mõlemate protsessorite kiirus on 2.6 GHz. Arvuti maksimaalne põhimälu ehk RAM on 6 GB, Maxtor Atlas RAID 0 10K Ultra320 SCSI kõvaketta maht on 5 * 147 GB. Operatsioonisüsteemina kasutasime Linux/UNIX süsteemi.

3. Kasutatud tarkvara

Valmisolevatest programmidest kasutasime töö käigus nelja peamist programmi: maskeerimisprogramme DUST ja GMASKER, PCR-i praimereid genereerivat PRIMER3 programmi ning genereeritud praimerite sekundaarsete seondumiskohtade olemasolu genoomis kontrollivat GTESTERIT.

DUST on tasuta tarkvara, mis maskeerib etteantud järjestusest kõik vähe-keerukad mitte-informatiivsed korduvad järjestused (*low complexity region*). Sellisteks järjestusteks on mikrosatelliidid ja mononukleotiidsed kordused. Programmi DUST kasutasime enne praimerite disaini üles-amplifitseeritava regiooni maskeerimiseks. Kasutatud DUST versioon DUST_LOWER transleerib järjestuses kirjeldatud kordused väiketähtedeks (Remm , unpublished).

Maskeerimisprogramm GMASKER kuulub korduvate järjestuste otsimis- ja maskeerimisprogrammide paketti GenomeMasker (Reppo et al., unpublished). Erinevalt eelpool kirjeldatud DUST-ist maskeerib antud programm etteantud järjestusest keerulisemad kordused, viimased antakse programmile sisendiks faili kujul. Keerulisemad kordused on genoomist leitud programmi BLACKLISTER abiga, mis otsib kasutaja poolt defineeritud pikkusega üle-esindatud sõnad (nn *black list* e must nimekiri) etteantud järjestusest.

GMASKERI eripäraks on see, et ta ei maskeeri tervet üle-esindatud sõna, vaid ainult korduva järjestuse 3' otsa viimase nukleotiidi asendades suure tähe väikese tähe sümboliga. Kuna programm ei kasuta järjestuse joondamise algoritme, vaid üle-esindatud sõnade musta nimekirja, on ta oluliselt kiirem kui joondamisalgoritme kasutavad programmid. Samast programmipaketist kasutasime ka programmi GTESTER, mis kontrollib etteantud sõna esinemiskordi siht-märkjärjestuses (Remm et al., unpublished).

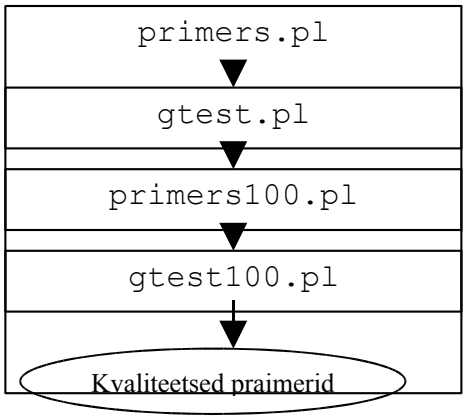
Praimerite disainimiseks kasutasime programmi PRIMER3 modifikatsiooni GM_PRIMER3 GenomeMasker paketist. Erinevalt originaal-programmist eemaldab GM_PRIMER3 vaatluse alt praimerid, mille 3' otsas on väiketäht (st on maskeeritud) (Remm, unpublished). PRIMER3 programmis saab kasutaja ise määrata PCR parameetrite optimaalsed, maksimaalsed ja minimaalsed väärtused (Rozen et al., 2000).

5. Programmide käivitamise ja praimeri disaini üldine põhimõte

Praimerite disainimisel oli eesmärgiks genereerida võimalikult kvaliteetsed praimerid antud organismi kogu genoomi mastaabis. Viimane tähendab seda, et praimer peab vastama kõigile nõutud PCR-i kvaliteeti mõjutavatele parameetritele.

Analüüsides geenide pikkusi, jõudsime järeldusele, et pikki geene pole võimalik ühe fragmendina konventsionaalse PCR-iga üles amplifitseerida, vaid vaja oleks moodustada eraldi amplikonid lookuste paljundamiseks. Üle 3,000 nukleotiidi pikkustest geenidest tekitasime blokid programmi `amplicons.pl` abil, nii et üks geen võib olla jagatud mitme amplikoni vahel, kuid üks amplikon ei või sisaldada rohkemat kui ühte geeni. Seejärel maskeerisime amplikonid programmidega DUST_LOWER ja GMASKER ning maskeerisime amplikonides ka SNP-de kohad. SNP-de maskeerimine on vajalik selleks, et vältida praimerite disainimist varieeruva nukleotiidse koostisega aladesse. Saadud järjestusele disainisime praimerid programmiga GM_PRIMER3. Maskeerimine ja praimerite disainimine on integreeritud programmi nimega `primers.pl`. Praimerite sekundaarsete seondumiskohtade

testimiseks kasutasime GTESTER programmi. Seondumiskohtade kvaliteedi kontrollimiseks kirjutasime programmi `gtest.pl`. Kui praimer sisaldas rohkemat kui ühte seondumiskohta antud organismi genoomis, siis eemaldati praimer (praimeripaar) vaatluse alt ning disainiti eba-kvaliteetset praimeripaari omanud amplikonile asemele 100 uut praimeripaari (vastav programm `primers100.pl`) ning testiti saadud 100 praimeripaari seondumiskohti (`gtest100.pl`) üle genoomi. Sellise põhimõttega (joonis 5) saavutati ca 98%-le disainitud amplikonidest kvaliteetsete praimerite disain.



Joonis 5. Praimerite disainimise programmide käivitamise järjekord.

6. Kasutatud PCR-i parameetrid

Meie poolt kasutatud praimeridisaini programmis, GM_PRIMER3-s määrasime PCR-i parameetritele minimaalse, maksimaalse ja optimaalse väärtuse. Programm ei disaini

praimereid, mille korral vastava PCR-i parameetri väärtus on maksimaalsest kõrgem või minimaalsest madalam; püütakse leida praimerid, mille vastava parameetri väärtus oleks võimalikult optimaalse lähedal. PCR-i parameetrite väärtuste valimisel juhindusime eelkõige publikatsioonides ilmunud andmetest. Genereeritava PCR-i produkti pikkus võib kõikuda vahemikus 100 - 6,000 nukleotiidi. Minimaalne produkti pikkus on valitud 100 aluspaari, kuna alla 100 bp pikkuseid produkte ei genereerita ning maksimaalne pikkus on 6,000 bp, sest disainitud produkti pikkus on alati alla 6,000 bp (vt täpsemalt algoritmi kirjeldust – Tulemused). Produkti optimaalseks pikkuseks valisime 600 aluspaari, kuna enamus genereeritud amplikone on alla 1000 aluspaari pikad. Praimeri optimaalne, minimaalne ja maksimaalne pikkus on vastavalt 21, 18 ja 30 aluspaari. Praimerite optimaalne, minimaalne ja maksimaalne sulamistemperatuur on vastavalt 60, 59, 65 ^{o}C. Praimeripaari kuuluvate praimerite sulamistemperatuuride maksimaalne lubatud erinevus on 4 ^{o}C , praimerite optimaalne, minimaalne ja maksimaalne GC nukleotiidide sisaldus on vastavalt 45, 20, 80 %. Soolakontsentratsioon (KCl) on 20 mM ja maksimaalselt lubatud järjestikku esinevaid mononukleotiidede arv on 4 tk.

7. Programmide aja- ja mälukasutuse mõõtmise metoodika

Programmide aja- ja mälukasutamise mõõtmiseks kasutasime UNIX-i käsku ‘top’, mis võimaldab interaktiivselt jälgida süsteemis jooksvaid protsesse. Vaatasime programmide CPU (CPU *usage*) ja mälu (MEM%) kasutust ning programmi tööaega (CPU *time*). CPU (*Central Processing Unit*) on arvuti keskprotsessor ehk arvuti ‘aju’, kus toimub kõigi loogika- ja aritmeetikatehete teostamine ja erinevate käskude täitmine (tarkvaras sisalduva info interpreteerimine ja vastavate instruktsioonide täitmine). CPU kasutus (CPU%) näitab, kui suures osas kasutab programm CPU-d (kui palju kogu programmi tööaja vältel (CPU *time*) kasutatakse arvuti protsessorit). Programmi tööaeg näitab, kui palju on kulunud aega programmi käivitamisest programmi töö lõpetamiseni. Programmi mälu kasutus näitab kui palju kasutab programm töötamisel kättesaadavat

füüsilist arvuti mälu (nt kui palju programm kirjutab arvutimällu), st mitu protsenti vabast füüsilisest mälust programm kasutab. Arvutasime ka kasutatud programmide kettavajaduse, so kui palju programm maksimaalselt kasutab kõvaketast oma töö sooritamiseks. Kettavajadus määrab ära see, kui palju programm kirjutab kõvakettale või kui palju ta loeb kettalt.

Tulemused

1. Kasutatud geenide kirjeldus

Praimerid disainisime viie organismi geenidele. Valitud organismideks olid *Homo sapiens, Mus musculus, Rattus norvegicus, Saccharomyces cerevisiae, Schizosaccharomyces pombe*.

Inimese, hiire ja roti genoomide pikkused on ühes suurusjärgus (vastavalt ca 2.9 Gb, 2.6 Gb, 2.75 Gb), kolme liigi geenide arvud, eksonite keskmised pikkused ja arvud geeni kohta on sarnased ning nad kodeerivad ka sarnase hulga konserveerunud intron-struktuuriga geene. Kõne all oleva kolme organismi üle 80% geenidest sisaldavad introneid, intronite arv ühe geeni kohta on suhteliselt kõrge. Näiteks roti geenis sisaldub keskmiselt 9.7 intronit (Gibbs et al., 2004) ja inimese genoomis keskmiselt 7-8 intronit geeni kohta (Garcia-Blanco et al., 2004).

