T A R T U Ü L I K O O L
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Tarkvarasüsteemide õppetool
Informaatika eriala

Jaanika Luik

Geeniontoloogia kasutamine oluliste seoste efektiivseks leidmiseks

Bakalaureusetöö

Juhendaja: Jaak Vilo, PhD

Autor: …… "….." mai 2004
Juhendaja: ……. "….." mai 2004
Õppetooli juhataja: …… "….." …….. 2004

TARTU 2004

Sisukord

Lühendid

cDNA	Complementary deoxyribonucleic acid	Komplementaarne DNA
cRNA	Complementary ribonucleic acid	Komplementaarne RNA
DAG	Direct Acyclic Graph	Suunatud atsükliline graaf
DBI	Database Interface	Andmebaasi kasutajaliides
DNA	Deoxyribonucleic acid	Desoksüribonukleiinhape
GO	Gene ontology	Geeniontoloogia
MGD	Mouse Genome Database	Hiire genoomi andmebaas
OBO	Open Biology Ontologies	Avatud bioloogia ontoloogiad
ORF	Open Reading Frame	Avatud lugemisraam
Perl	Practical Extraction and Report Language	Ekstraheerimis- ja raporteerimiskeel
RNA	Ribonucleic acid	Ribonukleiinhape
SGD	*Saccharomyces* Genome Database	Pärmi genoomi andmebaas
SQL	Structured Query Language	Struktuurpäringukeel

1. Sissejuhatus

Geneetika ja geenid on laiemasse huviorbiiti sattunud just viimase kümnendi jooksul, seda tänu tehniliste võimaluste hoogsale arengule. Suuremat meedia tähelepanu pälvivad kõikvõimalikud geneetilised muundamised, kloonimised. Vähem on räägitud erinevate geenide mõjust. Seda peamiselt teadmiste puudumise tõttu – kõikide geenide, õigemini nende poolt toodetavate valkude, funktsioon ei ole veel teada. Samas peab möönma, et nimetatud info polegi suuremale avalikkusele sihitud – kasutavad seda peamiselt spetsiifilisema teadustöö tegijad.

Erinevate valkude ja nende funktsioonide, protsessides osalemise ning asukoha kokku viimiseks on bioinformaatikas palju ette võetud. DNA asemel on fookuses RNA ja valgud ning väljakutseks üha kasvava andmehulga integreerimine valkude, rakkude ja lõpuks organismide täiskirjeldusega [LMG03].

Ühtset terminoloogiat kasutades püüavad erinevad tööriistad suuri andmehulki selekteerides viia kokku valke ja termineid. Sellised vahendid on suureks abiks teadlastele, kes taolisi andmeid vajavad ja kasutavad.

Geeniontoloogia annotatsioonide kaasamine andmehulkadega töötamisel võib tihti paljastada aspekte, miks kindel geenide grupp jagab sarnaseid väljendusmustreid. Bioloogilisi protsesse täidetakse ja kontrollitakse valkude poolt ning nende funktsiooni saab tuletada koos toimivate, juba osaliselt annoteeritud valkude identifitseerimisega [DB03]. Koos esile kerkivate (koos avalduvad ehk sarnaste ekspressiooniprofiilidega) geenide hulgad võivad kodeerida produkte, mis on kaasatud tavalisse bioloogilisse protsessi ja võivad asuda samas raku komponendis. Juhtudel, kus mõned tundmatud geenid ilmnevad koos hästi iseloomustatud geenide annoteerimisel identsetele või sarnastele GO protsessi terminitele, võib see viidata, et tundmatu geeniprodukt osaleb tõenäoliselt selles samas protsessis. Hüpoteesi järgi võib seega koosavalduvatel geenidel olla ühiseid jooni regulatsioonimehhanismides. Samas ei tohi unustada, et isegi tundes mõningaid valgu moodustanud osi (geene), pole see piisav valgu funktsiooni lõplikuks määramiseks [AM01].

Antud töö võib sisuliselt jagada kaheks: ülevaatlikuks teoreetiliseks osaks kasutatavatest andmetest ja seni tehtust ning praktiliseks osaks uue tööriista arendamisest ja katsetamisest.

2. Ülesande püstitus

Käesoleva töö eesmärgiks on luua tööriist, mille kasutamine võimaldaks geenide paremat tundmaõppimist ja funktsiooni tuvastamist nende hulkadega (klastritega) manipuleerimise teel. Klastrid ei moodustu suvalistest indiviididest vaid süsteemselt, näiteks võttes ühte DNA-l mingi arvu järjestikku asetsevaid geene (geenide klasterdamine on omaette suund ja selles töös lähemalt ei käsitleta). Klastreid analüüsides on võimalik identifitseerida geene, mis mitmetel erinevatel tingimustel ja manipuleerimisel järjepidevalt koos avalduvaid [VK01]. Need kogumid niiöelda paisatakse ontoloogiale, et siis kindlaks teha, kas leidub üks või mitu geenidele vastavat funktsiooni, protsessi või asupaika ehk milline GO kategooria kirjeldab nimetatud geenide hulka kõige paremini. Geenide uurimises tähtsaks osutub siinjuures statistiline tõenäosus, leitud osakaal - kas geenide ühele "ontoloogiapuu" tipule sattumise võimalus on sama suur kui mistahes teisele tipule või on sellise kokkulangemise tõenäosus suhteliselt väike ja seega statistiliselt oluline. Põhimõtteliselt uuritakse ühisosa tekitatud (komplekteeritud) hulga ja GO terminitega seotud geenide hulga vahel iga päringu korral. Mida suurema suhtelise ülekatte see moodustab, seda ilmsem, et leitud kategooria on nendele geenidele kõige iseloomulikum ja samas tõenäolisem, et päringusse kaasatud geenid võivad olla sama annotatsiooniga ning seeläbi omada sarnast funktsionaalsust.

Sellise tööriista tegemisel on vaja enim tähelepanu pöörata kiirusele - suur hulk samasuguseid operatsioone tuleb teha võimalikult väikese ajaga. See eeldab kiire algoritmi kirjutamist, andmebaasi otstarbekat kasutamist päringute tarvis. Käesolevas töös võrdlengi erinevaid algoritme ja andmestruktuure, et välja selgitada üks kiireim võimalikest, illustreerides selleni jõudmist testandmete varal.

I Teoreetiline osa

3. Geeniontoloogia

Aja jooksul on kogunenud palju infot geenide, valkude ja nende funktsionaalsuse kohta. Neid andmeid säilitatakse erinevates andmebaasides, üldjuhul liikidele vastavalt (FlyBase, Mouse Genome Database jne.). Biolooge huvitab info neist kõigist, et küsida erinevaid küsimusi ja uurida erinevaid probleeme. Neid huvitavad asjad nagu mis tõestab hiire geeni *Pax6* produkti osalust silma morfogeneesis või millised geenid või valgud aitavad kaasa epiteelkoe arengule. Kas metastaatilised rakud vähiuuringutes tekivad erinevates kasvajates sama protsessi tulemusena ja on seega potentsiaalse ravimisuunitlusega [DDBM03]? Lisaks tahetakse leida erinevate liikide geene, millel on sarnased omadused. Sellise info leidmine on võimalik ainult tänu spetsiaalsetele vahenditele ja tööriistadele ning hästi defineeritud annoteerimise süsteemile.
Geeniontoloogia (GO) projekt püüabki pakkuda spetsiifilise bioloogia valdkonna terminite kogu, millega on võimalik kirjeldada kõikide organismide geeniprodukte [TGOC01]. Ühtse terminoloogia kasutamine hõlbustab erinevate liikide geeniproduktide vaheliste suhete ja sarnaste omaduste identifitseerimist [CMBB+03].

3.1 GO konsortsium

GO konsortsium (*The Gene Ontology Consortium*) loodi organismide molekulaarsete tunnuste annoteerimiseks arendamaks adekvaatseid ontoloogiaid. See geeniontoloogia projekt on koostööl rajanev ettevõtmine, et adresseerida geeniproduktide konsistentseid kirjeldusi erinevates andmebaasides. GO kirjeldab, kuidas geeniproduktid käituvad raku kontekstis.
Projekt algas aastal 1998 eesmärgiga arendada ühine terminoloogia kirjeldamaks kolme organismi bioloogilisi funktsioone. Ühtse ontoloogia väljatöötamist toetasid esiti kolm andmebaasi: FlyBase (*Drosophila*, äädikakärbes), *Saccharomyces* Genome Database(SGD, pärm) ja Mouse Genome Database(MGD, hiir). Sellest ajast saati on GO konsortsium kasvanud, sisaldades mitmeid maailma peamisi taime-, looma- ja mikrobioloogia andmebaase.
GO kaastöötajad arendavad kolme struktuurset kontrollitud sõnastikku (ingl. k. *vocabulary*) ehk ontoloogiat, mis kirjeldavad geeniprodukte nendega seotud

liigist sõltumatute terminitega. Selle tegevuse juures on mitu aspekti: ontoloogiate eneste kirjutamine ja säilitamine, koos töötavates andmebaasides seoste loomine ontoloogiate, geenide ja geeniproduktide vahel ning tööriistade arendamine, mis hõlbustavad ontoloogiate loomist, säilitamist ja kasutamist. Praegusel hetkel koosneb GO ontoloogia peaaegu 16000 terminist ja on saamas *de facto* standardiks kõikide organismide bioloogiliste üksuste funktsionaalsete aspektide kirjeldamisel. Sellise staatuse saavutamisel on oluline roll sellistel GO omadustel nagu üldsuse kaasamine, selged eesmärgid, piiratud ulatus ja lihtne struktuur, pidev arenemine ja kohene kasutamine. Lisaks ja tänu sellele on GO-d kasutatud erinevais biomeditsiinilistes uurimustes, kaasa arvatud eksperimentaalsete andmete analüüsis ja tulemuste ennustamisel [BSGG+04].

GO mõistete kasutamine mitmetes kooskasutatavates andmebaasides hõlbustab üle nende tehtavaid päringuid. Ontoloogiad on sellise struktuuriga, et päringuid saab teha mitmel tasandil (valida, kui spetsiifilist infot vajatakse) ja seega lubab geeniproduktidele omadusi omistada mitmel tasandil, sõltuvalt sellest, kui palju on geeniproduktist teada.

Üks GO olulisi omadusi on ontoloogiate arendamise sõltumatus geeniprodukti seostest GO terminitega. Konsortsiumi liikmed konstrueerivad ja defineerivad ontoloogia terminid ja nendevahelised seosed. Seejärel kasutatakse ontoloogiaid valkude annoteerimiseks.

GO liikmete algne plaan oli koondada üheselt mõistetavad terminid, mis soodustavad päringuid üle erinevate andmebaaside. Peagi ilmnes, et annotatsioonid nende terminitega erinevate üksuste poolt pakuvad huvi ka paljudele teistele kasutajatele. Sellega seoses pakutakse ligipääsu ka päringu- ja annoteerimisprogrammidele ning geenide ja geeniproduktide annoteerimisel saadud tulemustele.

3.1.1 GO eesmärgid

GO püüab arendada liikidevahelist bioloogilist ontoloogiat, mille terminid kirjeldavad eluvormidele ühiseid molekulaarbioloogia erinevaid elemente ning mida saavad kasutada mitmed andmebaasid geenide ja nende produktide annoteerimiseks. Terminid on defineeritud, võivad omada sünonüüme ja organiseeritakse oma spetsiifilisuse järgi. Neid termineid kasutades kirjeldatakse bioloogilisi objekte.

GO pakub tööriistu nende ontoloogiatega manipuleerimiseks ja päringute teostamiseks: ontoloogiate täiendamiseks, üle interneti kasutamiseks, kuraatoritel terminite seostamiseks objektidega erinevaid meetodeid kasutades (järjendipõhised, mikrokiibid, valgu sidumise eksperimendid) [TGOC01]. Kiip (ingl. k. *array*) on massiiv ehk maatriks, matemaatikast tulenev piltlik väljend, mida rakendatakse makromolekulide analüüsil kasutatavate abivahendite korral; kahemõõtmeline suure hulga makromolekulide võrgustik [URL:EBC].

GO arendamise printsiipide hulka kuuluvad teiste hulgas „tõese tee reegel" (kõik rajad peavad vastama tegelikkusele; kirjeldatud punktis 3.2.2); terminite liigispetsiifilisuse vältimine, kuid esindatus vähemalt klassi tasandil; kõik GO atribuudid on varustatud vastavate viidetega ja kõik valkude annoteerimised GO terminitega omakorda tõestuskoodi ja viitega.

GO ei ole mõeldud bioloogiliste andmebaaside ühendamiseks. Sõnastike jagamine on samm selle poole, kuid iseenesest mitte piisav. Selleks on mitmeid põhjusi. Esiteks muutuvad teadmised kiiremini, kui toimub andmebaasides info uuendamine. Samuti väärtustavad erinevad kuraatorid andmeid erinevalt. See tähendab, et mõiste kasutusele võtmisel peab kokku leppima ka kuidas ja miks seda mõistet kasutatakse. Kolmandaks, GO ei üritagi kirjeldada bioloogia igat aspekti. GO terminid ja seosed ei proovi peegeldada geeniprodukti struktuuri või määrata, kas need on üksteisega evolutsioonilises mõttes seotud [BSGG+04].