Pärmide *S. cerevisiae* ja *S. pombe* genoomid on täielikult sekveneeritud (genoomide suurused vastavalt 12.06 Mb ja 13.8 Mb). *S. cerevisiae* genoomis sisalduvate geenide ennustatavaks arvuks pakutakse 5,300 - 5,400. Introneid on geenidest leitud vähe (5%) ning nad on lühikesed (Mackiewicz et al., 2002). *S. pombe* genoomi annoteerimisega on palju tegeletud ja ennustatavalt sisaldab ta genoom natuke vähem kui 5,000 (4,824) proteiini-kodeerivat geeni, mis on vähim teadaolev valku-kodeerivate geenide komplekt eukarüootses organismis. Erinevalt *S. cerevisiae*-st sisaldavad *S. pombe* geenid rohkem introne (43% geenidest sisaldavad introne, kokku introne 4,730). Ühes geenis on enamasti alla 3 introni (34 geenil on täheldataud 7-15 intronit), intronid on lühikesed (keskmine pikkus 81 nt). 50 *S. pombe* geeni omavad homoloogiat inimese haigustega (pooled vähiga) seotud geenidega. Nagu *S.cerevisiae* genoomgi sisaldab ka *S* .

pombe genoom vähe korduvat DNA-d ja on kompaktne (Wood et al., 2002). Mõlemat pärmi kasutatakse laialdaselt erinevates genoomiuuringutes. Olenemata sellest, et mõlemad organismid kuuluvad seeneriiki (samasse hõimkonda *Ascomycota*), on kaks organismi väga kaugelt suguluses (lahknenud 330 – 420 miljonit aastat tagasi) ning nende genoomne organiseeritus ning elutsükkel on väga erinev (Sipiczki, 2000). Kasutatud geenide ja eksonite arvust liikide ja nende annotatsioonikeskuste lõikes annab ülevaate tabel 5.

Tabel 5. Kasutatatud geenide ja eksonite arv liikide ja annotatsioonikeskuste läbilõikes

LIIK	GEENIDE ARV	EKSONITE ARV
*H. Sapiens**	21787/16887**/3995	275,913/222,348/55533
*M. musculus**	25,308/16,255**/333	293,610/152,556/1218
*R. norvegicus****	22159/4694**	218040/47536
S. cerevisiae	4122****	4363
S. pombe	5016	9938

*ENSEMBL/NCBI/VEGA annotatsioonikeskuste andmed
**ainult RefSeq geenid
***ENSEMBL/NCBI annotatsioonikeskuste andmed
****ainult verifitseeritud (eksperimentaalselt tõestatud) geenide eksonid

Geenide keskmised pikkused organismiti on varieeruvad. Organismidel, kelle geenid sisaldavad rohkem introne, on geenide keskmised pikkused suuremad (inimene, hiir, rott) kui neil organismidel, kelle geenides on intronid suhteliselt harva esinevad (*S.pombe, S. cerevisiae*) (tabel 6). Analüüsides geenide keskmisi pikkuseid annotatsioonikeskuste lõikes, on näha, et NCBI RefSeq geenide (mRNA-l põhinevad, mitte-redundantsed) keskmised pikkused on kõrgemad kui vastava organismi ENSEMBL annotatsioonikeskuse poolt ennustatud geenide pikkused. NCBI RefSeq geenide kvaliteedile annab usaldusväärsust asjaolu, et inimese manuaalselt annoteeritud (VEGA) geenide keskmised pikkused on samas suurusjärgus. Hiire VEGA annotatsioonikeskuse poolt annoteeritud geenide keskmine pikkus võrreldes ülejäänud kahe annotatsioonikeskuse geenide keskmiste pikkustega, on suhteliselt erinev. See asjaolu on tingitud tõenäoliselt faktist, et VEGA poolt on annoteeritud hiire genoomist statistiliselt eba-informatiivne ala

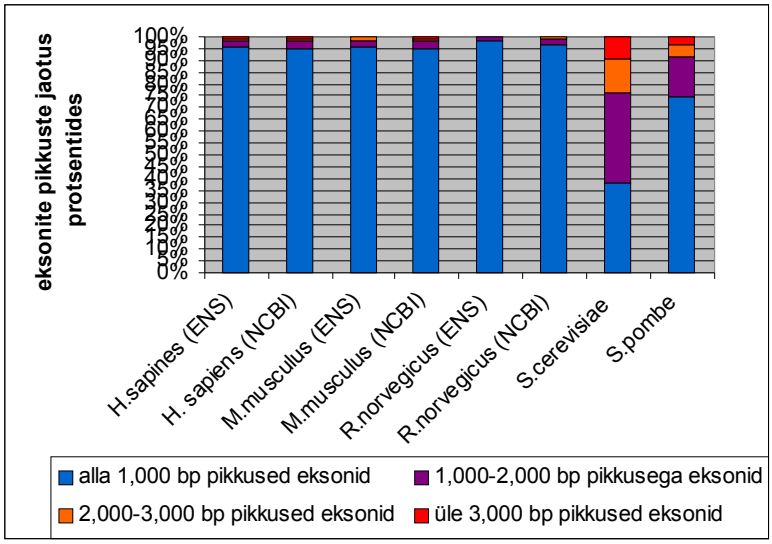
(13-nda kromosoomi suhteliselt lühike tihedalt geene sisaldav spetsiifiline regioon), st annoteeritud piirkond ei iseloomusta kogu genoomi.

Tabel 6. Keskmised geenide pikkused organismidele vastavate annotatsioonikeskuste lõikes

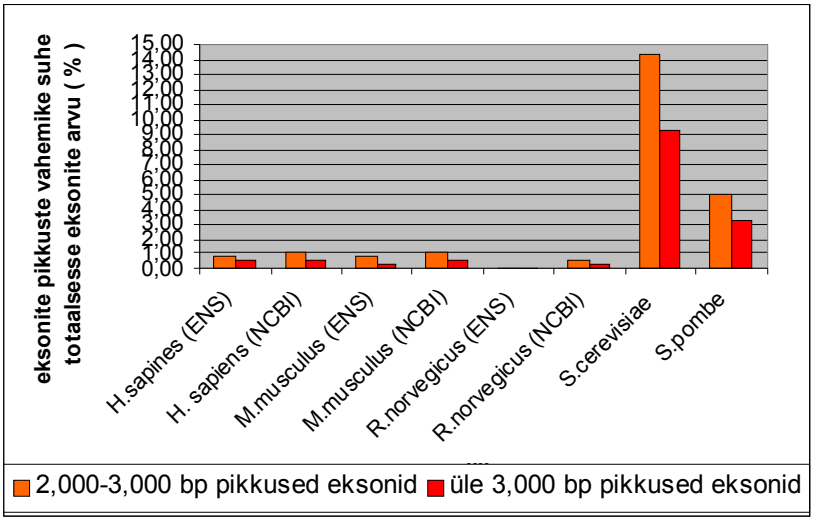
Liik	Geenide keskmine pikkus (bp)
*H.sapiens**	39,633/46,634/45,951
*M.musculus**	25,850/33,247/2,676
*R.norvergicus***	23,911/47,131
S.cerevisiae	1590
S.pombe	1478

*ENSEMBL/NCBI/VEGA
**ENSEMBL/NCBI

Samuti on sarnased sama organismi erineva annotatsioonikeskuse poolt ennustatud geenides sisalduvate eksonite pikkuste vahemikud. Kõigi viie organismi kõigi geenide eksonitest üle 90% on alla 3,000 aluspaari pikad (joonis 6). Inimesel, hiirel ja rotil on üle 90% eksonitest alla 1,000 aluspaari pikad. Pärmidel *S.pombe* ja *S.cerevisiae*-l on eksonite pikkuste jaotus erinev inimese, hiire ja roti eksonite pikkuste jaotusest (Joonis 6, joonis 7). Selle põhjuseks on asjaolu, et nende geenid sisaldavad võrreldes kolme ülejäänud organismiga suhteliselt vähe (vastavalt 43% ja 5%) introneid. Korrelatsiooni intronite arvu ja eksonite pikkuste vahel on näha ka analüüsides alla 1,000 aluspaari pikkuste eksonite arvu. *S.cerevisiae*-l on alla 1,000 nukleotiidi pikkuseid eksoneid ca 37% samal ajal kui *S.pombe*-l on sama pikkuse vahemikku jäävaid eksoneid ca 75% kogu eksonite arvust.



Joonis 6. Eksonite pikkuste vahemike jaotus organismide ja nendele vastavate annotatsioonikeskuste lõikes. Kõigi viie organismi geenide eksonitest üle 90% on alla 3,000 nukleotiidi pikad. ENS – annotatsioonikeskuse ENSEMBL annotatsiooni andemd, NCBI – annotatsioonikeskuse NCBI annotatsiooni andmed.



Joonis 7. 2,000 nukleotiidist pikemate eksonite osakaal kõigisse antud genoomis leiduvatesse eksonitesse organismide ja nendele vastavate annotatsioonikeskuste läbilõikes. ENS – annotatsioonikeskuse ENSEMBL annotatsiooni andmed, NCBI – annotatsioonikeskuse NCBI annotatsiooni andmed.

2. Praimeri disaini programmide loomine

Välja töötatud praimerite disaini metoodika koosneb viiest peamisest etapist – geenidest PCR-i abil üles-amplifitseeritavate amplikonide disainimine, amplikonidele praimerite disainimine, saadud praimerite sekundaarsete seostumiskohtade testimine genoomis, testimisel ebakvaliteetseks tunnistatud praimeripaaride asemele saja uue praimeripaari disainimine ja viimaks saja praimeripaari seostumiskohtade kontroll genoomis.

2.1 Amplikonide moodustamine

Esimene etapp on kõigist genoomis leiduvatest geenidest moodustada sobilikud PCR-il üles-amplifitseeritavad amplikonid (kirjutatud programm `amplicons.pl`). Kuna konventsionaalse PCR-i abil ei saa väga pikki fragmente üles-amplifitseerida, siis oli otstarbekas pikad geenid jaotada lühemateks osadeks. Amplikonid moodustasime maksimaalse pikkusega 3,000 aluspaari, kuna üle 90% kõigi valitud organismide geenide eksonitest on alla 3,000 nukleotiidi pikad. Kui geeni pikkus on üle 3,000 aluspaari, siis geen jaotatakse ekson-intron üleminekukohtadelt amplikonideks, nii et amplikoni pikkus ei ületaks lubatud piiri. Seega üks geen võib kuuluda erinevatesse amplikonidesse. Kui tegu oli eksoniga, mis on pikem kui 3,000 aluspaari, siis jagati ekson vastavalt sellele osadeks, kui palju kordi defineeritud amplikoni pikkus antud eksoni pikkusesse mahtus. Näiteks kui eksoni pikkus x on 8,000 nukleotiidi ja defineeritud amplikoni pikkus X on 3,000 bp, siis teostati antud eksonist moodustatavate amplikonide arvu N teada saamiseks järgnev aritmeetikatehe:

$$N = \text{ceil}(x/X) = \text{ceil}(8{,}000/3{,}000) = 3 \text{ (i)},$$

kus kasutatakse funktsiooni ceil(), mis ümardab jagamistehte tulemusena saadud arvu ülespoole.