GO ei ole geenide või geeniproduktide nomenklatuur. Ontoloogiad kirjeldavad molekulaarset fenomeni (näiteks apoptoos ehk programmeeritud raku surm), mitte bioloogilisi objekte (näiteks valgud või geenid). Erinevatel uurimisrühmadel on erinevad nimetamistavad; erinevatel organismidel on erinev arv geeniperekonna liikmeid. GO projekt keskendub bioloogiliste objektide atribuutide kirjeldamiseks mõeldud sõnastike arendamisega, mitte objektide endi nimetamisega. See on oluline saamaks aru, miks paljud geenid ja geeniproduktid on oma funktsiooni nimega.

Geneetilise järjendi projektid ja mikrokiibi eksperimendid toodavad elektroonselt genereeritud andmevooge, mis vajavad töötlemist arvutiga ligipääsetavatel süsteemidel. Süsteemidena, mis teevad valdkonna teadmised arusaadavaks nii inimestele kui arvutitele, luuakse bio-ontoloogiaid nagu GO, mis on hädavajalikud bioloogiliste vihjete eraldamiseks väga suurest andmete hulgast.

On võimatu reastada kõik GO potentsiaalsed kasutusvõimalused, kuid võib märkida, millistel juhtudel on seda juba kasutatud:
1) erinevate organismide valkudes sisalduva informatsiooni integreerimine,
2) valgu valdkondade funktsioonide määramine,
3) funktsionaalsete sarnasuste leidmine geenides, mis on üle- või alaesindatud haigustes ja meie vananemisprotsessis,
4) tõenäosuse ennustamine, et konkreetne geen on kaasatud haigustes, mida pole veel kindlate geenidega seostatud,
5) organismi arenemise käigus koos esindatud geenigruppide analüüs,
6) automaatsete viiside arendamine tuletamaks kirjandusest informatsiooni geeni funktsioonide kohta,
7) geneetika, ainevahetuse ja produkti vastastikuse toime võrgustike mudelite kinnitamine.

Segaproduktid

Mitmete ontoloogiate olemasolu võimaldab luua „segaprodukte", mis maksimeerivad iga ontoloogia kasutusvõimalusi samal ajal liiasusi vältides. Näiteks kombineerides arenguga seotud mõisteid GO protsessi-ontoloogias ontoloogiaga, mis kirjeldab *Drosophila* anatoomilisi struktuure, saab luua kärbse arengu ontoloogia. Seda saab korrata seega ka teiste organismide jaoks, ilma et peaks GO-d kurnama liigispetsiifiliste mõistetega. Samamoodi saaks luua biosünteetiliste radade ontoloogia, kombineerides biosünteesi mõisted GO protsessi-ontoloogias keemilise ontoloogiaga.

3.1.2 GO tagamaad

GO laseb annoteerida geene ja valke limiteeritud hulga atribuutidega. Näiteks ei luba GO geene kirjeldada mõistetega, millistes rakkudes või kudedes, millistes arenguetappides nad on esindatud või mis kirjeldavad nende osalust haigustes. See ei ole GO ülesanne, selleks otstarbeks on arendatud teised ontoloogiad. Samas toetab neid ka GO konsortsium tehes kättesaadavaks tööriistad ontoloogiate modifitseerimiseks.
GO ei ole ainus katse genoomi annoteerimiseks struktureeritud sõnastike abil. Samuti pole see ainuke selletaoline kasutuses olev kataloogiseeria. Taoliste kataloogide ja GO vahel on proovitud teha tõlketabeleid, kuid sellised ühendused pole terviklikud ega täpsed. Nimekirja vabalt kättesaadavatest ontoloogiatest, mis on GO-le sarnase struktuuriga, võib leida OBO (*Open Biology Ontologies*)

veebilehel[1]. Suurem nimistu, mis sisaldab OBO ontoloogiaid ja muid kontrollitud sõnastikke, aga ei täida OBO kriteeriume, on saadaval Ontology Working Group veebilehel[2] (haldajaks Microarray Gene Expression Data Society).

Kõik uued terminid on internetis nähtaval ning soovitused ja kommentaarid nende kohta arendajate poolt oodatud. GO terminoloogia on dünaamiline, seda saab muuta ja täiendada vastavalt kasvavatele teadmistele geenivaldkonnas [BAHI01]. GO projekt on pidevalt arenev ja igasugune tagasiside teretulnud kõigilt kasutajatelt.

3.2 GO andmebaas

3.2.1 Ontoloogiad

Ontoloogia on omavahel seotud terminite sõnastik. Antud juhul on kasutatavad terminid kõik bioloogiast. Ontoloogiad pakuvad sõnastikku teemakohaste teadmiste ning terminite (nimetatakse ka kategooriateks) vahel olevate seoste hulga väljendamiseks ja kommunikatsiooniks. Need võivad struktuurselt olla väga keerulised, aga ka suhteliselt lihtsad. Mis peamine - ontoloogiad koguvad valdkonna teadmised viisil, mida on lihtne arvutiga töödelda. Kuna ontoloogia terminid ja nendevahelised seosed on hoolikalt defineeritud, hõlbustab ontoloogiate kasutamine standardannotatsioonide tegemist, parandab arvutuslikke päringuid ja võimaldab toetada saadaval informatsioonist tuletatu konstruktsiooni.
Bioinformaatika vaatevinklist arendatakse ontoloogiaid funktsioonide ja funktsionaalsete seoste kirjelduste väljendamiseks, mis muutub üha tähtsamaks genoomi järjendite genereerimisel ja automaatsel annoteerimisel, seoste täpsemal defineerimisel, lokaalsete ja integreeritud funktsioonide määramisel; seoste defineerimisel ensüümide, nende asukoha metaboolsel teel, reageerimisproduktide ja substraatide ning superperekondade, kuhu nad kuuluvad, vahel [AM01].

GO-s on kolm organiseerimisprintsiipi: molekulaarne funktsioon, bioloogiline protsess ja rakuline paiknemine. Need kolm valiti klassifitseerimiseks seepärast, et need esindavad informatsiooni, mis on ühised kõigile eluvormidele ja on baasiks geenide ja geeniproduktide annoteerimisel. Samas võimaldab see ka

[1] http://obo.sourceforge.net/
[2] http://mged.sourceforge.net/ontologies/OntologyResources.php

valgu funktsiooni defineerimist mitmel tasandil, sisaldades selle biokeemilisi tegevusi, bioloogilisi rolle ja rakulist struktuuri [LMG03]. Praegune tingimus termini kaasamiseks on selle kehtivus rohkem kui ühe taksonoomilise klassi või organismi jaoks. Geeniproduktil (valgul) on vähemalt üks funktsioon ja seda kasutatakse vähemalt ühes protsessis. Valk võib olla seotud ühe või mitme rakukomponendiga. GO ontoloogiad on arendatud nii, et sisaldavad kõiki termineid, mis neisse valdkondadesse kuuluvad, arvestamata, kas bioloogiline atribuut on piiratud kindlate taksonoomiliste gruppidega. Seepärast on kaasatud ka bioloogilised protsessid, mis toimuvad ainult taimedes või imetajates.

3.2.1.1 Terminid

Terminite defineerimiseks kasutab GO võimalikult palju Oxfordi ülikooli molekulaarbioloogia sõnastikku (*The Oxford Dictionary of Molecular Biology*, 1997), teised definitsioonid on pärit biokeemia ja molekulaarbioloogia töödest või SWISS-PROT sarnastest allikatest.
Andmete uuenedes saab termineid klassifitseerida vananenuiks (ingl. k. *obsolete*), kuid päris ära ei kustutata neid tegelikult kunagi. See on vajalik ajaks, mil termin on kuulutatud vananenuks, kuid pole veel asendatud uuega. Vastasel juhul annaksid vastavad annotatsioonid sel ajal null-väärtusi. Et eristada leksiliselt identseid termineid, mida eraldiseisvad organismide kogukonnad on kasutanud erinevate annotatsioonidega, märgistatakse need „tähenduses" (ingl. k. *in the sense of, sensu*) väljaga, et termini mõistmiseks viidata vastavale kontekstile [BSGG+04].

3.2.1.2 Seosed

Seosed terminite vahel on „on" (ingl. k. *is a*) ja „osa" (ingl. k. *part of*) tüüpi. Neist esimene tähendab „laps-termini" (ingl. k. *child*) alamüksuseks olemist oma eellasele (ingl. k. *parent*) - laps on igal juhul tõene oma vanema suhtes; teine, „osa" tüüpi suhe, tähendab, et „laps" on eellase komponent, tema osa.
Kuigi sama ontoloogia piirides on eri tüüpi seoste haldamine raske, on see vajalik semantika täpseks kajastamiseks [TGOC01].

3.2.1.3 Ontoloogia: funktsioon

Molekulaarne funktsioon kirjeldab tegevusi (nagu katalüüs ja sidumine) molekulaarsel tasemel. GO funktsiooni mõisted esindavad pigem tegevusi kui hulki (molekule või komplekse), mis neid teostavad. Need mõisted ei täpsusta kus, millal või mis kontekstis tegevus aset leiab. Üldiselt vastavad funktsioonid tegevustele, mida individuaalsete valkude poolt läbi viiakse, samas teostatakse neist mõned valkude komplekside poolt.
Nagu eespool mainitud, on lihtne segamini ajada valku selle molekulaarse funktsiooniga. Just sel põhjusel on paljude GO funktsioonide juurde selguse mõttes lisatud sõna *activity*.

3.2.1.4 Ontoloogia: protsess

Bioloogiline protsess on ühe või mitme molekulaarse funktsiooni järjestus. Näide laiahaardelistest protsesside mõistetest on raku kasvamine ja säilitamine või signaali transduktsioon. Rohkem spetsiifilised mõisted on pürimidiini metabolism ja alfa-glükoosi transport. Võib olla raske eristada protsessi ja funktsiooni, aga üldine reegel on, et protsessis on rohkem eristatavaid samme.

3.2.1.5 Ontoloogia: rakuline paiknemine

Raku koostisosa on osa rakust, millel on kindel koht raku ehituses; või on see osa mõnest suuremast objektist, mis võib olla anatoomiline struktuur (näiteks endoplasmaatiline retiikulum või tuum) või valkude grupp (nagu ribosoom, proteiini dimeer).

3.2.2 Ontoloogiate haldamine

GO kolm ontoloogiat on üksteisest sõltumatud, nende vahel ei ole seoseid. Seega on konkreetse bioloogilise olemi atribuut terminiga seotud nii, et puudub olemi lisainfole (mis võib olla eksitav või vale) vihjamise risk [BSGG+04].
GO mõisteid hallatakse suunatud atsüklilise graafi struktuuris, mis erineb hierarhiast selle poolest, et „lapsel" (spetsiifilisem mõiste) võib olla mitu eellast (vähem spetsiifiline mõiste). Näiteks on protsessi mõistel heksoosi biosüntees kaks eellast – heksoosi metabolism ja monosahhariidi biosüntees. Seda

sellepärast, et biosüntees on metabolismi alamtüüp ja heksoos monosahhariidi tüüp. Kui geen on kaasatud heksoosi biosünteesi, on see automaatselt ka osa heksoosi metabolismist ja monosahhariidide biosünteesist, sest iga GO mõiste peab alluma nn. tõese tee reeglile: kui laps-mõiste kirjeldab valku, siis kõik selle vanem-mõisted peavad samuti sellele valgule viitama.

Tõese tee reegel

Rada laps-terminist selle kõige kaugema eellaseni (eellasteni) on ja peab olema alati tõene. Kui uue geeniprodukti avastamise või liigi spetsiifilisuse korral seda reeglit järgida ei õnnestu, tuleb ontoloogia üle vaadata, lisada uusi tippe ja termineid ning tõmmata uued seosed nende vahel. Selline laiendamine võib aga probleeme tekitada GO struktuuri konsistentsuse säilitamisel, kui lisanduvad peened molekulaarsete detailide tasandid ning seega tõstatada küsimuse, millal on õige aeg laiendamist piirama hakata, kuna see teeb ontoloogiad liialt liigispetsiifiliseks.

3.2.2.1 GO Slims

Genoomi või cDNA kogumi GO annotatsioonide tulemuste raporteerimisel on kasulik omada kolme GO ontoloogia kõrgtaseme vaadet. Neid tuntakse „GO Slims" nime all [URL:GOslim]. On selge, et GO *slim*ide jagamine kujutaks endast suurt eelist - see teeks GO terminite jaotumise kokkuvõtete võrdlemise väga lihtsaks. On ka ilmne, et erinevad grupid vajavad vastavalt vajadustele erinevaid GO *slim*e. Sel põhjusel pakub GO kataloogi erinevate *slim*ide hoidmiseks (*go/GO_slims/*). On ettenähtud, et seal saab olema kaks üldist klassi GO Slim faile. Üldisi konsortsiumi poolt tehtud GO *slim*e on vähemalt üks. Need on dünaamilised failid, mida uuendatakse, et peegeldada muutusi GO failides endis.
Ühenduse liikmete lisatud GO *slim*id on kasutatud kindlas publikatsioonis või analüüsis. Neid hoitakse kahel põhjusel. Esiteks lihtsa ligipääsu andmiseks mainitus kasutatud GO terminitele; teiseks taaskasutamiseks teiste liikmete poolt.
GO Slim failid peaksid olema sama struktuuriga nagu GO *flat*-failid. Üksik *slim*-fail peaks sisaldama GO kontseptsioone ühest, kahest või kõigist kolmest GO ontoloogiast.

3.2.3 Faili formaat

3.2.3.1 GO flat-fail

On kolm eraldi tekstifaili iga ontoloogia kohta. 23-nda veebruari 2004 seisuga on funktsioone, protsesse ja asukohta kirjeldavates failides vastavalt 7288, 8337 ja 1390 terminit. Lisaks fail terminite defineerimiseks, kus on kirjas termin, selle GO ID, definitsioon ja viide definitsioonile [URL:GOformat].

Vanem-laps suhe on failis kujul

vanem_termin (*parent_term*)
 laps_termin (*child_term*)

IS A suhe kujul

%termin0
 %termin1 % termin2

kus termin1 on nii termin0 kui ka termin2 alamklass;

PART OF suhe kujul

%termin0
 %termin1 < termin2 < termin3

kus termin1 on termin0 alamklass ja samas ka termin2 ja termin3 osa.

Faili süntaks on kujul

< | % term [; db cross ref]* [; synonym:text]* [< | % term]*

Näiteks rida molekulaarse funktsiooni ontoloogiast (tegelikus failis on see ühe reana)

%peroxidase activity ; GO:0004601, GO:0016685, GO:0016686, GO:0016687 ; EC:1.11.1.7 ; MetaCyc:PEROXID-RXN ; synonym:eosinophil peroxidase activity ;

synonym:lactoperoxidase activity ; synonym:myeloperoxidase activity % oxidoreductase activity\, acting on peroxide as acceptor ; GO:0016684

3.2.3.2 OBO flat-fail

See formaat on peamiselt GO definitsioonfaili laiendus mõningate muudatustega. Üks oluline erinevus on, et tundmatud märgendid, ükskõik, mis kontekstis ei genereeri tingimata saatuslikke vigu. See lubab parseril lugeda ja töötada infot sisaldavate failidega, mida konkreetne tööriist ei kasutagi.
Dokument on struktureeritud järgmiselt:

<päis>
<stants>
<stants>
…

Stants on dokumendi märgistatud osa, mis näitab, et kirjeldatakse kindlat tüüpi objekti. Stantsi struktuur :

[<Objekti tüüp>]
<märgis>: <väärtus>
<märgis>: <väärtus>

Kõik märgis-väärtus paarid asuvad eraldi real või on rida katkestatud \<*newline*> kombinatsiooniga. Nii märgis kui väärtus võivad sisaldada veel järgmisi sümbolikombinatsioone: \n (reavahetus), \: (koolon), \W (tühik), \t (tabulatsioon), \, (koma), \" (jutumärgid), \\ (kaldkriips). Tähtis on meeles pidada, et neid saab kasutada vaid siis ,kui neil pole parserile tähendust. Varjestamata tühikuid eiratakse.
Näide märgis-väärtus paaridest:

[Term]
id: GO:0048266
def: "A change in the behavior of an organism as a result of nociceptor activation\, or nociception. Nociceptors\, which are peripheral receptors for pain\, include receptors which are sensitive to painful mechanical stimuli\, extreme heat or cold\, and chemical stimuli. Nociception is the perception of pain." [GO:jic, http://cancerweb.ncl.ac.uk]

Dokumendi päis koosneb märgis-väärtus paaride seeriast. Kohustuslik on esimese märgisena päises formaadi versioon, teiste märgiste järjekord pole oluline.

Kohustuslikud märgised on
1) *format-version* – flat-faili kodeerimise versioon; see lubab parseritel kasutada erinevaid *flat*-fail formaate, isegi kui selle põhistruktuur muutub,
2) *typeref* – tüübi kirjeldamise dokumendile viitav URL. Tüübikirjeldus on dokument selles *flat*-fail formaadis, mis sisaldab seose tüübi definitsiooni. Iga dokument, mis sisaldab mittesisseehitatud seosetüüpe nõudvaid märgiseid, peab sisaldama *typeref* rida.

Mittekohustuslikud märgised on
1) *version* – konkreetse faili versioon,
2) *date* – kuupäev dd:MM:yyyy HH:mm formaadis,
3) *saved-by* – viimati salvestanud kasutaja nimi,
4) *auto-generated-by* – faili genereerinud programm,
5) *remark* – üldised kommentaarid faili kohta,
6) *subsetdef* – termini alamhulga kirjeldus. Selle märgis peaks sisaldama alamhulga nime, tühikut ja seejärel jutumärkides alamhulga kirjeldust.

Hetkel toetab *flat*-faili formaat kahte stantsitüüpi: [Term] ja [Typedef].

Märgendid [Term] stantsis
Termini kirjelduse osa koosneb märgend-väärtus paaridest. Iga termini kirjeldus algab *id* märgisega. Sealne väärtus määrab termini, millele kõik järgnevad märgised viitavad.
Termini kirjeldus ei pea olema täielik. Termin võib sisaldada mitmeid kirjeldusi ühes failis (või mitmeid kirjeldusi mitmetes failides). Iga kirjeldus võib anda lisainformatsiooni. See teeb parseritele spetsiaalse või valikulise informatsiooni lugemise väga lihtsaks. Parsimisel võivad esineda vead, kui kirjeldus räägib vastu eelmisele kirjeldusele (näiteks üks termini kirjeldus annab teise termini nime, kui teine kirjeldus) või parser lõpetab analüüsi, aga terminil puudub ikka mõni kohustuslik väli (näiteks nimi).

Kohustuslikud märgised on

1) *id* – termini unikaalne ID, mis võib olla mistahes string; see märgis peab olema alati esimene igas termini kirjelduses,
2) *name* – termini nimi. Igal termini on ainult üks defineeritud nimi.

Mittekohustuslikud märgised on
1) *alt_id* – defineerib alternatiivse termini ID, neid võib olla mitu,
2) *comment* – kommentaar terminile,
3) *def* – konkreetse termini definitsioon. Märgise väärtus peaks olema jutumärkides tekst, mille järgneb *dbxref (database crossreference,* andmebaasi ristviide*)* nimistu, mis sisaldab viiteid selle definitsiooni originaali kirjeldustele. Näiteks:

definition: "The breakdown into simpler components of (+)-camphor, a bicyclic monoterpene ketone." [UM-BBD:pathway "", http://umbbd.ahc.umn.edu/cam/cam_map.html ""] ,

4) *xref_analog – dbxref* (andmebaasi ristviide, *database crossreference*), mis kirjeldab analoogset terminit teises sõnastikus (võib olla mitu analoogi),
5) *xref_unk* – tundmatut tüüpi *dbxref*, mida võib olla mistahes arv; seda märgist ei soovitata kastutada,
6) *subset* – näitab termini alamhulka, kuhu see kuulub; selle märgise väärtus on faili päises *subsetdef (subset definition,* alamhulga definitsioon*)* märgises defineeritud alamhulga nimi; termin tohib kuuluda mitmesse alamhulka,
7) *synonym* – märgend termini mittetäpse sünonüümiga ja mõne originaali kirjeldava *xref*-ga; märgendi sisu peaks olema jutumärkides, millele järgneb mittekohustuslik *dbxref* nimistu, mis kirjeldab sünonüümi originaali,
8) *exact_synonym* – analoogiline eelnenule, sisaldab täpset sünonüümi,
9) *narrow_synonym* – analoogiline punktile 7), sisaldab spetsiifilist sünonüümi,
10) *broad_synonym* – analoogiline punktile 7), sisaldab vähem spetsiifilist sünonüümi,
11) *is_a* – kirjeldab alamklassilist suhet terminite vahel; terminid ilma *is_a* seoseta on „juured" (ingl. k. *root*),
12) *is_obsolete* – kas termin on vananenud või ei,
13) *use_term* – pakub, millist terminit kasutada vananenud termini asemel; väärtuseks on teise termini ID; see pole kohustuslik vananenud termini esinemise korral, kuid on soovitatav,
14) *relationship* – kirjeldab seosetüüpi kahe termini vahel, väärtuseks on seosetüübi ID ja sihttermini ID; seosetüübi nimi peab olema sama, mis defineeritakse *typedef* märgise stantsis,

15) *replaced_by* – selgitab, milline termin tõrjus välja vananenud termini. Väärtuseks on selle teise termini ID. See märgis pole kohustuslik vananenud termini olemasolu korral, aga on soovitatav.

Dbxref formaat
Dbxref definitsioonid on kas kujul
<*dbxref* nimi>
või
<*dbxref* nimi> „<*dbxref* kirjeldus>"

Kokkuleppe kohaselt on *dbxref* nimi kooloniga eraldatud võti-väärtus paar, aga see pole nõue. *Dbxref* kirjeldus võib olla nullist või enamast tähemärgist *dbxref*-i kirjeldav string.

Dbxref nimistuid kasutatakse, kui märgise väärtus sisaldab mitmeid *dbxref*-e. Need nimistud on kujul [<*dbxref* definitsioon>, <*dbxref* definitsioon>,…].

Märgendid [Typedef] stantsis
[Typedef] stantsid toetavad kõik samu märgendeid, mis [Term] stants. Need lihtsalt kirjeldavad objektide erinevaid klasse.

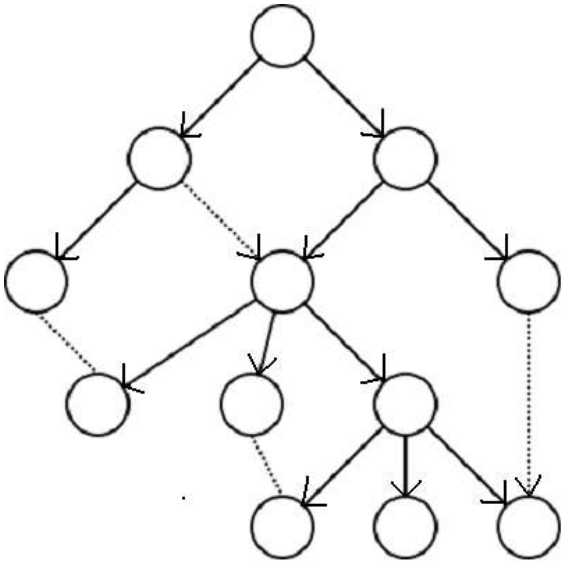
Erinevalt GO *flat*-faili formaadist ei kirjelda see failiformaat juurega suunatud atsüklilist graafi. See formaat kirjeldab juureta, tõenäoliselt tsüklilist, suunatud graafe. Seda analüüsivad parserid peavad olema võimelised arvestama tsüklilise struktuuri ja mitme juure (või siis hoopis ilma juureta) olemasolu võimalusega.

3.2.3.3 XML-fail

XML-faili uuendatakse iga kuu, see koostatakse *flat*-failidest ja geeni assotsiatsioonifailidest. Saadaval on kaks faili - üks koos assotsiatsioonidega, teine ilma.
GO XML andmebaas koosneb põhiliselt paaridest GO:*termid*. Igal terminil võib olla üks go:*name*, go:*accession*, go:*definition* ja mitu go:*dbxref*-e või go:*association*-e. Neist kolm esimest selgitavad ennast ise, go:*dbxref* tähistab terminit lisaandmebaasis ja go:*association* iga termini assotsiatsioone. Go:*association*-iga võib olla koos nii go:*evidence* kui ka go:*gene_product*, millest esimene hoiab assotsiatsiooni tõestava tõestuskoodi ga:*dbxref*-i ja teine geeni sümbolit ja go:*dbxref*-i.

3.2.3.4 MySQL

GO andmebaas on disainitud GO ja OBO stiilis ontoloogiate hoidmiseks koos annotatsioonide ja muude andmetega. OBO stiilis ontoloogiate ja GO andmebaasi keskne kontseptsioon on graaf. GO ja OBO terminid on tipud selles ja seosed nende vahel on jooned (nooled).



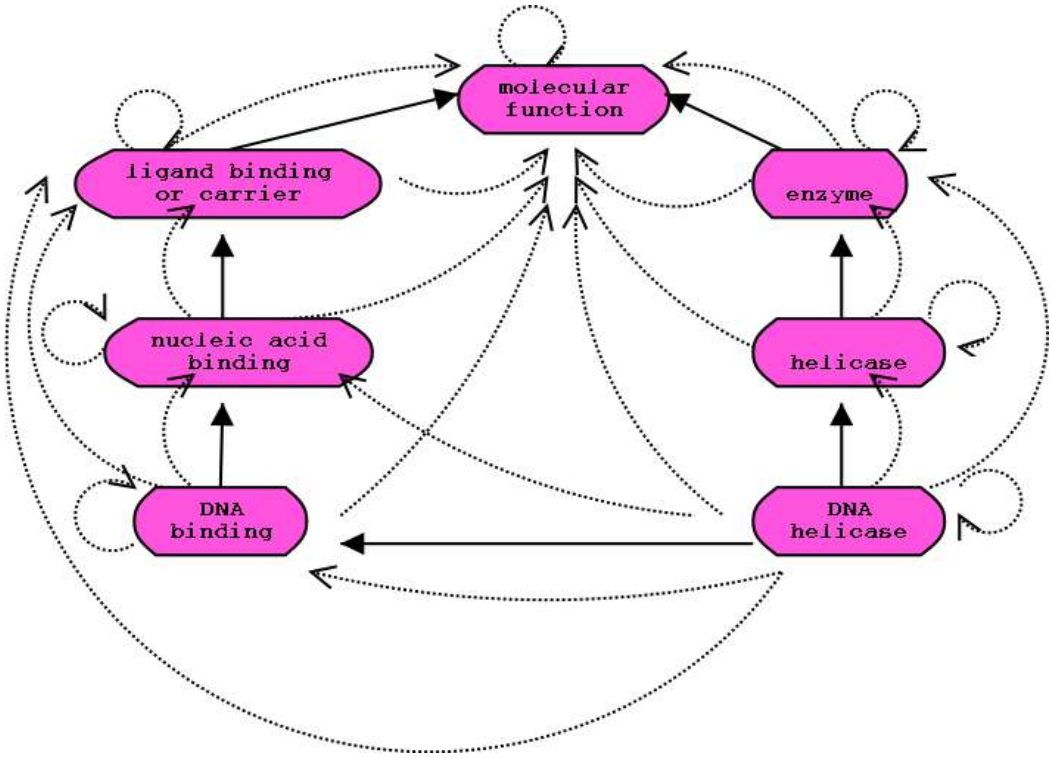
Joonis 1. Suunatud atsükliline graaf ontoloogiate hoidmiseks [URL:MySQL].