Seejärel jaotame eksoni pikkuse x vastavalt amplikonide arvule N võrdse pikkusega osadeks järgnevat tehet kasutades:

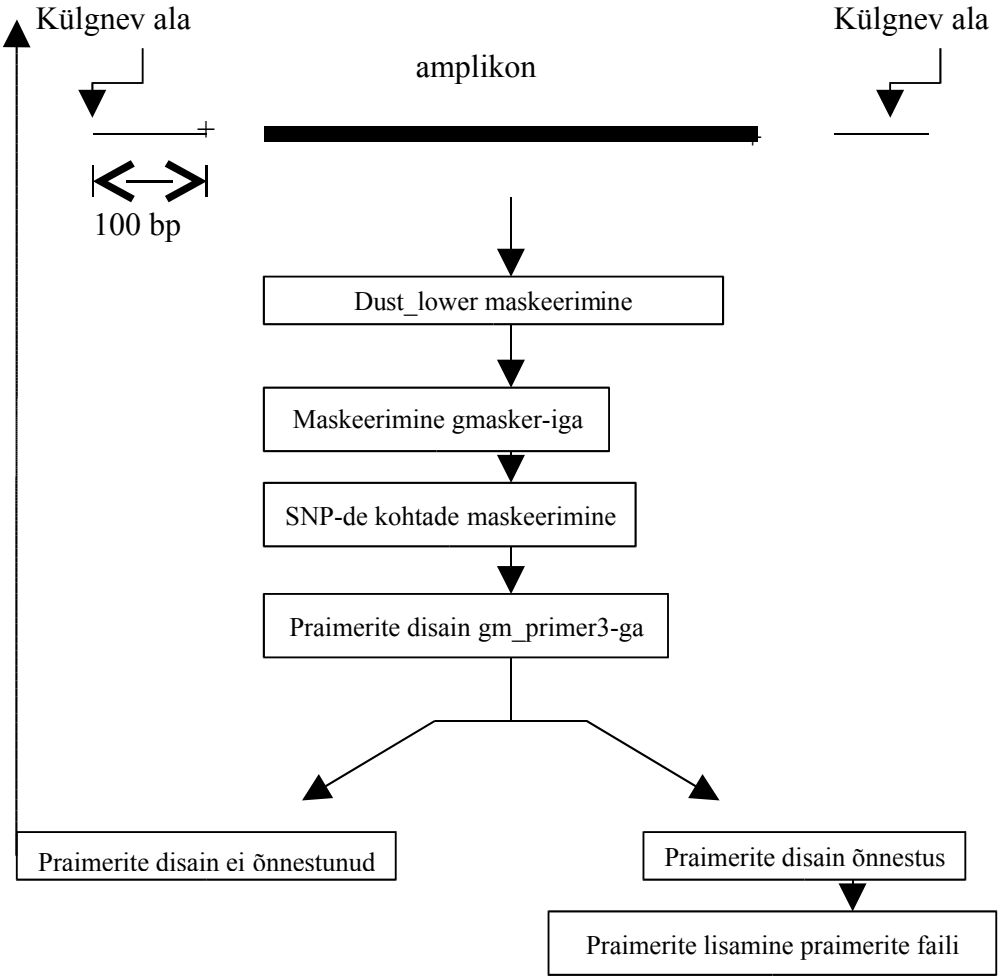
$$Y = \text{floor}(x/N) = \text{floor}(8{,}000/3) = 2{,}000 \text{ (ii)},$$

kus kasutatakse funktsiooni floor(), mis ümardab jagamistehte tulemusena saadud arvu allapoole. Eksoni osadeks jaotamisel viimase amplikoni genereerimisel määratakse amplikoni lõppkoordinaadiks eksoni lõppkoordinaat vältides sellega viimase jagamistehte (ii) tulemusena osade nukleotiidide 'kaotsi' minemist eksoni lõpust.

2.2 Praimerite disainimine amplikonidele

Genereeritud amplikonidele liitsime kummalegi poole otsa 100 aluspaari külgnevat regiooni (nn *flanking region*) praimerite disainimise kohaks. Seejärel teostasime saadud järjestuste maskeerimise programmidega DUST_LOWER ning GMASKER. DUST_LOWER maskeerib 15 nukleotiidi pikkused madala-keerukusega alad genoomist. GMAKSER saab sisendiks programmiga DUST_LOWER maskeeritud amplikonide järjestused ja korduvate järjestuste nn musta nimekirja ning maskeerib antud järjestuses olevate korduvate järjestuste 3' otsa. Järgnevalt maskeeritakse DUST_LOWER-i ja GMASKER-i poolt töödeldud järjestuses SNP-de kohad. SNP-de maskeerimine toimus inimese, hiire ja roti amplikonide korral, kuna pärmide genoomides sisalduvate SNP-de kohta pole saada piisavalt andmeid. SNP-de asukoha teadasaamiseks kasutati inimese, hiire ja roti korral NCBI dbsnp tabelite versioone, vastavalt 119, 119, 117. Külgnevatesse regioonidesse disainitakse üks praimeripaar programmi GM_PRIMER3 poolt. Kui amplikoni järjestus sisaldas palju korduvat järjestust, siis selliste amplikonide külgnevatele piirkondadele ei õnnestunud praimereid disainida. Sellest probleemist saime üle tsüklite lisamisega antud programmi. Üks tsükkel kujutab endast amplikoni algus- ja lõppkoordinaadile 100 aluspaari võrra pikema külgneva piirkonna lisamist kui eelmises tsüklis ning seejärel kirjeldatud maskeerimiste läbiviimist ja praimerite disaini. Katsetasime läbi erinevaid tsüklite arve ning testimise tulemusena kujunes optimaalseks tsüklite arvuks 12 (13-ne ja enama tsükliga olulist praimerite arvu

tõusu polnud märgata). Viimases tsüklis (12-ndas) liidetakse algsele amplikonile praimerite disainiks 1,200 nukleotiidi külgnevat regiooni 5' ja 3' otsa. Kuni kuuenda tsüklini kasutab GMASKER kordusjärjestuste 'musta' nimekirjana faili, kus on kirjas 16-ne (inimene, hiir, rott) või 12-ne (*S.cerevisiae, S.pombe*) nukleotiidi pikkused genoomis rohkem kui 10 korda esinevad järjestused. Alates kuuendast tsüklist kasutab GMASKER 'musta' nimekirja, kus on kirjas genoomis üle 30 korra esinevad vastavalt, kas 16 või 12 nukleotiidi pikkused järjestused. Praimereid disainiva programmi nimi on `primers.pl` (joonis 8).



Joonis 8. Programmi `primers.pl` tööpõhimõte. Moodustatud amplikonidele lisatakse nii 5' kui ka 3' otsa 100 bp külgnevat regiooni, seejärel maskeeritakse saadud järjestus programmidega DUST_LOWER ja GMASKER ning toimub järjestuse maskeerimine SNP-de

kohalt. Viimase etapina disainitakse külgnevasse regiooni praimerid. Juhul kui ei õnnestunud antud regiooni praimereid disainida, alustatakse uuesti esimesest etapist ning lisatakse amplikoni 5' ja 3' otsa 100 bp võrra pikem külgnev regioon kui eelmises tsüklis. Selliseid tsükleid teostatakse 12 korda.

2.3 Praimerite seondumiskoha unikaalsuse kontroll genoomis

Programm `gtest.pl` kontrollib amplikonidele disainitud praimerite sekundaarsete seondumiskohtade olemasolu kogu genoomis ja jätab meelde amplikonid, millele disainitud praimerid omasid rohkemat kui ühte seondumiskohta üle genoomi. Seondumiskohtade unikaalsuse kontrolli üle genoomi teostab programm GTESTER.

2.4 Praimerite disaini uus ring

Genoomitestis ebakvaliteetseteks osutunud praimerite asemele püütakse disainida `primers.pl` programmi tööpõhimõttega analoogse programmi `primers100.pl` abil vastavale amplikonile 100 uut praimeripaari. Saadud primerite seostumiskohti genoomis testitakse `gtest.pl` programmiga analoogse programmi `gtest100.pl` abil. Saadud kvaliteetsetest praimeritest, st praimeritest, mis omasid unikaalset seostumiskohta genoomis, moodustatakse andmebaasi tabel.

3. Tulemuste kirjeldus

Kuna üks geen võib kuuluda mitmesse disainitud amplikoni, siis on amplikonide koguarv suurem geenide koguarvust. Disainitud amplikonide arv organismiti korreleerub geenide keskmise pikkusega – mida pikem on organismi keskmine geenipikkus, seda rohkem on ka disainitud amplikone ühe geeni kohta. Samuti on näha seost geenides sisalduvate intronite ja keskmiselt ühele geenile disainitud amplikonide arvu vahel. Kui arvestada annotatsioonikeskuste andmeid, mis sisaldavad infot kogu genoomi geenide

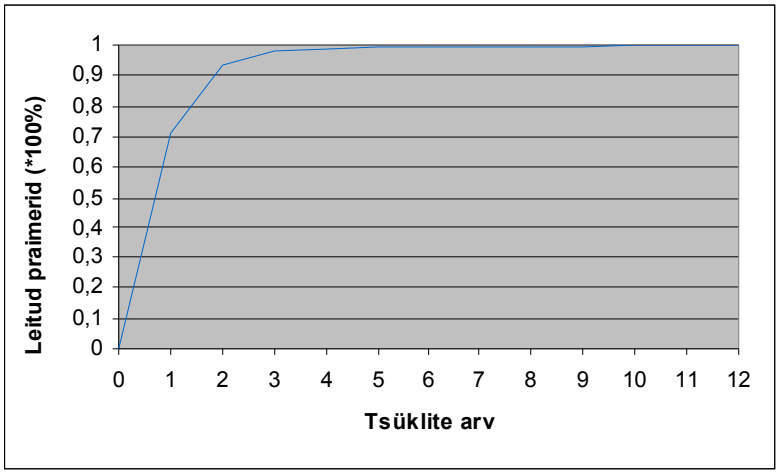
kohta, siis inimese genoomi korral on ühele geenile keskmiselt disainitud ca 5.35 amplikoni , hiire, roti, *S. cerevisiae* ja *S. pombe* korral on vastavad arvud ca 4.35, 4.4, 1.1 ja 1.1 (Tabel 7).

Tabel 7. Disainitud amplikonide arvud ja disainitud amplikonide keskmised arvud geeni kohta

Liik	Amplikonide arv	Amplikonide arv geeni kohta
*H.sapiens**	108,850/97,230/21,508	5.0/5.7/5.4
*M.musculus**	98,126/77,499/705	3.9/4.8/2.1
*R.norvergicus***	81,281/23,710	3.7/5.1
S.cerevisiae	4,609	1.1
S.pombe	5,657	1.1

*ENSEMBL/NCBI/VEGA
**ENSEMBL/NCBI

Programmi `primers.pl` abil leiti esimese tsükli töö tulemusena olenevalt organismist 70-77 % praimeritest, viie esimese tsükli jooksul leiti üle 99%-le amplikonidest praimerid. 12-nda tsükli lõpuks oli leitud praimerid kõigi organismide korral 99.99%-le amplikonidele. Maksimaalselt 0.01%-le amplikonidele ei suudetud praimereid disainida, kuna nende ümber polnud piisavalt järjestust sekveneeritud või oli amplikonidega külgnev ala väga korduv ja seetõttu ära maskeeritud (Joonis 9).



Joonis 9. Primers.pl poolt amplikonidele leitud praimeritepaaride arvu suhe totaalsesse amplikonide arvu programmis toimuvate tsüklite läbilõikes.

Praimerite seostumiskohtade kontrollimisel üle genoomi programmiga gtest.pl, osutus olenevalt liigist kuni 3.5 % disainitud praimeritest ebakvaliteetseteks (vaatluse alla ei ole võetud VEGA poolt annoteeritud hiire geenide kohta käivaid andmeid, kuna hiirel on VEGA poolt annoteeritud vaid üks lühike geenitihe ala, mis ei peegelda kogu hiire genoomis leiduvate geenide ja nendega külgnevate alade iseloomu; tabel 8, 2. veerg).

Tabel 8. Liikide ja vastavate annotatsioonikeskuste lõikes peale gtest.pl-i ja gtest100.pl-i tööd disainitud kvaliteetsete praimeripaaride suhe totaalsesse amplikonide arvu.

LIIK	PEALE ESIMEST PRAIMERIDISAINI RINGI	PEALE TEIST PRAIMERIDISAINI RINGI
*H.sapiens**	96.5/97.0/97.4	97.5/98/98.5
*M.musculus**	96.8/96.7/94.5	97.9/97.6/96
*R.norvergicus***	97.3/97.9/	98.5/98.6
S.cerevisiae	97.6	97.8
S.pombe	97.6	97.9

*ENSEMBL/NCBI/VEGA
**ENSEMBL/NCBI

Ebakvaliteetseid praimereid omanud amplikonidele disainiti sada uut praimeripaari primers100.pl abil. Kõigile programmi primers100.pl sisendiks läinud amplikonidele leiti vähemalt üks praimeripaar ja ülimalt 100 praimeripaari, mille seondumiskohtade arvu kontrolliti üle genoomi programmi gtest100.pl-iga. Tabelid 9 ja 10 annab ülevaate disainitud kvaliteetsete praimeripaaride arvust vastavalt peale gtest.pl ja gtest100.pl tööd.

Tabel 9. Disainitud praimeripaaride arvud võrreldes totaalsete amplikonide arvudega peale `gtest.pl`

Liik	Amplikonide arv	kvaliteetsete praimeripaaride koguarv peale gtest.pl-i tööd
*H.sapiens**	108,850/97,230/21,508	105,073/94,221/20,958
*M.musculus**	98,126/77,499/705	94,55/74,993/666
*R.norvergicus***	81,281/23,710	79120/23209
S.cerevisiae	4,609	4,498
S.pombe	5,657	5,520

*ENSEMBL/NCBI/VEGA

**ENSEMBL/NCBI

Tabel 10. Disainitud praimeripaaride arvud võrreldes totaalsete amplikonide arvudega peale `gtest100.pl` tööd.

Liik	Amplikonide arv	kvaliteetsete praimeripaaride koguarv peale gtest100.pl-i tööd
*H.sapiens**	108,850/97,230/21,508	106,088/95,116/21,185
*M.musculus**	98,126/77,499/705	96,041/75,650/676
*R.norvergicus***	81,281/23,710	80,041/23,380
S.cerevisiae	4,609	4,509
S.pombe	5,657	5,537

*ENSEMBL/NCBI/VEGA
**ENSEMBL/NCBI

Peale `gtest100.pl`-i tööd oli, sõltuvalt organismist ja talle vastavast annotatsioonikeskusest polnud suudetud disainida 1.4% - 2.5% (tabel 11) kõigist disainitud amplikonidest neile vastavaid praimeripaare. Statistika tegemisel ei arvestatud organismi *Mus musculus* VEGA annotatsioonikeskuse andmeid, kuna need olid statistiliselt eba-informatiivsed.

Tabel 11. Amplikonide, millele praimereid ei suudetud disainida suhe totaalsesse amplikonide arvu liikide ja neile vastavate annotatsioonikeskuste lõikes.

Liik	Disainimata praimeritega amplikonide suhe amplikonide koguarvu (%)
*H.sapiens**	2.5/2.0/1.5
*M.musculus**	2.1/2.4/4.0
*R.norvergicus***	1.5/1.4
S.cerevisiae	2.2
S.pombe	2.0

*ENSEMBL/NCBI/VEGA

**ENSEMBL/NCBI

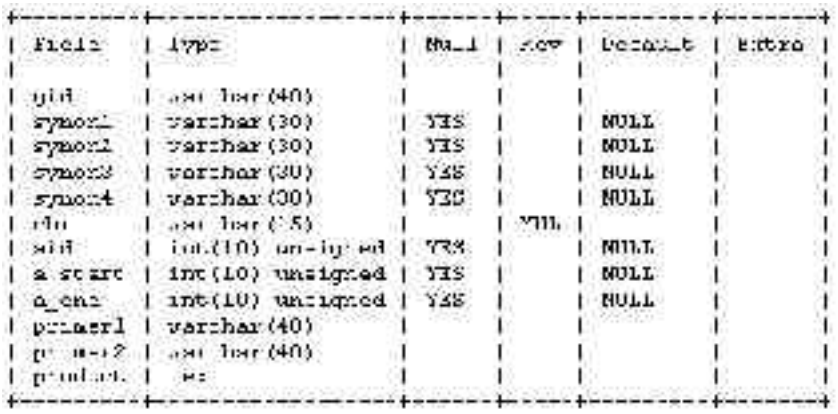
Analüüsisime inimese ENSEMBL-i annotatsiooniandmetel põhinevaid amplikone, millele ei suudetud praimereid leida. Vaatasime, mitu erinevat geeni kuulusid kõikidesse nendesse amplikonidesse, millele ei suudetud praimereid disainida ning, mitmele geenile genoomist pole õnnestunud ühtegi vastavat amplikoni üles-amplifitseeritavat praimeripaari disainida. Inimese annotatsioonikeskuse ENSEMBL korral oli meil teada 21,787 geeni andmed, meil ei õnnestunud 1418-le geenile ehk 6.5%-le kõikidest geenidest vähemalt ühele, seda geeni iseloomustavale amplikonile praimereid disainida. Geene, mida kirjeldavate amplikonide üles-amplifitseerimiseks ei õnnestunud ühtegi praimeripaari disainida oli 646 ehk 45.6% sellistest geenidest, mille üles-amplifitseerimiseks vähemalt ühele amplikonile praimeripaari ei suudetud disainida. Geenid, mille üles-amplifitseerimiseks pole ühtegi praimeripaari õnnestunud disainida on üle 95%-i ühe-eksonilised geenid. Geenide koguarvust moodustavad geenid, mille üles-amplifitseerimiseks pole ühtegi praimeripaari ca 3.0%.

4. Praimerite portaali ja vastavate andmebaasi(de) loomine

Selleks, et ka peale praimerite disainijate keegi saaks praimereid kasutada, lõime võrguvahendusel kasutatava praimerite portaali. Portaali nimeks panime *PrimerParadise*. Portaali realiseerimiseks valisime kasutatavateks tehnoloogiateks veebi-programmeerimisekeele PHP, andmebaasisüsteemi MySQL, kliendipoolse skriptimiskeele JavaScript ja veebilehe välimuse kujundamiseks CSS (*Cascading Style Sheets*)

tehnoloogia. Lisaks eukarüootide genoomidele disainitud praimeritele, sisaldab portaal ka prokarüootsete organismide geenidele disainitud praimereid ning eukarüootsete organismide genoomis sisalduvate SNP-de paljundamiseks vajalikke praimereid.

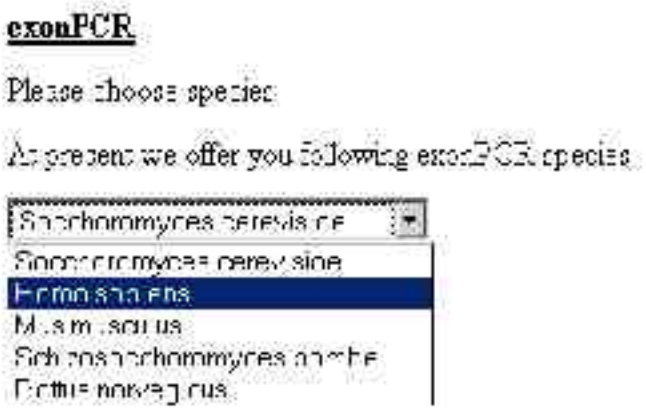
Portaali tegemiseks lõime neli andmebaasi – primerPCR, euPCR, proPCR ja snpPCR. Esimene andmebaas primerPCR sisaldab tabeleid PHP-lehe töö jaoks (kasutaja lehel navigeerimise jaoks). ProPCR ja snpPCR on analoogilise struktuuriga kui euPCR ja sisaldavad vastavalt prokarüootide geenidele ja eukarüootsete organismide SNP-dele disainitud praimereid. EuPCR, mis sisaldab eukarüootsete organismide geenide üles-amplifitseerimiseks praimereid sisaldab kahte tüüpi tabeleid. Ühed, mis on vajalikud veebi-programmeerimiseks ja teised, mis sisaldavad iga organismi ning talle vastava annotatsioonikeskuse lõikes käesoleva töö raames disainitud praimereid. Tabeli nimed on pandud vastavalt antud liigile, annotatsioonikeskusele ja kasutatavale genoomi versioonile. Näiteks inimese 34-ndat genoomi versiooni ja ENSEMBL annotatsioonikeskuse andmeid kasutades lõime praimeritabeli *human_ensembl_34*. Praimereid sisaldava tabeli struktuur on toodud joonisel 10.