Seda kirjeldavad vastavalt tabelid *term* ja *term2term*. Alternatiiv on vaadata GO ja OBO stiilis ontoloogiaid koosnemas binaarsetest seostest, mis on seosetüüp kahe olemi vahel.

Hetkel kasutatavad seosetüübid on *is_a*, *part_of* ja *develops_from* anatoomilistes ontoloogiatest. Need tähendavad vastavalt on-, osa-, ja arenenud- seoseid. See seis võib tulevikus muutuda, kui soovida suuremat hulka seoste tüüpe, kus mõned on teiste spetsifikatsioonid. Selleks säilitatakse neid terminitega samast tabelis. Tüüpe eraldatakse teadmisega, et need tuletatakse seose tüübi ontoloogiast.

Tehes ontoloogiatele orienteeritud päringuid on tihti vajalik mingit viisi graafi läbida. On võimalik kasutada tabelit *term2term* graafi korduvaks läbimiseks, aga see nõuab mitmeid SQL päringuid. Enamik SQL tööriistu aga ei toeta sellist rekursiivset pärimist, mis on mõningate selliste päringute puhul vajalik.

Siiski on sellist sorti päringud GO andmebaasis võimalikud, sest tee igast tipust tema kõikide eellasteni (mis on võrdväärne teega igast tipust lasteni) eelarvutatakse. Seda nimetatakse suhte transitiivseks sulundiks ning asub tabelis *graph_path*, mis sisaldab ka termidevahelisi kaugusi. Nimetatud tabel näitab, kas eksisteerib tee vanema ja lapse vahel ning kaugust nende vahel. Iga võimaliku tee kohta on eraldi kirje.
Arvutatakse refleksiivne transitiivne sulund, mis tähendab, et iga termin on seotud lisaks teistele tippudele iseendaga (vahemaa on sel juhul null) ja tabel sisaldab ka vastavat kirjet. Praktikas teeb see lihtsamaks eelmainitud päringute teostamise.



Joonis 2. Refleksiivse transitiivse sulundi näide DNA helikaasist ja selle vanematest. Jämedad jooned tähistavad otsest *is a* suhet (säilitatakse tabelis *term2term*); katkendjooned tähistavad kaudset pärilikkussuhet (sulundit), mis säilitatakse tabelis *graph_path* [URL:MySQL].

GO andmebaas kasutab tabelite ühendamiseks surrogaatvõtmeid. Neil pole mingit tähendust ja erinevad ligipääsunumbritest. Näiteks on tabelis *term* veerud *id* ja *acc*. *Id* välja kasutatakse tabelite ühendamiseks, *acc* väli on OBO ligipääsunumbrite säilitamiseks.

Moodulid

GO skeem on modulaarse disainiga. Moodustatud on järgmised moodulid:

1)*go-general* - üldised tabelid, mitte GO ega OBO spetsiifilised,
2)*go-graph* - tabelid, mis väljendavad GO/OBO stiilis ontoloogia keskset kontseptsiooni; tipud on terminid ning nooled nendevahelised suhted, mida vastavalt kirjeldavad tabelid *term* ja *term2term*,
3)*go-meta* - graafi tippude metaandmed, näiteks sünonüümid, lingid edasistele andmegaasidele, kommentaarid ja definitsioonid,
4)*go-associations* - geeniproduktide annotatsioonid GO/OBO termineid kasutades; säilitab geeni produkti enda metaandmed nagu ka info konkreetse assotsiatsiooni kohta GO termini ja geeniprodukti vahel (näiteks tõestuskood),
5)*go-seq* - geeniproduktiga seotud bioloogiline järjend,
6)*go-optimisations* - ühtlustus, mis laseb teatud päringuid kiiremini teha. Täpsemalt arvutatakse kõigi GO seoste täielik sulund; see säilitatakse *graph_path* tabelis, kus on rida igale tipp-eellane suhtele GO ontoloogias.

3.2.4 Annotatsioonid

Ontoloogiate loomine ja assotsiatsioonid ontoloogia terminite ja geeniproduktide vahel on kaks iseseisvat operatsiooni. Koostööd tegevad andmebaasid annoteerivad oma geene ja geeniprodukte GO terminitega kahte printsiipi järgides. Esiteks, annotatsioonid peavad allikale viitama. Selleks võib olla kirjanduslik viide, mõni teine andmebaas või arvutuslik analüüs. Teiseks peab iga annotatsioon viitama tõestusmaterjalile eelnimetatud allikas, millele assotsiatsioon GO termini ja geeniprodukti vahel põhineb. GO kasutab tõestusmaterjali tüübile viitamiseks lihtsat kindlat sõnastikku, mis on toodud tabelis 1. Iga annotatsiooni varustamine nii eksperimendi meetodi kui viitega on mõeldud abiks teadlastele annotatsiooni usaldatavuse hindamiseks ja on oluline edasiseks arendustööks ja nende annotatsioonide kasutamiseks [TGOC01].

IMP	*inferred from mutant phenotype*	järeldatud mutantsest fenotüübist
IGI	*inferred from genetic interaction*	järeldatud geneetilisest koosmõjust
IPI	*inferred from physical interaction*	järeldatud füüsilisest koosmõjust
ISS	*from sequence or structural similarity inferred*	järeldatud järjestikkusest sarnasusest
IDA	*inferred from direct assay*	järeldatud otsesest katsest
IEP	*inferred from expression pattern*	järeldatud väljendumise mustrist

IEA	*inferred from electronic annotation*	järeldatud elektroonilisest annotatsioonist
TAS	*traceable author statement*	jälgitav autori seisukoht
NAS	*non-traceable author statement*	mittejälgitav autori seisukoht
ND	*no biological data available*	bioloogilised andmed puuduvad
IC	*inferred by curator*	kuraatori järeldus

Tabel 1. Geeni annoteerimisel kasutatavad tõestuskoodid.

Geeniproduktid

Lihtne on segamini ajada valku ja selle funktsiooni, sest tihti on need kirjeldatud täpselt samade sõnadega. Ometi on olemas formaalne erinevus: ühel valgul võib olla mitu funktsiooni, samas kui mitmel valgul on sama funktsioon.
Paljud valgud seonduvad komplekselt funktsioneerivateks hulkadeks või valgugruppideks, mis sisaldavad tihti väikeseid molekule. Nende ulatus komplekshulkades on suhteliselt lihtsatest erinevate valkude ühendist keeruliste valguühenditeni (näiteks ribosoom).
Praegusel hetkel ei ole väikesed molekulid GO-s esindatud. Tulevikus saaks luua segaprodukte linkides GO olemasolevate väikeste molekulide andmebaasidega (näiteks Klotho ja LIGAND).

Geeniontoloogia terminid lisatakse bioloogia andmebaaside geeniproduktidele nende annotatsioonis. Praegu on erinevate andmebaaside toel GO-d kasutatud rohkem kui miljoni valgu annoteerimiseks [BSGG+04]. GO annotatsioonid ongi assotsiatsioonid geeniproduktide ja neid kirjeldavate GO terminite vahel. Neist esimene on RNA või proteiinsaadus (valk). Kuna üks geen võib osaleda mitme erineva atribuudiga produkti genereerimisel, soovitab GO oma termineid siduda geeniprodukte tähistavate andmebaasi objektidega, mitte geenide endiga. Kui individuaalsete geeniproduktide eristamiseks identifikaatoreid pole, võib GO termineid siduda geeni identifitseerijaga; geen on seotud kõigi GO mõistetega, mida saab kasutada iga selle geeni produkti puhul.

Geeni produkt võib olla annoteeritud null või enama tipuga igast ontoloogiast; produkti annoteerimine ühe ontoloogiaga on sõltumatu selle annoteerimisest teiste ontoloogiatega. Annotatsioonid peaks peegeldama geeniprodukti normaalset funktsiooni, protsessi ja asukohta; muteerumise või haiguse seisukohast vaadeldud aktiivsus või asukoht ei kuulu tavaliselt selle alla. Samas sõltub nn. normaalsituatsioon annoteerija vaatekohast. Näiteks kasutavad paljud

viirused proteiine peremeestena, et viia lõpuni viirusprotsess. Sel juhul teeb peremeesproteiin oma seisukohalt midagi ebanormaalset, kuid viiruse jaoks on see täiesti normaalne. GO annoteerijad lahendavad need juhud geeni assotsiatsioonifaili *Taxon* veergu kaks takson ID-d lisades. Neist esimene kuulub organismile, mis kodeerib geeni produkti ja teine sellele, mis kasutab geeniprodukti.
Individuaalset geeniprodukti, mis on osa kompleksist, võib annoteerida terminitega, mis kirjeldavad kompleksi tegevust (funktsiooni või protsessi). See on kõnekeeles tuntud kui kompleksi potentsiaali annotatsioon ning on viis informatsiooni, mida kompleks andmebaasi objektide ja komplekse tähistatavate identifitseerijate eemalolekul teeb, kogumiseks.
Individuaalsete geeniproduktide annoteerimiseks vastavalt kompleksi atribuutidele on eriti kasulik molekulaarse funktsiooni annotatsioonideks juhtudel, kus kompleks on aktiivne, aga individuaalsed alamüksused mitte. Need funktsiooni annotatsioonid peavad sisaldama „soodustab" (ingl. k. *contributes to*) kirjet.

Iga liigi andmebaasis on soovitatav geeniprodukti annoteerida ontoloogia kõige detailsemal tasandil, mis korrektselt valgu bioloogiat kirjeldab. Meeles tuleb pidada nn. tõese tee reeglit – terminile annoteerimine kaasab annoteerimise ka kõigile terminite eellastele, nii et kasulik oleks ka neid kontrollida.

GO konsortsiumi andmebaasid kasutavad geenide või nende produktide GO terminitega annoteerimiseks manuaalseid või automaatseid meetodeid. Mõlemad annotatsioonid tehakse vastavalt eelpool mainitud kahele printsiibile.

On oluline vahe, kas geen või selle produkt on annoteeritud „teadmata" (ingl. k. *unknown*) funktsiooni, protsessi ja/või komponendiga või ei ole üldse annoteeritud. „Teadmata" tähendab, et keegi on annoteerida proovinud, kuid ei leidnud informatsiooni. Annotatsiooni puudumine tähendab, et polegi proovitud seda teha. Iga „teadmata" annotatsiooni jaoks mistahes kolmest ontoloogiast peaks kuraatorid viitama tunnistusele andmebaasis, mis seletab olulise bioloogilise informatsiooni kirjandusest või muus nende poolt kasutatud allikast mitteleidmist. Tõestuskood sellele on ND tähistamaks andmete puudumist (ingl. k. *no data*). Erandiks on juhud, kui paber otseselt ütleb, et midagi on teadmata; sel juhul saab paberit vaadelda viitena koos TAS või NAS tõestuskoodiga. Veel üks erand tekib juhul, kui geeni produktil on järjestikune sarnasus domeeniga, mis on dokumenteeritud kui „teadmata". Sel juhul on tõestuskood ISS.

Kui ei olda kindel, kus geeniprodukt toimib, tuleks see ära märkida selle kahele tipule annoteerimisega, millest üks võib olla teise eellane. Need kaks annotatsiooni võivad olla kas sama või erineva tõestuskoodiga. Samasugustel juhtudel üldiste või spetsiifiliste funktsioonide ja protsessidega tuleb toimida samamoodi. Kui kirjanduses on konfliktseid väited, võib annoteerida ka mitme üksteisega vastuolus oleva tipuga.

GO terminid on seotud ka hulga inimese valkudega, mida kirjeldavad Swiss-Prot/TrEMBL/InterPro ja Ensembl. Need annotatsioonid on saadaval GOA Human failis EBI ja GO FTP veebilehtedel. GOA projekti andmed on kättesaadavad ka EMBL/DDBJ/GenBank nukleotiidjärjestuses säilitatuna EMBL-Bank baasis.

GO annotatsioonide õigsus on kõigi konsortsiumi liikmete jaoks prioriteetne. Iga liikmeorganisatsioon on vastutav oma annotatsioonide täpsuse, ajakohasuse ja vigade parandamise eest. GO konsortsium otsib ka võimalikke viise kvaliteedi edasise kindlustamise parandamiseks nagu tööriistade arendamine potentsiaalsete vigaste annotatsioonide automaatseks avastamiseks.

Nagu ontoloogiad, nii ka geeniproduktide/GO assotsiatsioonide hulgad kaasa aitavatelt gruppidelt on saadaval GO veebilehel[3]. Geeniproduktide ja GO terminite vaheliste assotsiatsioonide failid, liikmeorganisatsioonide poolt loodud, on olemas nende omadel FTP külgedel. Geeni assotsiatsioonifailid sisaldavad objektide ID-sid faili loomisele kaasa aidanud andmebaasis, nagu ka viiteid ja tõestuskoode. On ka faile, mis sisaldavad GO terminitega annoteeritud geeniprodukti SWISS-PROT/TrEMBL proteiinijärjestuse identifikaatoreid.

3.2.4.1 Faili formaat

Koostööd tegevad andmebaasid ekspordivad GO-sse tabulatsiooneraldusega faili, mitteametliku nimega geeni assotsiatsioonifail [URL:GOannot], kus on andmebaasi objektide ja GO terminite vahelised lingid. Vaatamata erikeelele võib andmebaasi objekt tähistada geeni või geeni produkti. Faili veerud on kirjeldatud allpool, õiges järjestuses ja näidetega.

Sisu *DB_Object_ID* väljas on andmebaasi objekti identifitseerija, mis võib, aga ei pea vastama täpselt paberil kirjeldatule. Näiteks võib paberil kirjeldatud proteiin

[3]http://www.geneontology.org

toetada geeni, mis kodeerib proteiini, annotatsioone (geeni ID *DB_Object_ID* väljas) või proteiini annotatsioone (proteiini ID *DB_Object_ID* väljas).
Välja *DB_Object_Symbol* sisu peaks olema sümbol, mis võimalusel tähendaks midagi bioloogile (näiteks geeni sümbol). See pole ID või ligipääsunumber - unikaalse identifitseerija annab *DB_Object_ID* - kuid ID-sid saab kasutada *DB_Object_Symbol* väljas, kui ühtki bioloogilise tähendusega tähist ei ole (näiteks juhul, kui nimetu geen annoteeritakse).

Objekti tüüp (geen, transkribeering, proteiin, proteiinstruktuur, kompleks) *DB_Object_Type* väljas peab vastama *DB_Object_ID* sisule. Tekst *DB_Object_Name* ja *DB_Object_Symbol* väljas võib viidata samale andmebaasi sisestusele (see ongi soovitatav) või siis üldisemale olemile. Näiteks võivad mitmed erinevad ühe geeni transkribeeringud olla eraldi annoteeritud, igaüks unikaalse transkribeeringu *DB_Object_ID*-ga, kuid samas omada sama geeni sümbolit *DB_Object_Symbol* väljas.

Annotatsioonifaili väljad

Flat-faili formaat hõlmab 15 tabulatsiooniga eralduvat välja. Tärniga väljad on kohustuslikud.

Veerg	Sisu	Näide
1.*	DB	SGD
2.*	DB_Object_ID	S0000296
3.*	DB_Object_Symbol	PHO3
4.	Qualifier	
5.*	GO ID	GO:0015888
6.*	DB:Reference(|DB:Reference)	SGD:8789|PMID:2676709
7.*	Evidence	IMP
8.	With (or) From	
9.*	Aspect	P
10.	DB_Object_Name	acid phosphatase
11.	DB_Object_Synonym(|Synonym)	YBR092C
12.*	DB_Object_Type	geen
13.*	taxon(|taxon)	taxon:4932
14.*	Date	20010118
15.*	Assigned_by	SGD

DB - geeni assotsiatsioonifaili toetav andmebaas, üks andmebaasi lühendite tabeli väärtustest.
DB_Object_ID - annoteeritava objekti unikaalne identifikaator andmebaasis.
DB_Object_Symbol - unikaalne ja kehtiv sümbol, millega sobib *DB_Objet_ID*; teisiti nimetatud geeni või proteiini jaoks saab kasutada ORF nime. Annoteeritud geeniproduktide korral saab kasutada geeni produkti sümbolit (kui olemas) või mitmed geeniprodukti annotatsioonid võivad jagada geeni sümbolit.
Qualifier - veerg lippudele, mis modifitseerivad annotatsiooni tõlgendamist. Kasutatavad väärtused on *NOT (eitus)* ja *contributes_to* (kaasa aitama). *NOT*-i kasutatakse märkimaks, et geeniprodukt ei ole seotud GO terminiga. See on tähtis juhtudel, mil termini seostamist geeni produktiga tuleb vältida. Näiteks kui proteiinil on järjestussarnasus ensüümiga (millele vastab GO:nnnnnnn), aga eksperimentaalselt pole näidanud mingit ensüümi aktiivsust, siis võib seda annoteerida kui NOT GO:nnnnnnn.

NOT-i saab kasutada ka kui osutav viide selgelt seda ütleb. Lisades *GOid*-le ette *NOT* lubab see annoteerijatel võtta seisukoha, et konkreetne geeniprodukt ei ole seotud konkreetse GO terminiga.

Kvalifitseerija *contributes_to* tähendab, et osa kompleksist on annoteeritud molekulaarse funktsiooni terminile, mis kirjeldab terve kompleksi tegevust.

GOid - GO identifitseerija terminile mis on seotud väljasisuga *DB_Object_ID*-s.
DB:Reference - unikaalne andmebaasi põhine identifitseerija viitele, mille alusel *GOid* ja *DB_Object_ID* seotakse. See võib olla kirjanduslik viide või andmebaasi kirje. Süntaks *DB:accession_number*. Ühel real võib olla mitu viite identifitseerijat.
Evidence - tõestusallikas; üks koodidest IMP, IGI, IPI, ISS, IDA, IEP, IEA, TAS, NAS, ND, IC.
With (or) From - üks järgmistest: *DB:gene_symbol, DB:gene_symbol [allele_symbol], DB:gene_id, DB:protein_name, DB:sequence_id, GO:GO_id*. Seda välja kasutatakse lisa identifitseerija märkimiseks annotatsioonidele, mis kasutavad mingeid kindlaid tõestuskoode. Kirjanduses leidub juhte, kus andmebaasi identifitseerijat ei ole (näiteks füüsiline koosmõju või järjestuslik sarnasus proteiiniga, aga ID ei ole). Annotatsioonid, kus tõestuseks on IGI, IPI või ISS ja *With* kardinaalsus on null, peaksid viitama seletusele, miks *With* väljas ei ole sisu, sest nendel tõestuskoodidel peaks kardinaalsus olema suurem kui üks.

GO:GO_id-d kasutatakse vaid juhul, kui *Evidence=IC* ja viitab kuraatori järeldusel baseeruvale GO terminile (terminitele). Neil juhtudel kasutatakse GO terminite, millest järeldus tehti, määramiseks välja *DB:Reference*.
Aspect - kas P (bioloogiline protsess), F (funktsioon) või C (rakuline paiknemine).
DB_Object_Name - geeni või geeniprodukti nimi.
Synonym - *Gene_symbol* või muu tekst.
DB_Object_Type - mis sorti objekti annoteeritakse; kas geen, transkribeering, proteiin, proteiinstruktuur või kompleks.
Taxon - taksonoomilised identifitseerijad. Kardinaalsus 1 korral geeni produkti kodeeriva liigi ID; kardinaalsus 2 korral on esimene ID produkti genereeriva liigi ID, teine ID on seda geeniprodukti kasutava liigi oma.
Date - annotatsiooni tegemise kuupäev kujul YYYYMMDD.
Assigned_by - andmebaas, mis annotatsiooni tegi. Üks andmebaasi lühendite tabeli väärtustest. Kasutatakse individuaalse annotatsiooni allika leidmiseks. Vaikeväärtus on *DB* veerus. Väärtus erineb sellest veerust juhul kui annotatsioon on registreeritud teise baasi nimega.

Oluline on, et mitmed väljad sisaldavad viiteid teiste andmebaaside sisule kujul *dbname:dbaccession* (abnimi:abligipääs). Need väljad on *GOid* (kus abnimi on alati GO), *DB:Reference, With, Taxon* (kus abnimi on alati taxon).

24. mai 2004 seisuga on 30 erinevat annotatsioonifaili.

3.3 GO tööriistad

Koostööd tegevad andmebaasid annoteerivad oma valgud või geenid GO mõistetega, pakkudes viiteid ja näidates, millised tõestusmaterjalid on saadaval annotatsiooni toetamiseks.
Kui sirvida mõnda toetavat andmebaasi, võib näha, et igal geenil või geeniproduktil on loend temaga seotud GO mõistetest. Samuti avalikustab iga andmebaas nende seoste tabeli, mis on vabalt saadavad GO ftp leheküljel[4]. Samuti võib ontoloogiaid sirvida ka laialt kasutuses olevate veebipõhiste brauseritega. Nende täisnimekiri ja ka teisi tööriistu GO-s geeni funktsiooni analüüsimiseks on saadaval GO Tools lehel[5]. Allpool kirjeldatavate tähtsamate tööriistade võimaluste koond on tabelis 2.

[4] ftp://ftp.geneontology.org/pub/go/
[5] http://www.geneontology.org/GO.tools.html

Vahend/võimalus	GO terminiga seotud geenide leidmine	GO terminiga seotud geenide arv	Geenile annoteeritud GO terminite leidmine	GO terminite otsing, definitsioonid, seosed	Üle- ja alaesindatud GO terminite leidmine geenihulga kohta	Geneetiliste andmete klasterdamine, analüüs	Otsingu täpsustamine (liigi, tõestuskoodi, ontoloogia valik)	Statistilise kasutamin väärtus, F täpsustest, standardhä Bonferron
AmiGO	•						•	
MGI GO	•	•		•				
GenNav	•							
MAPPFinder	•				•			
GoFish	•							
CGAP GO Browser	•							
TAIR	•	•						
GoMiner	•							•
MatchMiner		•	•		•		•	•
FatiGO		•	•		•		•	•
GO TermFinder			•					•
Goblet			•				•	
Ontologizer			•					
FunSpec			•					•
Onto-Express			•					
EASE			•		•			
QuickGO				•				
Term Mapper				•				
GeneMerge					•		•	
FuSSiMeG					•			
EP:GO				•		•	•	
GeneOntology@RZPD							•	
Ontology Traverser							•	•
GoSurfer								•
eGOn								
ProToGO								•
PANDORA						•		
GOBrowser								

Tabel 2. GO tööriistade võimaluste koondtabel.

3.3.1 Brauserid

AmiGo

HTML-põhine brauser [URL:AmiGO], millega saab uurida nii geeniontoloogiat kui ka annotatsioone. Kuvatakse liigendatud read, mille alguses asuvad märgid +, -, ja . tähendavad vastavalt kuvatud haru jagunemist, haru jaotuse kokku tõmbamist või et tegemist on tipuga (alamateta üksus). Ära märgitakse ka *PART OF* ja *IS A* suhted.

AmiGo brauser võimaldab päringuid üle mitme liigi andmebaasi, seega võib leida sarnased geenid eri organismidest, mis mingi kindla ontoloogiaga seotud on. Samas paistavad näiteks MGI andmebaasist pärit andmed erinevad, seda mitmel põhjusel:
1)Amigo brauser ei kuva annotatsioone IEA tõestuskoodidega,
2)kui on mitu annotatsiooni sama tõestuskoodiga, kuvatakse vaid üks neist,
3)täpsusti „tuletatud millest" (ingl. k. *developed from*) on asendatud „koos" (ingl. k. *with*) täpsustiga.

Filtreerida saab liigi ja tõestuse tüübi järgi. Edasi kuvatakse nimekiri mõistega seotud annotatsioonidest ja nende järglastest.

AmiGo saab esile tuua nende geeniproduktide järjestuse, mille andmebaasidesse on lisatud vastavad SwissPort IS lingid annoteeritud geeniproduktidele. Valitud järjestused saab kuvada FASTA väljundis (koostatakse *seqdblite* tabeli alusel). Samal ajal on olemas ka Blast päring leidmaks samasuguseid järjestusi.

Tihti tehakse eksperimente, mille tulemused näitavad, et midagi ei ole (ingl. k. *NOT*) kaasatud konkreetsesse protsessi või millelgi ei ole kindlat rolli. Annotatsioonifaili struktuur lubab eitust.

AmiGo'ga üsna sarnane on MGI GO brauser. MGI geeni detailselt kirjeldavalt leheküljelt on viited teistele andmebaasidele.

Kehtivad annotatsioonid on võrgust vabalt saadaval[6].

[6] http://www.geneontology.org/doc/GO.current.annotations.shtml

Siit saab neid kasutada seoses ontoloogiatega. Moodustatud failis on kirjas geeni produkt MGI ligipääsunumbriga, mis on ühendatud vastava ontoloogiaga (märgitud ka vastav GO ID).

MGI GO Browser

Otsida saab GO mõistet või alammõistet ja näha kõiki hiire geene, millele see viitab. Ontoloogiad saab sirvida ka terminite vaheliste suhete ja terminite definitsioonide nägemiseks ning ka hiire geenide arvu, millele see mõiste või tema alammõiste viitab. MGI GO brauser [URL:MGI] kasutab otse GO-d MGI andmebaasi, kus hiire geeni annotatsioone igal ööl uuendatakse.

DAG-Edit

See Java programm pakub GO sirvimis-, päringu- ja redigeerimisvõimalust ning seda ka igale teisele ontoloogiale, mis on DAG andmestruktuuriga. DAG-Edit [URL:DAG] uusim versioon on laetav avalikult veebilehelt SourceForge.

QuickGO EBI-s

GO brauser integreeritud EBI InterPro-ga [URL:QuickGO] võimaldab otsida GO termineid, näha nende suhteid ja definitsioone, samuti saadaval vastavused SWISS-PROT võtmesõnadele, Enzyme Classification ja Transport Classification andmebaasile või InterPro sisenditele.