Joonis 10. Disainitud praimereid sisaldav tabel. *gid* – antud amplikonis sisalduva vastava annotatsioonikeskuse geeni süstemaatiline nimi; *synon(1-4)* – *gid* identifikaatoriga geenile vastavad sünonüümsed geeninimed; *chr* – kromosoomi nimi; *aid* – amplikoni, kuhu vastav geen kuulub, identifikaator; *a_start* ja *a_end* – vastava amplikoni algus- ning lõppkoordinaadid genoomil; *primer1 ja primer2* – vastavalt + ja – ahela praimer; product –

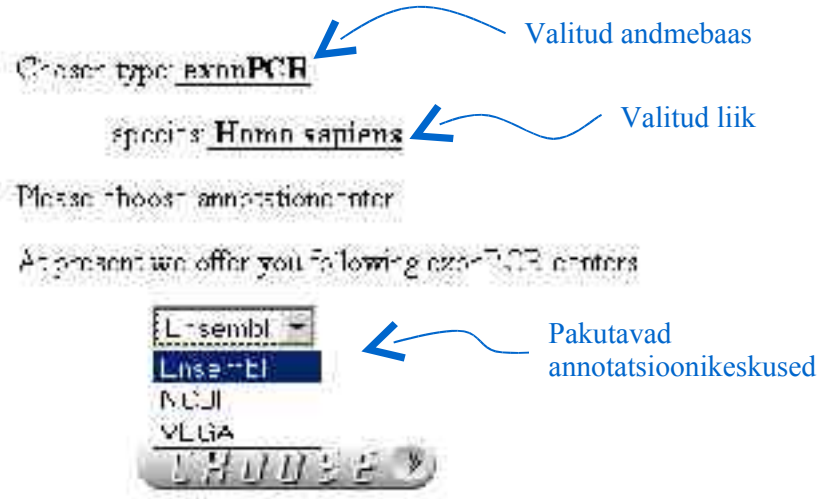
PCR-i produkt. Kromosoomi nimi ning amplikoni identifikaator määravad üheselt ära praimeripaari.

Portaal asub aadressil *http://kobra.ebc.ee/exonPCR/primerPCR* ja on inglise keeles, kuna on mõeldud internatsionaalseks kasutamiseks. Sisenedes lehele kuvatakse avaleht, kust kasutaja saab valida kolme praimerikollektsiooni vahel – SNPPCR, EUPCR, PROPCR (vastavalt andmebaasi nimedele). Valides lingi 'EUPCR' avaneb kasutajale alamleht, kust ta saab valida liigi (joonis 11)



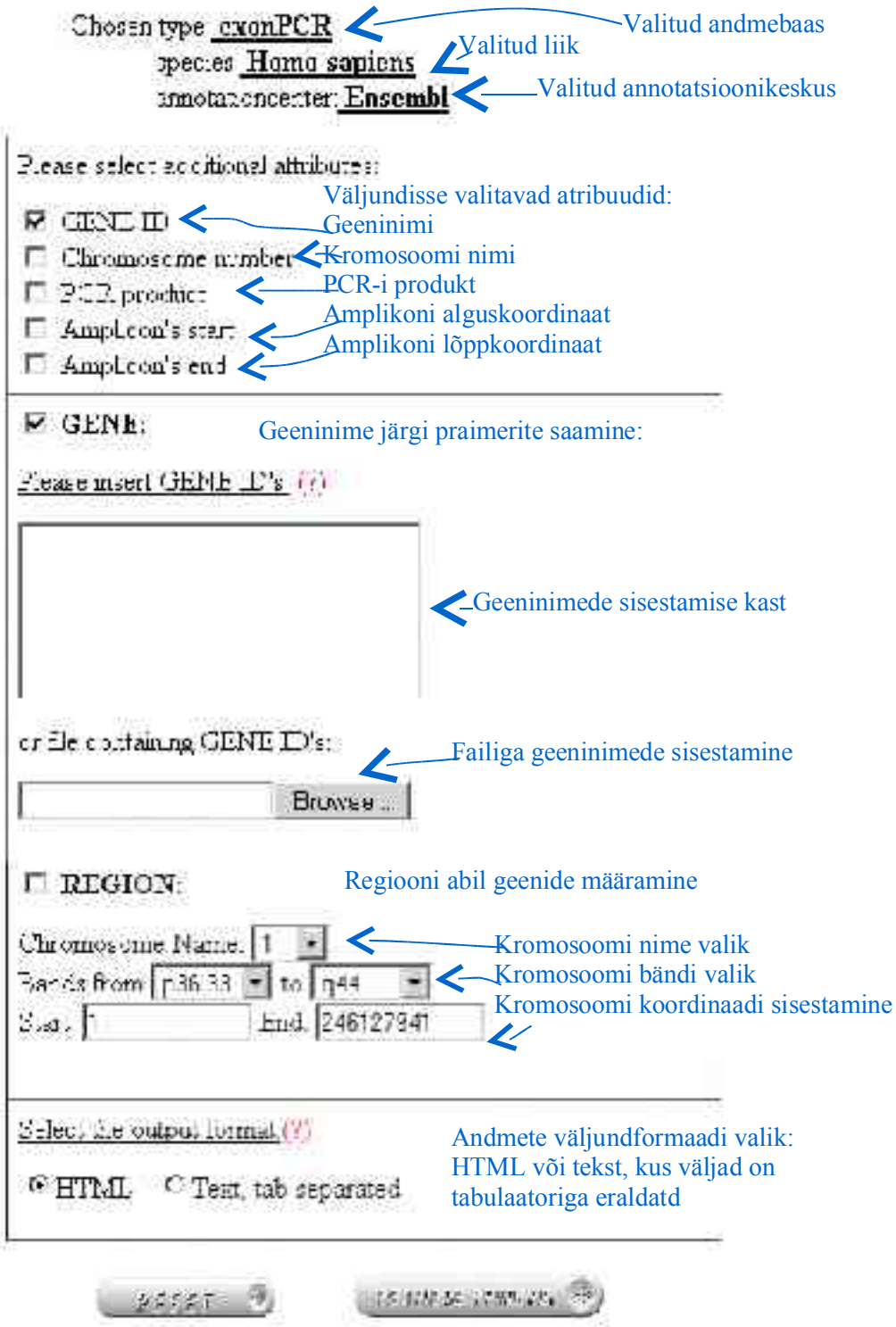
Joonis 11. Kasutajal on võimalik valida 5 liigi vahel – *Homo sapiens, Mus musculus, Rattus norvegicus, Saccharomyces cerevisiae, Schizosaccharomyces pombe.*

Valides liigi avaneb kasutajale järgmine aken, kust on võimalik valida, mis annotatsioonikeskuses annoteeritud geenidele soovitakse praimereid (joonis 12).



Joonis 12. Valides liigiks inimese on meil võimalik valida kolme annotatsioonikeskuse vahel – ESEMBL, NCBI ja VEGA.

Peale annotatsioonikeskuse valimist, kuvatakse veebilehitsejasse alamleht, kust kasutaja saab valida väljundisse tulevad väljad (joonis 13). Valikus on geeninimi, kromosoomi nimi, PCR-i produkt, amplikoni algus- ning lõppkoordinaat. Samuti on võimalik valida, kas tahetakse kasutajat huvitava geeni paljundamiseks vajalikke praimereid saada geeninime või regiooni järgi, kus geen lokaliseerub. Geeninimesid on võimalik sisestada ka kasutades faili, kus on huvipakkuvad nimed eraldatud üksteisest tühikute või reavahedega. Regiooni on võimalik valida, kas kromosoomi koordinaatide või bändide järgi. Bändide valimise võimalus kuvatakse vaid selliste organismide korral (EUPCR-is inimene, hiir ja rott), kelle puhul oli saadaval info bändide kohta. Viimaks on võimalik valida väljundformaat. Valiku all on HTML struktuuriga leht või tabulaatoriga eraldatud tekst (tekstifail) (joonis 13). Väljundisse kuvatakse päritud andmed vastavalt kasutaja soovile, kas HTML-lehena või lingina tabuleeritud failile. Juhul kui kasutaja päringule vastab üle saja kirje ja kasutaja soovis näha andmeid HTML-lehel, siis kuvatakse 100 esimest kirjet HTML-formaadis ja kogu päringu vastus kirjutatakse tabuleeritud faili (HTML-lehele ilmub link vastavale failile). Viimast pidasime vajalikuks sellepärast, et kui kasutaja päringule vastab ca sada praimeripaari või rohkem, siis tal ei ole HTML-formaadis neid enam mugav (või võimalik) analüüsida ning ta saab tabuleeritud faili konventeerida talle mugavamalt analüüsitavasse formaati (nt Exceli failiks).



Joonis 13. Kasutajal on võimalik valida väljundandmed, sisestada, kas geeni identifikaator või huvipakkuv kromosoomi regioon määramaks geeni ning väljundformaat.

Väljundis näidatakse ära kasutaja poolt küsitud väljade nimed, millal viimati tabelit, kust praimerid päriti muudeti, totaalne leitud praimerite arv ning küsitud väljade kohta käivad andmed (geeninimi, praimeripaar jt).

Portaali on testitud erinevate veebilehitsejatega. Täisfunktsionaalsusega töötab järgnevates brauserites: Mozilla (seeria 1), Opera (seeria 5, 6, 7), Internet Explorer (seeria 5 ja 6), Netscape Navigator (platvorm 6 ja 7).