EP:GO Browser

Ehitatud EBI Expression Profiler [VKKS+03] sisse. Tegu on komplektiga tööriistadest geenieksрessioonide ning teiste geeneetiliste andmete klasterdamiseks (EPCLUST), analüüsiks ja visualiseerimiseks. Expression Profileri tööriistad lasevad eraldada iga GO kategooriaga assotsieeritud geenid ja geeni ekspressiooni, reguleeriva järjendi ja valk-valk koostoime analüüsi [CMBB+03].

EP:GO [URL:EPGO] on geeniontoloogia brausimis- ja analüüsivahend. Ontoloogiat saab sirvida GO terminil klikkides, mis avab konkreetse termini alamterminid. Nii jätkates jõutakse „ontoloogiapuu" lehtedeni, mis on ühtlasi ka kõige spetsiifilisemad kategooriad.
Iga terminiga ühel real kaasneb järgmine info:

1) osalussuhe – kas IS A või PART OF,
2) GO ID – GO termini ID kujul GO:nnnnnnn, kus n on number,
3) lingid edasistele tegevustele kujul <U:L>, kus U viitab päringule mõnest andmebaasist konkreetse GO termini kohta ning L linkidele teistes brauserites konkreetse GO termini kohta,
4) GO termin,
5) termini alamterminite arv kujul (+otsesed alamterminid: kõik alamterminid),
6) termini tasand, tipu sügavus,
7) GO terminiga seotud geenide arv andmebaasis kujul [andmebaas: otse terminiga seotud geenide arv: terminiga laste kaudu seotud geenide arv].

Termini valimisel kuvatakse lisaks tema vanem ja kõik tema otsesed lapsed ning ülal kirjeldatud info nende kohta. Viimase tasandi termini korral kuvatakse kõik tema otsesed vanemad.

Iga päringu korral GO termini kohta saab valida andmebaasi (TIGR, FlyBase, MGI jt.) ning otsida

1) GO kategooriaid ja kirjeldusi,
2) assotsiatsioone,
3) väljundit geeni klastrite kontekstis,
4) ainult konkreetse GO termini kategooriat ja assotsiatsioone,
5) kõiki konkreetse GO termini alamkategooriad ja assotsiatsioone.

Iga päringu korral kuvatakse ka GO termini asukoht hierarhias, koos eelpool kirjeldatud lisadega.

TAIR Keyword Browser

Otsib ja sirvib Gene Ontology, TAIR Anatomy ja TAIR Developmental Stage termineid, lubab vaadata nende detaile ja omavahelisi seoseid [URL:TAIR]. Sisaldab linke geenidele, publikatsioonidele, mikrokiibi eksperimente ja annotatsioone, mis on seotud mõiste või selle „lapsega".

3.3.2 Tööriistad

GO TermFinder

GO TermFinder [URL:TermFinder] proovib määrata, kas vaadeldava geenide grupi annotatsioonide tase on oluline genoomi kõigi geenide annotatsioonide kontekstis. Olulisuse määramiseks uurib TermFinder gruppi geene leidmaks GO mõisteid, millele vastab kõrge hulk geene võrreldes kordade arvuga, mil mõiste on seotud teiste geenidega genoomis. Olulisuse mõõtühikuks on p-väärtus – mida väiksem see on, seda suurem on statistiline olulisus.
MGI GO TermFinderi tulemusleht on tabeli vormis, geenide hulka kirjeldavad ühised GO mõisted või nende eellased. Tabelis on GO mõisted; arv, palju kordi mõistet on nimekirjas olevate geenide annotatsiooniks kasutatud ning kordade arv, palju on mõistet kasutatud geenide annoteerimiseks kogu genoomis. Lisaks p-väärtus ja nimekiri kõigist annoteeritud geenidest.
Sarnane TermFinder tööriist on olemas ka SGD pärmi andmebaasil. Tulemused on sarnased MGI TermFinderi tulemustele, aga lisaks on ontoloogia kontekstis statistiliselt olulised punktid kuvatud graafina ja värvitud vastavalt p-väärtusele.
Taolist visualiseerimist pakub GenNav [URL:GenNav], mis lisaks otsib GO termineid ja neile annoteeritud valke ning pakub graafilise pildi mõistete GO DAG-is asumise kohta.

Term Mapper

See tööriist [URL:TermMapper] laseb seostada geenide annoteerimiseks kasutatud spetsiifilised, granulaarsed GO terminid nende üldisemate vanem-terminitega, näiteks GO Slim terminid. Valida võib nii palju GO Slim termineid, kui vaja, aga korraga ainult ühest ontoloogiast kolme seast.

FatiGO

FatiGO [ADD04,URL:FatiGO] on veebiliides, mis teostab lihtsat andmekaevandust DNA mikrokiibi jaoks, kasutades selleks geeniontoloogiat. Andmekaevandus sisaldab endas klastrile kõige iseloomulikuma geeniontoloogia termini assigneerimist. GO terminid on seotud inimese, hiire, kärbse, ussi ja pärmi geenide ja valkudega.

FatiGO kasutab Fisheri täpsustesti 2x2 tõenäosustabelite jaoks, et võrrelda kaht gruppi geene ja eraldada nimistut GO terminite jagunemisest erinevate gruppide vahel. Testi tulemused korrigeeritakse mitmekordseks testimiseks, et saada täpsustatud p-väärtus. Tulemused kuvatakse HTML ja txt formaadis. Lisaks "puu" kujul esitus geeni nimistuga assotsieeritud GO terminitest ja konkreetse terminiga seotud geenide arv.

FatiGO võimalused jagunevad üldjoontes järgmiselt:
1. Geenide hulgale GO terminite leidmine - sisestatud geeninimistule leitakse annoteeritud terminid, tulem sorteeritakse protsendi järgi.
2. Kahe geeninimistu võrdlemine - kahe geenide hulga puhul GO terminite jagunemise erinevuse uurimine, nende üle- ja alaesindatus. Kaasatakse võrdlushulk geene, milleks tavaliselt on kõik ülejäänud geenid. Samuti võib valida, kas mitmekordne test sisaldab korrigeeritud p-väärtust või ei. Tulemuseks on võrdlev graafiline vaade GO terminite ja p-väärtuse jagunemisest:

Unadjusted p-value - korrigeerimata p-väärtus; enamikel juhtudel saadakse see juhuslikke permutatsioone kasutades, kus iga p-väärtus baseerub ainult iga rea või geeni tulemustel.
Step-down min p adjusted p-value - korrigeeritud p-väärtus. Siin täidetav protseduur koos testi statistikaga on võrdne Fisheri täpsustesti 2x2 võimalikkustabeliga. Juhuslike permutatsioonide arv on 10000; juhul, kui võimalikke andmete ümberseadmisi on vähem kui 10000, kasutatakse täielikku numeratsiooni.
FDR (independient) adjusted p-value - korrigeeritud p-väärtus kasutamas Benjamini & Hochbergi FDR protseduuri.
FDR (arbitrary depandent) adjusted p-value - korrigeeritud p-väärtus kasutamas Benjamini & Yekutieli FDR protseduuri.

Üheaegselt testitakse mitmeid nullhüpoteese (iga GO termini kohta üks) termini sageduse mitteerinemise kohta igas klastris.

MAPPFinder

MAPPFinder [URL:MAPPFinder] on lisaprogramm, mis töötab koos GenMAPP ja geeniontoloogiaga, et identifitseerida globaalseid bioloogilisi trende geneetiliste andmete väljendumises. MAPPFinder ühendab andmete mikrokiibi (selle vastamisel kasutaja defineeritud kriteeriumile 'olulise' geeni väljendumise

muutumise jaoks) iga terminiga geeni ontoloogia hierarhias(t), arvutades selleks iga GO ontoloogia termini kohta muutunud geenide protsendi. Siis arvutab MAPPFinder vanem-termini ja kõigi selle laste suhtes muutunud geenide kumulatiivse summa, andes täieliku pildi konkreetse GO terminiga seotud geeni ekspressioonide muutustest. Seda protsenti ning keskmisel ja hüpergeomeetrilise jaotuse standardhälbel põhinevat z-skoori kasutades saab kasutaja filtreerida tulemusest terminid, millel on oodatust kõrgem arv geeni ekspressiooni muutusi. Mõlemad MAPPFinderi tulemused eksporditakse tekstifailina ja kuvatakse MAPPFinder brauseris, lastes kasutajal kiiresti identifitseerida need bioloogia valdkonnad, mis näitavad korreleeruvaid geeni väljendumise muutusi. Klikkides GO terminile MAPPFinder brauseris avaneb vaade GenMAPPis, kus on nimekiri kõigist selle terminiga seotud geenidest. Kokkuvõttes loob MAPPFinder geeni ekspressiooni profiili üle kõigi GO-s esindatud bioloogia valdkondade, lubades kasutajal näha, kus esinevad andmetes kõige suuremad korreleeritud geeni väljendumiste muutused.

EASE

Eraldiseisev NIAID tarkvarapakett Windows operatsioonisüsteemil kasutamiseks, mis on kasulik antud geeninimekirja dominantse bioloogilise 'teema' kokkuvõtteks. EASE [URL:EASE] arvutab üleesindatuse statistika iga võimaliku geeniontoloogia termini kohta vastavalt kõikidele geenidel andmehulgas (eksperimenti kaasatud geenid).

EASE on mõeldud edasiseks abiks uurijatele geenide grupi bioloogia tundma õppimiseks. Need grupid on moodustatud sarnaste väljendumisomaduste põhjal, näiteks käitumine aja jooksul, korrelatiivsus mõne eksperimentaalse parameetriga, käitumine eksperimendi käigus. Hetkel koondab EASE kolm tööriista:

Lingi loomise (ingl. k. *link-out*) vahend laseb uurijatel kasutada uusi *online* analüüsi tööriistu nende avaldamisel. See on põhiliselt ette antud geenide põhjal tavalise URL-i moodustamise meetod. Tänu lihtsale konfiguratsioonile saab EASE'i seada liidesena lemmik *online* tööriistade hulka. Niiviisi saab geeninimistu laadida ühe korra ja kasutada seda mistahes arvu *online* vahenditega, ilma iga tööriista jaoks uut geeniloendit moodustamata.

Analüüsi vahend leiab geenide kategooriad, mis on valitud geenide hulgas üleesindatud võrreldes sellega, mis on esindatud mikrokiibil või terves liigi genoomis. Sellised üleesindatud kategooriad tähistavad antud listi bioloogilisi 'teemasid'.

Annotatsiooni tööriista kasutatakse kõiki geene kirjeldava informatsiooni tabeli kiireks genereerimiseks, ilma et iga geeni jaoks oleks vaja kasutada erinevaid veebilehti.

GoMiner

GoMiner [URL:GoMiner] on tööriist *omic* andmete (sisaldab andmeid geeni ekspressiooni mikrokiipidelt) bioloogiliseks interpreteerimiseks. *Omic* eksperimendid genereerivad tihti loendeid suurest hulgast geenidest, mis erinevad näidetes avaldumises ning tõstatavad küsimuse kõige selle bioloogilise tähenduse üle.
GoMiner mõjutab geeni ontoloogiat, et identifitseerida neis listides esinevaid bioloogilisi protsesse, funktsioone ja komponente. Selle asemel, et analüüsida mikrokiibi tulemusi "geen geeni järel" meetodil, klassifitseerib GoMiner geenid bioloogiliselt koherentsetesse kategooriatesse ja määratleb need kategooriad. Pilguheit GoMineri läbi võib luua hüpoteese edasise uurimise juhtimiseks.

MatchMiner

MatchMiner [URL:MatchMiner] on tööriistade komplekt, mis võimaldab teisaldada ühildumatud ID-d sama geeni kohta. Et teha kindlaks, kuidas erinevad ID-d seoses on, kasutatakse USCS, LocusLink, Unigene ja OMIM andmeid. MatchMiner koosneb kolmest erinevast veebipõhisest vahendist: interaktiivne otsing, komplekti otsing ja komplektide ühendamine.

GeneMerge

GeneMerge [URL:GeneMerge] on mitmekülgne geneetika programm, mida saab kasutada suure hulga funktsionaalsete geneetiliste andmete analüüsiks. Täpsemalt on GeneMerge kasulik mikrokiibi andmete ja teiste suurte bioloogiliste andmehulkade analüüsiks.
GeneMerge tagastab funktsionaalsed ja kategoorialised geneetlised andmed antud geenihulga kohta ja pakub statistilisi järjestatud tulemusi konkreetsete funktsioonide või andmehulkade kategooriate üleesindatuse kohta. Teiste hulgas on GeneMerge võimeline teostama reguleeriva ja metaboolse raja analüüsi, populatsiooni geneetilisi hüpoteeside teste, ristandeksperimentide võrdlusi ja kromosoomilise klasterdamise teste. Andes ette hulga geene võib GeneMerge näiteks öelda, kas need geenid on statistiliselt konkreetses funktsiooni või

biokeemia klassis üleesindatud, genoomi regioonis klasterdatud või assotsiatsioonis kindla RNA või deletsiooni fenotüübiga.
GeneMerge'i eelis teiste samalaadsete programmidega võrreldes on, et pole piiranguid andmete analüüsimisel eelgrupeeritud geeni assotsiatsioonide andmehulkade vaatekohast. Võib mahalaadida või luua geeni assotsiatsiooni faile analüüsimaks andmeid limiteerimata arv vaatenurgast.

GOblet

GOblet [HGL03,URL:GOblet] annoteerib GO terminite baasil anonüümset cDNA- või valgujärjendit. Oma päringut saab kasutaja täpsustada valides DNA või valgu, määrates kasutatava andmebaasi ja nn. headuse mõõdu. Päringu teostamisel luuakse lennult unikaalne URL, mida võib järjehoidjaga varustades kasutada ka hiljem. Kuigi praegu lubatakse sessiooni ajal üht päringuseeriat otsingu kohta, võib alustada palju otsinguid, mis on ligipääsetavad nende URL-ide kaudu. Hetkel säilitatakse süsteemis tulemusfaile vähemalt üks nädal.
Tulemuse pealehel kuvatakse päringu järjend uuesti (see on kasulik, kui kasutaja on alustanud mitu otsingut) ja tabamused on tabelis tähtsuse järgi sorteeritud. Lisaks on nähtaval proteiini kirjeldus vastavalt allikaks olevale andmebaasile ja liigile ning lingid originaal BLAST väljund failile, nagu ka algdokumentidele baasides SWISS-PROT, FlyBase, ENSEMBL jne. Iga vastavuse korral näidatakse sellega assotsieeritud GO ID-de täielik nimekiri ja lingid EBI QuickGO brauserile.
Tulemuslehel kuvatakse ka kõigist valkudele vastavatest GO terminitest konstrueeritud puu. Selle koondesituse eelis on, et kõige tähtsamad harud on kergesti avastatavad, samuti on olemas ka toetavad proteiinid, nii et iga informatsioonikild on kergesti kontrollitav.

GARBAN

GARBAN [URL:GARBAN] on vahend cDNA mikrokiipidelt ja valgutehnikatest tulenevate andmete analüüsiks ja sagedaseks funktsionaalseks annoteerimiseks. GARBAN on seadistatud bioinformaatika tööriistadega, et kiiresti võrrelda, klassifitseerida ja graafiliselt esitada paljusid andmehulki, püüdes lihtsustada molekulaarsete märgendite identifitseerimist patoloogia ja farmakoloogia uuringutes. GARBANil on viited peamiste geneetika ja valgu andmebaasidele (Ensembl, GeneBank, SWISS-PROT jt.) ja järgib GO konsortsiumi kriteeriume ontoloogia klassifitseerimisel.

FunSpec

FunSpec (Functional Specification) [URL:FunSpec] kasutab sisendina listi pärmi geene ja väljastab kokkuvõtte funktsiooni klassidest, rakulisest lokalisatsioonist, proteiini kompleksidest jms., millega nimistu seoseid leiab. Lisaks on koondatud palju avaldatud andmeid nende seoste võrdlemiseks. Olemas on lingid publikatsioonidele.
Hüpergeomeetrilist jaotust kasutades välja arvutatud p-väärtus näitab tõenäosust, et antud nimekirja ühisosa suvalise funktsionaalse kategooriaga tekib juhuslikult. Bonferroni parandus eraldab p-väärtuse künnise, mis osutus oluliseks individuaalse testi jaoks läbiviidud testide arvu põhjal ja niiviisi leitakse ebaolulised väärtused multi-testimisel üle andmebaasi kategooriate. Pärast Bonferroni parandust kuvatakse ainult need kategooriad, mille rikastumise võimaluse tõenäosus on väiksem kui p-väärtus/#CD, kus #CD on valitud andmebaasi kategooriate arv. Ilma Bonferroni korrektiivita on kuvatud kõik kategooriad, mille puhul seesama rikastumise tõenäosus on väiksem p-väärtuse künnisest individuaalses testis.

GoSurfer

GoSurfer [URL:GoSurfer] kasutab geeniontoloogia (GO) informatsiooni genoomi arvutustest või mikrokiibi analüüsist saadud geeni hulkade analüüsiks. Tegemist on interaktiivse andmete kaevandamise tööriistaga. See assotsieerib kasutaja sisestatud geenid GO terminitega ja visualiseerib need terminid hierarhilise puuna. Kasutaja saab manipuleerida väljundpuuga erinevaid võtteid kasutades, nagu künniste seadmine või statistiliste testide kasutamine. Leitud olulise tähtsusega GO terminid saab esile tõsta. Kõik seonduv informatsioon on eksporditav teksti või graafilisel kujul.

Ontologizer

Java põhine Ontologizer [URL:Ontologizer] laseb kasutajail automaatselt genereerida HTML-lehti, mis sisaldavad kokkuvõtet nii funktsionaalsetest annotatsioonidest ühe või mitme geenigrupi (klastri) kohta kui ka detailselt iga geeni kohta selles klastris. Kuigi Ontologizer on peamiselt mõeldud mikrokiibi klasteranalüüsi arendamise hõlbustamiseks, saab seda kasutada ka mistahes geenide või geeni produktide, mille kohta GO annotatsioonid olemas on, funktsionaalse annotatsiooni edasi arendamiseks.

Ontology Traverser

Ontology Traverser [URL:OntologyTraverser] (ontoloogia läbija) on veebipõhine tööriist mikrokiibi geenide *listi* rikastamise analüüsiks. Traverser töötab geeni nimistuid geeni ontoloogia andmestruktuuriga ühendades ja leides rikastumisstatistika igale GO tipule, mis asub mistahes liikumisteel GO annotatsioonini. Analüüsis arvestatakse kõiki GO tasemeid ja kasutatakse tervet GO andmestruktuuri seotust.
Arvutatav statistika iga GO tipu kohta sisaldab listi ja massiivi esinemissagedust, rikastumise p-väärtust ja proovide identifitseerijaid iga tippu läbivate annotatsiooni radadega. Tagastatakse tabelid täistulemuse ja oluliste tulemuste kohta. Täistulemuse tabel sisaldab kirjeid kõigi mitte nulliga võrduvate annotatsiooniteedega tippude kohta geeni loendist. Oluliste tulemuste tabel sisaldab tippe, mille p-väärtus on väiksem kui 0,05, vastavalt uuritavate loendite hüpergeomeetrilisele mudelile.

eGOn

eGOn (explore Gene Ontology) [URL:eGOn] kasutab geeni ontoloogia informatsiooni üle genoomi teostatud arvutustest või mikrokiibi analüüsist saadud geeninimistute analüüsis. GO infot saadakse LocusLinkist ja iga GO kategooria on puuna visualiseeritud. Kasutaja saab puud "mõjutada" mitmete vahenditega.
Võrrelda saab omavahel mitmeid erinevaid geenihulki. Et määrata kindlaks, millised GO rajad (selles puus) on seotud olulise hulga geenidega kindlas geenide komplektis, viiakse läbi statistilised testid. Kõik puude joonised ja tulemusinfo on eksporditav. Sisendiks on tabulatsiooneraldusega geeni identifitseerijat sisaldav tekstifail.

Onto-Tools

Siia kuulub hulk vahendeid [URL:Onto]. OntoGate on ligipääsuga GenomeMatrixile, kus on ontoloogia terminite ja nendega seotud lisa andmete sisestused, mille abil leida eri liikide geenid, mis on terminitega seotud. OntoGate sisaldab annoteeritud geenide aminohapete järjendite BLAST otsingut.
Onto-Express [DKM0+03] transleerib GO termineil põhinevaisse funktsionaalsetesse profiilidesse need geeninimistud, mis osutusid geeni ekspressiooni eksperimentides erinevalt reguleeritavateks. Profiilid konstrueeritakse kategooriatele nagu biokeemiline ja molekulaarne funktsioon,

bioloogiline protsess, rakuline roll ja komponent, kromosoomi asukoht. Iga kategooria kohta arvutatakse statistiliselt oluline väärtus ja tulemused kuvatakse graafiliselt. Onto-Express on disainitud saadaval funktsionaalsete annoteerimisandmete kaevandamiseks ja abiks oluliste bioloogiliste protsesside leidmisel.
Onto-Design abil saab moodustada tavalisi mikrokiipe, valides hulga UniGene'i klastrite ID-sid, mis esindavad GO terminitega kirjeldatud bioloogilist protsessi.
Onto-Compare laseb kasutajal määrata iga massiiviga seotud funktsionaalse nihke ja aitab määrata parima mikrokiibi, mis GO termineid kasutades seda bioloogilist fenomeni kirjeldab.
Onto-Translate on veebipõhine tööriist, mis aitab identifitseerida sama informatsiooni üle erinevate andmebaaside ja vähendada suvaliste geeninimistute ülekoormust.
Onto-Miner laseb uurida erinevaid avalikke bioinformaatika andmebaase klooni ID-d, UniGene'i geeni sümbolit, LocusLinc id-d jne. kasutades ja viia läbi seeria päringuid kogu geeninimistut kasutades. Seda saab kasutada suvaliste geeniloendite kohta detailse geeni info saamiseks.

Veel üldisemad vahendid on näiteks GOBrowser (GO terminid on vaadeldavad Exploreri laadses brauseris), Manatee (kiire geenide identifitseerimine, funktsionaalsete seoste leidmine), PubSearch (geenide otsing ja annoteerimine artiklite võtmesõnade järgi), SOURCE (kogub informatsiooni mitmetest avalikest andmebaasides).

II Praktiline osa

4. Geeniontoloogia kasutamine seoste leidmiseks

4.1 Mudel

Mudeli moodustavad

1)ontoloogiad – organismist sõltumatud struktureeritud sõnastikud,
2)annotatsioonid – organismispetsiifilised ja kirjeldavad geeni produkti molekulaarset funktsiooni, bioloogilises protsessis osalemist ja rakulist paiknemist.

4.1.1 Perli andmestruktuurid

Üks võimalus vajalike andmete leidmiseks ja hoidmiseks on Perli andmestruktuurid: paisktabelid, listid. Kuna kasutatavad andmed, ontoloogiad ja assotsiatsioonifailid, on saadaval lihtsal *flat*-fail kujul, siis sealt Perliga nende mällu lugemine ja sobivatesse struktuuridesse paigutamine on suhteliselt vähest aega nõudev.

4.1.2 MySQL andmebaas

Teine võimalus andmete hoidmiseks on andmebaas. Siin tuleb silmas pidada andmete tähtsust ja nende säilitamist eraldi tabelites kiirete operatsioonide tarvis. Antud juhul läheb selleks vaja kahte tabelit - neis ühes ontoloogiad, teises assotsiatsioonid. Ontoloogiad on tabelis kirjetena termin, vanem (mõlemad tüüpi *char*) ja nendevaheline kaugus (tüüp *int*); assotsiatsioonide tabeli kirje koosneb termini ID-st, selle vastavast geeniproduktist, geeniprodukti sünonüümidest, vastava andmebaasi objekti ID-st ja tõestuskoodist (kõik tüüpi *char*), mille alusel need kaks seotud on (tabel 3). Tõestuskoodi tabelisse