5. Programmide tööaeg, mäluvajadus ja kettavajadus

Programmide tööaeg sõltus oluliselt vaatluse all oleva organismi genoomi suurusest ja seal sisalduvate geenide hulgast. Mida väiksem oli genoom (vähem aega kulus üle genoomi praimerite testimise peale) ja vähem disainitud amplikone (vähem kulus aega praimerite disainimiseks), seda lühem oli praimeri disainis kasutatud programmide tööaeg. Siinkohal tuuakse kõige rohkem aega ja mälu võtnud organismi (*Homo sapiens,* annotatsioonikeskus ENSEMBL) praimerite disainil kasutatud programmide tööks kulunud aegade jaotus (tabel 12) ja CPU kasutus, mälu vajadus ning kettakasutus (tabel 13). Kõige ajakulukamateks etappideks praimerite disainil osutusid ühe praimeripaari (`primers.pl`) ja 100 praimeripaari (`primers100.pl`) disainimine amplikonidele. Kõigi organismide korral võttis `primers100.pl` töö rohkem aega kui `primers.pl`, kuna esimese programmi korral proovitakse igas tsüklis disainida ühe amplikoni paljundamiseks 100 praimeripaari. Portaalist päringu tegemise kiirus sõltub mitmetest asjaoludest. Oluline on interneti ühenduse kiirus, kuid samuti kasutaja poolt sooritatud päringu keerukus, st kui palju ja missuguseid välju soovitakse väljundis näha, kas kasutatakse praimerite saamiseks geeninimesid või sisestas regiooni, kui palju geeninimesid kasutati või, kui pikk regioon sisestati, kas sooviti väljundit näha HTML-ina või tabuleeritud tekstina jpm.

Sooritasime portaalist järgneva päringu: praimerite kollektsioon EUPCR, liik *Homo sapiens,* annotatsioonikeskus ENSEMBL, väljundisse soovisime saada kõiki erinevaid pakutavaid välju (geeninimi, praimerid, PCR-i produkt, amplikoni algus- ja

lõppkoordinaat, kromosoomi nimi), tahtsime saada kõikide esimeses kromosoomis sisalduvate geenide üles-amplifitseerimiseks praimereid, väljundit soovisime tabuleeritud tekstina. Tulemuseks saime väljundisse 10,438 kirjet tabuleeritud tekstina. Selline päring võttis aega 106 minutit. Selline pikk aeg on tingitud tõenäoliselt mahukast andmebaasi päringust. Mäluvajadus antud protsessi sooritamiseks oli 2.71 Mb ja kettakasutus ca 15 Mb. Võrdluseks, sooritades eelpool kirjeldatud päringu, kuid regiooni asemel sisestame ühe geeni identifikaatori, siis päring andis vastuseks kaks kirjet, aega võttis 3 sekundit, mäluvajadus on samuti 2.71 Mb ja kettakasutus on ca 6 kb.

Tabel 12. Inimese ENSEMBL-i poolt annoteeritud geenidele praimeridisainil kasutatud programmide maksimaalsed tööajad.

Programmi nimi	Tööaeg (h)
amplicons.pl	0.0083
primers.pl	3.10
gtest.pl	1.75
primers100.pl	5.60
gtest100.pl	4.20
Totaalne	14.66

Praimerite genereerimisel on üheks mälunõudlikumaks programmiks järjestuse maskeerimist teostav GMASKER (tabel 13), kuna jätab meelde genoomis, kas rohkem kui 10 või 30 korda esinevad 16 või 12 nukleotiidi pikkused sõnad. Kõige mälunõudlikum etapp praimeri disaini juures on praimerite sekundaarsete seostumiskohtade kontroll genoomis, mida teostab programm GTESTER. GTESTER vajab palju mälu, kuna loeb testimisel kettalt mällu kromosoomide paisktabelite moodustamisel genereeritud indeksid. Kettavajadus on kõige kõrgem GTESTER programmil, viimast samuti genoomist moodustatud paisktabeli indeksite tõttu.

Tabel 13. Kõige rohkem ressurssi nõudnud organismi (*Homo sapiens,* ENSEMBL) genoomis sisalduvatele geenidele praimerite disainimise protsessi erinevate etappide CPU- ja mälukasutus ning kettavajadus.

Programmi nimi	**käsk, toiming**	**CPU%**	**mälu-kasutus(Mb)**	**ketta-vajadus (Mb)**
amplicons.pl	amplikonide genereerimine	10.1	6.2	19
primers.pl/ primers100.pl	andmebaasist amplikoni järjestuse tõmbamine	13.3	24.9	148
	DUST_LOWER	99.7	0.6	302
	GMASKER	99.1	131.0	429
	SNP-de maskeerimine	28.2	31.0	292
	GM_PRIMER3	98.7	6.2	381
gtest.pl/ gtest100.pl	GTESTER	51.5	348.0	22,784
Portaalist päringu sooritamine	mysqld	99.4	18.6	ca päringu andmete maht

Arutelu

Valitud viie eukarüootse organismi kõigist disainitud amplikonidest on võimalik disainida praimereid vähemalt 97.5%-le. Sõltuvalt organismist jäid 1.4 - 2.5 %-le disainitud amplikonidest praimerid disainimata. Analüüsides tulemusi, võime öelda, et ühte ja sama metoodikat kasutades saab erinevate eukarüootsete genoomide geenide üles-amplifitseerimiseks disainida praimereid sama efektiivselt.

Mõnevõrra erinevat lähenemist tuleb siiski kasutada natuke väiksemate eukarüootsete genoomidega (*S. pombe* ja *S. cerevisiae*) töötamisel, kuna nende genoomid ei sisalda nii palju korduvat DNA-d kui suure-genoomsed organismid (*H.sapiens, M.musculus, R.norvergicus*). Praimerite disainil amplikonide eel-maskeerimisel kasutasime GMASKER programmi, viimane kasutab kogu genoomis rohkem kui teatud arv (antud olukorras 10 või 30) korda leiduvate N-nukleotiidipikkuste motiivide nimekirja. Inimese, hiire ja roti korral kasutasime motiivi pikkust 16 nukleotiidi, kuna see on maskeerimise seisukohast efektiivseim suurus (Andreson, unpublished). Kuna pärmide *S.cerevisiae* ja *S.pombe* korral 16-nukleotiidilisi korduvaid motiive genoomis on väga vähe, ning seepärast amplikonide maskeerimine ei tõsta disainitavate PCR-i praimerite kvaliteeti, siis kasutasime pärmide korral maskeerimiseks ‘musta’ nimekirja, mis sisaldas 12-nukleotiidi pikkuseid kordusjärjestusi.

Analüüsides amplikonide järjestusi, millele ei õnnestunud praimereid disainida, oli näha GMASKER programmi poolt tugevat maskeeringut, vähem oli amplikonide praimerite disainimiskohtades sekveneerimata järjestust (alla 1% sellistest amplikonidest, millele ei suudetud praimereid disainida). GM_PRIMER3-e poolt ei leitud praimereid ka sellepärast, et praimerite sulamistemperatuurid ja GC-nukleotiidide sisaldus olid paljudel kordadel lubatust kõrgemad ja sulamistemperatuur oli paljudel juhtudel ka lubatust madalam ning rohkesti oli probleeme kahe praimeri (+ ja – ahela) lubatust kõrgema sulamistemperatuuride erinevusega.

Kuna suurtes genoomides leidub palju korduvat nukleotiidset järjestust ning keerulise struktuuriga regioone, siis leidub alasid genoomis, mille kummalegi poole 1,200

nukleotiidi pikkusele alale polegi võimalik disainida meie poolt kasutatud parameetri väärtustega üle genoomi unikaalseid praimereid. Et siiski oleks võimalik genoomist kõik kodeerivad alad üles-amplifitseerida, siis võiks tõsta `primers.pl` programmis realiseeritavat tsüklite arvu kuni kõigile amplikonidele on suudetud praimerid disainida. Sellisel juhul leiduks amplikone, mis oleksid liiga pikad, et neid konventsionaalse PCR-i abil üles-amplifitseerida. Lahenduseks oleks nt *long*-PCR-i kasutamine, mis võimaldab pikki DNA fragmente paljundada. Edasi võimaldaks *nested*-PCR juba spetsiifilisemate praimerite abil soovitud kodeeriva ala üles-amplifitseerida.

Programmide, mis disainivad praimereid (`primers.pl` ja `primers100.pl`), tööaegu oleks võimalik lühendada sellega, et enne praimerite disaini maskeerida kogu genoom programmidega DUST_LOWER ja GMASKER ning SNP-de kohalt. Edasi kasutada praimerite disainil juba saadud maskeeritud järjestust. Kogu genoomi vastav maskeerimine võtaks tõenäoliselt palju aega, kuid samas lühendaks see `primers.pl` ja `primers100.pl` programmide tööaegu, mis on oluline sama organismi samal genoomi versioonil, kuid erinevatel annotatsiooniandmetel põhinevate geenide amplifitseerimiseks disainitavate praimerite genereerimisel. Põhjus, miks pole tervete genoomide maskeerimist teostanud on genoomide suur maht, mis nõuab palju vaba kettaruumi olemasolu.

Praimerite portaali efektiivsemaks muutmisel on mitmeid ideid. Kindlasti on vaja praimerite andmebaasis olemasolevaid andmeid genoomi uute versioonide ilmumisel ja annotatsiooniandmete muutumisel uuendada. Samuti on oluline disainida ja lisada portaali erinevate eukarüootsete organismide geenide amplifitseerimiseks vajalikke praimereid. Tulevikus on plaanis ka lisada portaalile kasutajate sisse-logimise utiliit, nii et kasutajad, kes teist- või enama kordselt portaali keskkonda sisse logivad, võivad näha oma eelmise seansi korral sooritatud toiminguid. Selline sisse-logimise süsteem annab meile võimaluse portaali kasutajatelt rohkem tagasisidet saada. Sisse-logimise utiliit annaks ka portaali arendajatele ülevaate, sellest, mis organisme, genoomiregioone ja geene rohkem uuritakse. Samuti saaks küsida portaali külastajatelt, kes on eksperimentaalselt kasutanud meie poolt disainitud praimereid, tagasisidet selle kohta, kas meie metoodika abil on õnnestunud genereerida kvaliteetseid praimereid.

Kokkuvõte

Käesolevas töös töötati välja metoodika suuregenoomiliste eukarüootsete organismide kõigi kodeerivate alade amplifitseerimiseks vajalike praimerite disainimiseks. Disainimise protsess algas amplikonide, mida on võimalik konventsionaalse PCR-i abil paljundada, genereerimisega. Edasi liideti amplikonide algus- ja lõppkoordinaadile praimerite disainimiseks külgnev ala. Vältimaks praimerite disainimist kordusjärjestusse, maskeeriti amplikoni nukleotiidses järjestuses nii lihtsad kordusjärjestused kui ka keerulisemaid kordusjärjestusi sisaldavad regioonid, samuti maskeeriti järjestus SNP-ide asukohtadest. Maskeeritud amplikonile kummalegi poole külgnevasse alasse disainiti üks praimer. Tsükleid, mis algasid praimeri disainiks amplikoni algus- ja lõppkoordinaadile regioonide lisamisega, sooritati mitmeid kordi, nii et igas järgnevas tsüklis lisati amplikonile kaks korda pikem külgnev ala kui eelmises tsüklis. Tsükli kordamine on oluline, kuna ühe tsükliga lisatud külgnev ala pole paljudele amplikonidele vajalike praimerite disainimiseks piisav, sest võib sisaldada palju korduvat DNA-d või järjestust, millele pole võimalik lubatud PCR-i parameetrite väärtustega praimereid disainida. Seejärel kontrolliti saadud praimerite sekundaarsete seondumiskohtade olemasolu genoomis. Amplikonidele, millele oli disainitud praimerid, mis omasid genoomis rohkemat kui ühte seondumiskohta, disainiti asemele mitu uut praimeripaari kasutatdes eelpool kirjeldatud tsüklit. Saadud praimeritele teostati unikaalsuse kontroll üle genoomi ning praimerid, mis omasid tõenäosust rist-hübridiseerimiseks, kõrvaldati vaatluse alt.

Kirjeldatud metoodikaga disainisime praimerid viie eukarüootse organismi – *Homo sapiens, Mus musculus, Rattus norvergicus, Saccharomyces cerevisiae, Schizosaccharomyces pombe* kõigile geenidele. Keskmiselt 98%-le disainitud amplikonidest õnnestus genereerida genoomis unikaalset seostumiskohta omav praimeripaar. Disainitud praimeritele loodi andmebaas ning veebiliides nimega *PrimerParadise* (*http://kobra.ebc.ee/exonPCR/primerPCR*) praimerite interaktiivseks kättesaamiseks.

Summary

There has been developed a strategy for PCR primer design for the large-scale amplification of genes from genomic sequence of eucaryotic organisms. The process started with generating the amplicons suitable for the amplification with conventional PCR. Next we added sequence to the start and the end of the amplicon for the primer design. Preventing design of primers to repeated regions we premasked amplicons at low-complexity regions and at longer over-represented words in genomic sequence. Thereafter we masked amplicons from SNP positions. To the both sides of the masked amplicons we designed one primer. Cycle which started with adding sequence to both sides of each amplicon for primer design was repeated many times. Latter is important because there could not be designed primers with good quality to the region added to amplicons with one cycle. The cross-hybridization of primers was tested. With the cycle described before we designed new primerpairs to the amplicons with primers having secondary binding-sites in target genome. Next we checked again the secondary binding sites of primerpairs over the genome and primers having more than one binding site were rejected.

Method described was used for primer design to all genes of five eucaryotic organisms - *Homo sapiens, Mus musculus, Rattus norvergicus, Saccharomyces cerevisiae, Schizosaccharomyces pombe.* Average 98% of all designed amplicons were retrieved primers with quality. To retrieve PCR primers online web based graphical user interface called *PrimerParadise* was made (`http://kobra.ebc.ee/exonPCR/primerPCR`).

Kasutatud kirjandus.

1. Albertson Donna G, Colin Collins, Frank McCormick, Joe W Gray. Chromosome aberrations in solid tumors. *Nature Genetics* 34, 369 - 376 (01 Aug 2003)
2. Albertson DG, Pinkel D. Genomic microarrays in human genetic disease and cancer. Hum Mol Genet. 2003 Oct 15;12 Spec No 2:R145-52. Epub 2003 Aug 05.
3. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol. 1990 Oct 5;215(3):403-10.
4. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 1997 Sep 1;25(17):3389-402.
5. Ashburner M, Misra S, Roote J, Lewis SE, Blazej R, Davis T, Doyle C, Galle R., et al. An exploration of the sequence of a 2.9-Mb region of the genome of Drosophila melanogaster: the Adh region. Genetics. 1999 Sep;153(1):179-219.
6. Baldi Pierre, Brunak Søren. Bioinformatics: The Machine Learning Approach, second edition. Cambridge, 2001.
7. Baldino F Jr, Chesselet MF, Lewis ME. High-resolution in situ hybridization histochemistry. Methods Enzymol. 1989;168:761-77.
8. Batzoglou S., Lior Pachter, Jill P. Mesirov, Bonnie Berger and Eric S. Lander. Human and Mouse Gene Structure: Comparative Analysis and Application to Exon Prediction. Genome Res. 2000 Jul;10(7):950-8.
9. Benson Dennis A. et al. GenBank: update. Nucleic Acid Res. 2004, Vol. 32.
10. Birney E., Andrews D., Bevan P. et al. Ensembl 2004. Nucleic Acids Res. 2004 Jan 1;32
11. Birney E, Durbin R. Using GeneWise in the Drosophila annotation experiment. Genome Res. 2000 Apr;10(4):547-8.
12. Borer PN, Dengler B, Tinoco I Jr, Uhlenbeck OC. Stability of ribonucleic acid double-stranded helices. J Mol Biol. 1974 Jul 15;86(4):843-53.
13. Brendel V, Xing L, Zhu W. Gene structure prediction from consensus spliced alignment of multiple ESTs matcing the same genomic locus. Bioinformatics 2004, Feb 24

14. Breslauer KJ, Frank R, Blocker H, Marky LA. Predicting DNA duplex stability from the base sequence. Proc Natl Acad Sci U S A. 1986 Jun;83(11):3746-50.
15. Brightwell G, Wycherley R, Waghorn A. SNP genotyping using a simple and rapid single-tube modification of ARMS illustrated by analysis of 6 SNPs in a population of males with FRAXA repeat expansions. Mol Cell Probes. 2002 Aug;16(4):297-305.
16. Brooksbank Catherine, Camon Evelyn, Harris Midori A. et al. Nucleic Acid Res 2003, Vol 31.
17. Burge Christopher B. Chpping away the transcriptome. Nature Genetics, 2001, Vol 27.
18. Burge C, Karlin S. Prediction of complete gene structures in human genomic DNA. J Mol Biol. 1997 Apr 25;268(1):78-94.
19. Collins JE, Goward ME, Cole CG, Smink LJ, Huckle EJ, Knowles S, Bye JM, Beare DM, Dunham I. Reevaluating human gene annotation: a second-generation analysis of chromosome 22. Genome Res. 2003 Jan;13(1):27-36.
20. Curwen V, Eyras E, Andrews TD, Clarke L, Mongin E, Searle SM, Clamp M. The Ensembl automatic gene annotation system. Genome Res. 2004 May;14(5):942-50.
21. Delcher A.L., S. Kasif, R.D. Fleischmann, J. Peterson, O. White, and S.L. Salzberg. Alignment of Whole Genomes. Nucleic Acids Research, 27:11 (1999), 2369-2376.
22. Delcher A.L., A. Phillippy, J. Carlton, and S.L. Salzberg. Fast Algorithms for Large-scale Genome Alignment and Comparison. Nucleic Acids Research (2002), Vol. 30, No. 11 2478-2483.
23. Duret L, Mouchiroud D, Gautier C. Statistical analysis of vertebrate sequences reveals that long genes are scarce in GC-rich isochores. J Mol Evol. 1995 Mar;40(3):308-17.
24. Dunham I, Shimizu N, Roe BA, Chissoe S, Hunt AR, Collins JE, Bruskiewich R., et al. The DNA sequence of human chromosome 22. Nature. 1999 Dec 2;402(6761):489-95.
25. Fickett JW, Guigo R. Estimation of protein coding density in a corpus of DNA sequence data. Nucleic Acids Res. 1993 Jun 25;21(12):2837-44.
26. Fickett JW, Tung CS. Assessment of protein coding measures. Nucleic Acids Res. 1992 Dec 25;20(24):6441-50.
27. Florea L, Hartzell G, Zhang Z, Rubin GM, Miller W. A computer program for aligning a cDNA sequence with a genomic DNA sequence. Genome Res. 1998 Sep;8(9):967-74.

28. Galperin Michael Y. The Molecular Biology Database Collection: 2004 update. Nucleic Acids Res. 2004, Vol. 32.
29. Garcia-Blanco MA, Baraniak AP, Lasda EL. Alternative splicing in disease and therapy. Nat Biotechnol. 2004 May;22(5):535-546.
30. Gelfand MS, Mironov AA, Pevzner PA. Gene recognition via spliced sequence alignment. Proc Natl Acad Sci U S A. 1996 Aug 20;93(17):9061-6.
31. Gibbs RA, Weinstock GM, Metzker ML, Muzny DM, Sodergren EJ, Scherer S et al. Genome sequence of the Brown Norway rat yields insights into mammalian evolution. Nature. 2004 Apr 1;428(6982):493-521.
32. Gusfield Dan. Algorithms on strings, trees and sequenses. Computer science and computational biology. University of California, Cambridge University Press. 1997.
33. Healy J, Thomas EE, Schwartz JT, Wigler M. Annotating large genomes with exact word matches. Genome Res. 2003 Oct;13(10):2306-15. Epub 2003 Sep 15.
34. Hillier L, Green P. OSP: a computer program for choosing PCR and DNA sequencing primers. PCR Methods Appl. 1991 Nov;1(2):124-8.
35. Huang X, Adams MD, Zhou H, Kerlavage AR. A tool for analyzing and annotating genomic sequences. Genomics. 1997 Nov 15;46(1):37-45.
36. Hu Y, Tanzer LR, Cao J, Geringer CD, Moore RE. Use of long RT-PCR to characterize splice variant mRNAs. Biotechniques. 1998 Aug;25(2):224-9.
37. Hubbard T, Barker D, Birney E, Cameron G, Chen Y et al. The Ensembl genome database project. Nucleic Acids Res. 2002 Jan 1;30(1):38-41.
38. Höhl Michael, Stefan Kurtz, and Enno Ohlebusch. *Efficient multiple genome alignment*. Bioinformatics 2002 18: 312S-320S.
39. Jiang J, Jacob HJ. EbEST: an automated tool using expressed sequence tags to delineate gene structure. Genome Res. 1998 Mar;8(3):268-75.
40. Jones J, Field JK, Risk A Comparative guide to gene prediction tools for the bioinformatics amateur. RM. Int J Oncol, 2002, Vol 20.
41. Kamel A, Abd-Elsalam. Bioinformatic tools and guideline for PCR primer design. *African Journal of Biotechnology* 2003 2(5):91-95.

42. Kan Z, Rouchka EC, Gish WR, States DJ. Gene structure prediction and alternative splicing analysis using genomically aligned ESTs. Genome Res. 2001 May;11(5):889-900
43. Karolchik, D., Baertsch, R., Diekhans, M., Furey, T.S., Hinrichs, A., Lu, Y.T., Roskin, K.M., Schwartz, M., Sugnet, C.W., Thomas, D.J., Weber, R.J., Haussler, D. and Kent, W.J. The UCSC Genome Browser Database. 2003. *Nucl. Acids Res.* 31:51-54.
44. Karolchik, D., Hinrichs, A.S., Furey, T.S., Roskin, K.M., Sugnet, C.W., Haussler, D. and Kent, W.J. The UCSC Table Browser data retrieval tool. 2004 *Nucl. Acids Res.* 32 (90001):D493-D496.
45. Kent WJ. BLAT--the BLAST-like alignment tool. Genome Res. 2002 Apr;12(4):656-64.
46. Kent W. James, Sugnet Charles W., Furey Terrence S., et al. The Human Genome Browser at UCSC. Genome Res. 2002 Jun;12(6):996-1006.
47. Kopsidas G, Kovalenko SA, Islam MM, Gingold EB, Linnane AW. Preferential amplification is minimised in long-PCR systems. Mutat Res. 2000 Nov 30;456(1-2):83-8.
48. Krane Dan E., Michael L. Raymer. Fundamental Concepts of Bioinformatics. ISBN: 0805346333 Publisher: Benjamin Cummings. Published: 09/12/2002.
49. Krogh A. Two methods for improving performance of an HMM and their application for gene finding. Proc Int Conf Intell Syst Mol Biol. 1997;5:179-86.
50. Kulikova Tamara, Aldebert Philippe, Althorpe Nicola et al. The EMBL Nucleotide Sequence Database. Nucleic Acid Res. 2004, Vol. 32.
51. Kurtz Stefan. Reducing the space requirement of suffix trees. Softw. Pract. Exper., 29 (13), 1149–1171 (1999).
52. Kurtz S and Schleiermacher C. REPuter: fast computation of maximal repeats in complete genomes. Bioinformatics 1999 15: 426-427.
53. Kämpke T, Kieninger M, Mecklenburg M. Efficient primer design algorithms. Bioinformatics. 2001 Mar;17(3):214-25.
54. Lexa M, Valle G. PRIMEX: rapid identification of oligonucleotide matches in whole genomes. Bioinformatics. 2003 Dec 12;19(18):2486-8.

55. Li P, Kupfer KC, Davies CJ, Burbee D, Evans GA, Garner HR. PRIMO: A primer design program that applies base quality statistics for automated large-scale DNA sequencing. Genomics. 1997 Mar 15;40(3):476-85.
56. Lipman DJ, Pearson WR. Rapid and sensitive protein similarity searches. Science. 1985 Mar 22;227(4693):1435-41.
57. Lukashin AV, Borodovsky M. GeneMark.hmm: new solutions for gene finding. Nucleic Acids Res. 1998 Feb 15;26(4):1107-15.
58. Macara IG, Baldarelli R, Field CM, Glotzer M et al. Mammalian septins nomenclature. Mol Biol Cell. 2002;13(12):4111-3.
59. Mackiewicz P, Kowalczuk M, Mackiewicz D, Nowicka A, Dudkiewicz M, Laszkiewicz A, Dudek MR, Cebrat S. How many protein-coding genes are there in the Saccharomyces cerevisiae genome? Yeast. 2002 May;19(7):619-29.
60. Makalowski W. The human genome structure and organization. Acta Biochim Pol. 2001;48(3):587-98.
61. Mathé C, Sagot MF, Schiex T, Rouze P. Current methods of gene prediction, their strengths and weaknesses. Nucleic Acids Res. 2002 Oct 1;30(19):4103-17.
62. Mhlanga MM, Malmberg L. Using molecular beacons to detect single-nucleotide polymorphisms with real-time PCR. Methods. 2001 Dec;25(4):463-71.
63. Mironov AA, Fickett JW, Gelfand MS. Frequent alternative splicing of human genes. Genome Res. 1999 Dec;9(12):1288-93.
64. Mott R. EST_GENOME: a program to align spliced DNA sequences to unspliced genomic DNA. Comput Appl Biosci. 1997 Aug;13(4):477-8.
65. Nadershahi A, Fahrenkrug SC, Ellis LB. Comparison of computational methods for identifying translation initiation sites in EST data. BMC Bioinformatics. 2004 Feb 16;5 (1):14.
66. Nelson Mark. Data Compression with the Burrows-Wheeler Transform. Dr. Dobbs' J., Sept. 1996.
67. Ning Z, Cox AJ, Mullikin JC. SSAHA: a fast search method for large DNA databases. Genome Res. 2001 Oct;11(10):1725-9.

68. Owczarzy R, Vallone PM, Gallo FJ, Paner TM, Lane MJ, Benight AS. Predicting sequence-dependent melting stability of short duplex DNA oligomers. Biopolymers. 1997;44(3):217-39.
69. Parra G, Agarwal P, Abril JF, Wiehe T, Fickett JW, Guigo R. Comparative gene prediction in human and mouse. Genome Res. 2003 Jan;13(1):108-17.
70. Pruitt KD, Katz KS, Sicotte H, Maglott DR. Introducing RefSeq and LocusLink: curated human genome resources at the NCBI. Trends Genet. 2000 Jan;16(1):44-47.
71. Pruitt KD, Maglott DR. RefSeq and LocusLink: NCBI gene-centered resources. Nucleic Acids Res 2001 Jan 1;29(1):137-140
72. Roest Crollius H, Jaillon O, Bernot A, Dasilva C, Bouneau L, Fischer C, Fizames C, Wincker P, Brottier P, Quetier F, Saurin W, Weissenbach J. Estimate of human gene number provided by genome-wide analysis using Tetraodon nigroviridis DNA sequence. Nat Genet. 2000 Jun;25(2):235-8.
73. Rogic S, Mackworth AK, Ouellette FB. Evaluation of gene-finding programs on mammalian sequences.Genome Res. 2001 May;11(5):817-32.
74. Rogic S, Ouellette BF, Mackworth AK. Improving gene recognition accuracy by combining predictions from two gene-finding programs. Bioinformatics. 2002 Aug;18 (8):1034-45.
75. Rozen S, Skaletsky H. Primer3 on the WWW for general users and for biologist programmers. Methods Mol Biol. 2000;132:365-86.
76. Rubin E, Levy AA. A mathematical model and a computerized simulation of PCR using complex templates. Nucleic Acids Res. 1996 Sep 15;24(18):3538-45.
77. Rychlik W, Spencer WJ, Rhoads RE. Optimization of the annealing temperature for DNA amplification in vitro. Nucleic Acids Res. 1990 Nov 11;18(21):6409-12.
78. Schmutz J, Grimwood J, Myers RM. Assembly of DNA sequencing data. Methods Mol Biol. 2004;255:319-32.
79. Sheffield VC, Cox DR, Lerman LS, Myers RM. Attachment of a 40-base-pair G + C-rich sequence (GC-clamp) to genomic DNA fragments by the polymerase chain reaction results in improved detection of single-base changes. Proc Natl Acad Sci U S A. 1989 Jan;86(1):232-6.

80. Simsek M, Adnan H. Effect of single mismatches at 3'-end of primers on polymerase chain reaction. Medical Sciences (2000), 2, 11-14
81. Sipiczki, M. Where does fission yeast sit on the tree of life? *Genome Biol.* **1**, 1011.1-1011.4 (2000).
82. Smirnov DA, Burdick JT, Morley M, Cheung VG. Method for manufacturing whole-genome microarrays by rolling circle amplification. Genes Chromosomes Cancer. 2004 May;40(1):72-7.
83. Snyder EE, Stormo GD. Identification of protein coding regions in genomic DNA. J Mol Biol. 1995 Apr 21;248(1):1-18.
84. Syvanen AC. Accessing genetic variation: genotyping single nucleotide polymorphisms. Nat Rev Genet. 2001 Dec;2(12):930-42.
85. Wheelan SJ, Church DM, Ostell JM. Spidey: a tool for mRNA-to-genomic alignments. Genome Res. 2001 Nov;11(11):1952-7.
86. Wilhelm J, Pingoud A. Real-time polymerase chain reaction. Chembiochem. 2003 Nov 7;4(11):1120-8.
87. Williams M, Rainville IR, Nicklas JA. Use of inverse PCR to amplify and sequence breakpoints of HPRT deletion and translocation mutations. Environ Mol Mutagen. 2002;39(1):22-32
88. Wood V, Gwilliam R, Rajandream MA, Lyne M, Lyne R et al. The genome sequence of Schizosaccharomyces pombe. Nature. 2002 Feb 21;415(6874):871-80.
89. Zhang CT, Zhang R. Evaluation of gene-finding algorithms by a content-balancing accuracy index. J Biomol Struct Dyn. 2002 Jun;19(6):1045-52.
90. Zhang L, Cui X, Schmitt K, Hubert R, Navidi W, Arnheim N. Whole genome amplification from a single cell: implications for genetic analysis. Proc Natl Acad Sci U S A. 1992 Jul 1;89(13):5847-51.
91. Zhang Q. Michael. Computational Prediction of Eukaryotic Protein-Coding Genes. Nat Rev Genet. 2002 Sep;3(9):698-709.
92. Zhang Z, Schwartz S, Wagner L, Miller W. A greedy algorithm for aligning DNA sequences. J Comput Biol. 2000 Feb-Apr;7(1-2):203-14.