lisamisel on tegelikult vaid filtreerimise eesmärk. Neist kahest tabelist on võimalik saada vajalikud andmed leiduda võivate seoste määramiseks.

```
mysql> describe assots;
+--------+----------+------+-----+---------+-------+
| Field  | Type     | Null | Key | Default | Extra |
+--------+----------+------+-----+---------+-------+
| term   | char(10) | YES  | MUL | NULL    |       |
| ab_obj | char(8)  | YES  |     | NULL    |       |
| geen   | char(15) | YES  | MUL | NULL    |       |
| g_syn  | char(30) | YES  |     | NULL    |       |
| t_kood | char(3)  | YES  |     | NULL    |       |
+--------+----------+------+-----+---------+-------+
5 rows in set (0.00 sec)


mysql> describe termin;
+---------+----------+------+-----+---------+-------+
| Field   | Type     | Null | Key | Default | Extra |
+---------+----------+------+-----+---------+-------+
| term    | char(10) | YES  | MUL | NULL    |       |
| t_vanem | char(10) | YES  | MUL | NULL    |       |
| vahe    | int(11)  | YES  |     | NULL    |       |
+---------+----------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

Tabel 3. MySQL tabelite ASSOTS ja TERMIN kirjeldused.

4.2 Andmete lugemine ja töötlus

4.2.1 Ontoloogiad

Geeniontoloogia terminid on kirjeldatud OBO *flat*-failis *gene_ontology.obo*. See on vabalt saadaval ning käesolevas töös aluseks kõigele GO terminitega seotule. Nimetatud faili parsimise käigus moodustab *goparent.pl* Perli paisktabelid (ingl. k. *hash*) GO, GO_parents ja GO_parkaugus.
Neist esimene, GO, sisaldab omakorda *hash*e, millest igaüks sisaldab kõiki OBO failist sisse loetud andmeid ühe GO kategooria kohta. Sisemiste *hash*ide võtmeteks saavad stantsi märgendid (kirjeldatud punktis 3.2.3.2). Lisaks

moodustatakse igale terminile loetelu (ingl. k. *list*) kõigist tema otsestest vanematest ja lastest.
GO_parents on *hash,* mille võtmeteks on GO terminite ID-d ning väärtusteks loetelud kõigist tema vanematest kuni juureni välja.
GO_parkaugus on GO_parents laiendus – igale ta võtmele, milleks on GO terminite ID-d, vastab *list,* kus termini vanemast järgmisel kohal on selle kaugus võtmeterminist. Seda paisktabelit kasutatakse MySQL andmebaasis tabeli TERMIN täitmisel.

4.2.2 Assotsiatsioonid

Assotsiatsioonide määramiseks on mõeldud annotatsioonifailid, mis sarnaselt ontoloogiafailidele on vabalt kättesaadavad. Annotatsioonifaili parsimisega moodustab *geenid.pl hash*i ASSOTS, mille võtmeteks on kõik GO ID-d ja vastavateks väärtusteks nendega otseselt seotud geenide *list*.
Kuna meid huvitavad kõik geenid, mis on ühe GO kategooriaga seotud, see tähendab nii otse kui ka *IS_A* ja *PART_OF* relatsioonide („laste") kaudu, on asjakohane genereerida *hash* ALL_ASSOTS. Seal on võtmeteks kõik GO ID-d, mis mingi osalussuhte kaudu on geeniga seotud. Neist igale vastab *list* geenidest, mis on konkreetse GO terminiga kas siis otse või kaudselt seotud. Selleks loeb *k6ik_geenid.pl* sisse OBO faili (ontoloogias orienteerumaks, moodustatakse paisktabelid GO ja GO_parents) ning kasutades *hash*i ASSOTS seob iga GO terminiga kõik temaga seotud geenid, mis annotatsioonifail sisaldab.
MySQL andmebaasis täidab vastava tabeli ASSOTS *abaas.pl*, mis parsib konkreetse annotatsioonifaili ning DBI moodulit kasutades MySQL käskudega tabeli väljad väärtustab.

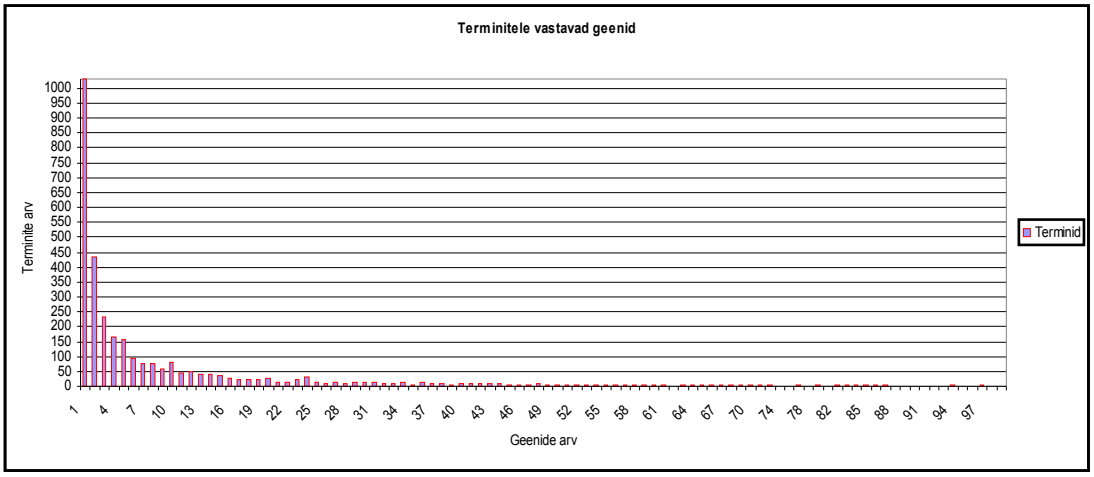
4.3 Oluliste seoste leidmine pärmi geenide näite varal

Antud kontekstis on tähtis leida päringusse antud geenide ja GO terminiga assotsieeritud geenide ühisosa. Selle ühisosa suurus vaadeldava hulga suhtes määrab, kas tegu on olulise või juhusliku (mittetähtsa) seosega. Seejuures tuleb arvestada ontoloogiate hierarhilisust – kui geeniprodukt on terminiga seotud, siis on ta seotud ka kõigi selle termini eellastega. Ühisosade kaudu seoste olulisuse leidmisel on järelikult kaks aspekti: 1) ühisosa suurus konkreetse terminiga

seotud geeniproduktidega ja 2) ühisosa suurus geeniproduktidega, mis niiöelda koonduvad üldisema termini alla seoses *IS_A* või *PART_OF* relatsioonidele. Teise variandi puhul on oluline ka leitud vanema tase, kui kaugeks vanemaks ollakse, sest lõpuks koonduvad kõik terminid ühe alla kolmest ontoloogiast.

Seoste leidmiseks kasutasin näitena toidupärmi (*Saccharomyces cerevisiae*) annotatsioonifaili *gene_association.sgd* (seisuga 12.aprill 2004). Primaarseks geeniks valin esimesel kohal oleva, teisi vaatlen kui sünonüüme. Kokku on primaarseid geene 6460. Paisktabel ALL_ASSOTS sisaldab võtmetena pärmi geenidega seotud GO terminite ID-sid, neid on kokku 3414.

GO terminite hulka ja nendega nii otse kui ka järglaste kaudu seotud geene illustreerib joonis 3.



Joonis 3. GO kategooriatega seotud assotsiatsioonide hulgad 1-100. Alates hulgast suurusega 100 on need seotud keskmiselt 1,4 terminiga.

4.3.1 Ühisosa leidmine: totaalne võrdlus

Üheks geenihulkade ühisosa leidmise võimaluseks on absoluutselt kõikide terminite, nendega seotud geenide, võrdlemine geenidega päringus. Seega tuleb kõiki iga GO terminiga assotsieeritud geene võrrelda päringugeenidega ning leida ühisosa suurus:

$Q = [g_1, g_2, \ldots, g_n]$, kus Q on päringusse olevate geenide hulk, milles on n geeni;

$T = [g_1', g_2', \ldots, g_m']$, kus T on GO kategooriaga seotud geenide hulk, milles on m geeni.

Leida iga $Q \cap T_i$, $i \in \{1, 2, \ldots, GO\ kategooriate\ arv\}$.

Meetod 1. open(GENES, "klastrid.txt") or die "Ei saanud lugeda: $!\n";

…

while(<GENES>) {

@L = split ;

@l=@L[0..$cluster_size];

foreach $key (keys %NR_RIDA){

…

$isect_size=*find_intersection(@l, @{$NR_RIDA{$key}});*

…

}

}

Kuna geeniontoloogia termineid vaadeldakse puuna ja iga terminiga assotsieeritud geenid on ühtlasi tema ja ta lastega seotud geenide ühend, on ilmne, et ülal nimetatud ühisosa otsimisel tühihulgani jõudes pole mõtet enam ühisosa otsida selle termini lastega assotsieeritud geenide seast. Siit ka järgmine meetod.

4.3.2 Ühisosa leidmine: sügavuti läbimine

Hulkadest ühisosa leidmise operatsioon selle meetodiga on iseenesest sama eelmise, totaalse võrdluse, meetodiga. Erinevus seisneb siin selles, et võrdlusesse kaasatakse vaid need terminitega seotud hulgad, kus ühisosa tõesti on ette näha. See tähendab, et kui leitakse esimene termin, millega seotud geenide ja päringugeenide hulga ühisosa on tühihulk, lõpetatakse ühisosa otsimine nimetatud termini järglastega assotsieeritud geenide hulkades, kus selle puudumine on ilme. Samas peetakse meeles ka kõik tipud, mida juba võrreldud on ning seega võimalikku tühihulki andvat alampuud ei läbitagi.

Sama meetodit on võimalik kasutada ka efektiivsemalt. Kui puus liikumise katkestavaks künniseks valida õige suurus, väheneb otsimisaeg veelgi. Selleks künniseks on ühisosa suurus, mis on minimaalne olulise hulkade kattuvuse leidmiseks.

Meetod 2.

```
open(GENES, "klastrid.txt") or die "Ei saanud lugeda: $!\n";
while(<GENES>) {
                                                    …

                                        @L = split ;

                                @l=@L[0..$cluster_size];

                                  add_@l_to_%GENES;
}

$threshold=…;
foreach $key (keys %GENES){
        %seen=();

                                $root='GO:0000000';

                                  tree_search($root);
                                                                              }

sub tree_search
{
        $in=@_[0];
        if($seen{$in}++){return;}
        $isect_size=find_intersection(@l, @{$NR_RIDA{$key}});
        if($isect_size<=$ threshold){return;}
        foreach $child ( @{ ${ $GO{ $in }}{ 'children' } } ){

                                                                yhisosa($child);
        }
}
```

4.3.3 Totaalne võrdlus vs sügavuti läbimine

Mõlema meetodi kasutamisel on omad arvutuslikud ja seega ajalised erinevused. Meetod 1 korral võrreldakse mööndusteta kõiki GO terminitega seotud hulki, meetodis 2 loobutakse edasistest võrdlustest GO termini järglastega, kui nimetatud termini ja päringugeenide hulga ühisosa osutub tühihulgaks. Selleks peab programmis olema vastav klausel, mis leitud tühihulga korral võrdlused selle haruga lõpetab. Lisaks seatakse iga uue päringu korral esiti GO kategooriate „lipud" (ingl. k. *flag*) nulliks ning neid läbima hakates omakord üheks. Tekib küsimus, kas meetod 2 oma lisatingimusega kaalub üles meetodi 1, mis kulutab aega kõikide tippude läbimisele.

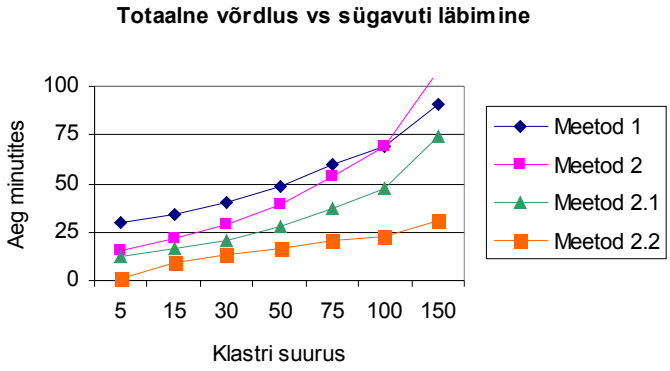
Meetodite võrdlemiseks kasutasin lisaks pärmi annotatsioonifailile *gene_association.sgd* faile *nimed.txt* ja *klastrid.txt*. Andmeid neist viimases failis on kirjeldatud allikas [VBJR+00]. Tegemist on 6221 toidupärmi geeniga, mille ekspressiooniprofiilid on kirjeldatud 80 bioloogilise katse raames. Sellelt geenide hulgalt on leitud 6221 klastrit (ingl. k. *cluster*) nii, et iga klastri keskpunktiks on valitud järjekordne geen. Klaster on sarnaste objektide kogum, mis on sarnased antud klastri sees. Klastri keskpunktile järgnevate geenide järjestus sõltub nende ekspressiooniprofiili sarnasusest esimese geeni profiilile 80 katse käigus, mille mõõduks on kasutatud koosinuskaugust 80-mõõtmelises ruumis. Koosinuskaugus d_{rs} kahe vektori v_r ja v_s vahel 2-mõõtmelises ruumis arvutatakse järgnevalt.

$$d_{rs} = 1 - \frac{x_{rx} x_{sy}}{(x_{rx} x_{ry})^{\frac{1}{2}} (x_{sx} x_{sy})^{\frac{1}{2}}}$$

Nimed.txt sisaldab kaheelemendilisi ridu, kus esimesel kohal numbrid 0 kuni 6220 ja teisel kohal neile vastavad pärmi geenid. Neist numbritest on 6221-elemendilised read failis *klastrid.txt*, kus ridade esimesteks elementideks on numbrid nimetatud vahemikust. Igast sellisest reast saab moodustada omakorda erineva suurusega geeniklastreid, millele otsida iseloomulikku GO terminit ühisosa leidmise teel. Seda faili ja ühisosa leidvat programmi muutes saab läbi viia katseid erinevate andmehulkadega töötamise tarvis.
Ühisosade leidmiseks võrdlevad mõlemad meetodid programmidest *total_search.pl* ja *tree_search.pl* kõiki GO terminiga seotud geene

päringugeenidega. Kuna neist viimased on numbrilisel kujul, teisendas *tee_vordluslist.pl hash*i ALL_ASSOTS igale võtmele vastava *list*i sisu numbriliseks *list*iks (hoitakse *hash*is NR_RIDA teegi *lib.storable* abil), kasutades selleks *nimed.txt* andmeid. Moodustatud paisktabel NR_RIDA sisaldab sellevõrra vähem andmeid, kui palju ei leita paisktabeli ALL_ASSOTS võtmete väärtustele vastavusi failist *nimed.txt*. Kokku sisaldab *hash* NR_RIDA 3389 võtit. See ei ole aga taksitus meid huvitavate ühisosade leidmisel.

Nagu punktis 4.3.2 mainitud saab sügavuti läbimise meetodit kiirendada künnise ehk minimaalse ühisosa valikuga. Selles veendumiseks valisin katsesse meelevaldselt künnised 2 ja 10.
Edasi toimus numbriliste loendite võrdlus kirjeldatud kahel meetodil, mille tulemusi illustreerib järgnev joonis.



Joonis 4. Meetodite 1 (valikuta kõigi GO kategooriate võrdlus) ja 2 (GO hierarhia järgi sügavuti läbimine künnisega 0) ning neist viimase puhul künniste 2 (meetod 2.1) ja 10 (meetod 2.2) rakendamisel töötamise kiirus 6221 geeniklastri võrdlemisel GO terminitega seotud geenidega.

Osutub, et suuremate klastrite korral jääb meetod 2 oma kiiruses alla meetodile 1. Seda seletab asjaolu, et teatud klastrisuuruseni jõudes kontrollib meetod 2 kõiki GO tippe, sest ühisosa on alati vähemalt 1. Sellest hetkest hakkab kiirust kahandama ka iga tipu eelneva läbikäimise kontrollimine ja vajadusel vastavate „lippude" seadmine. Meetodi 2 töötamise ajakulu vähendab märgatavalt künnise seadmine (näited meetoditega 2.1 ja 2.2). Selle sobivat suurust püüan hiljem leida.

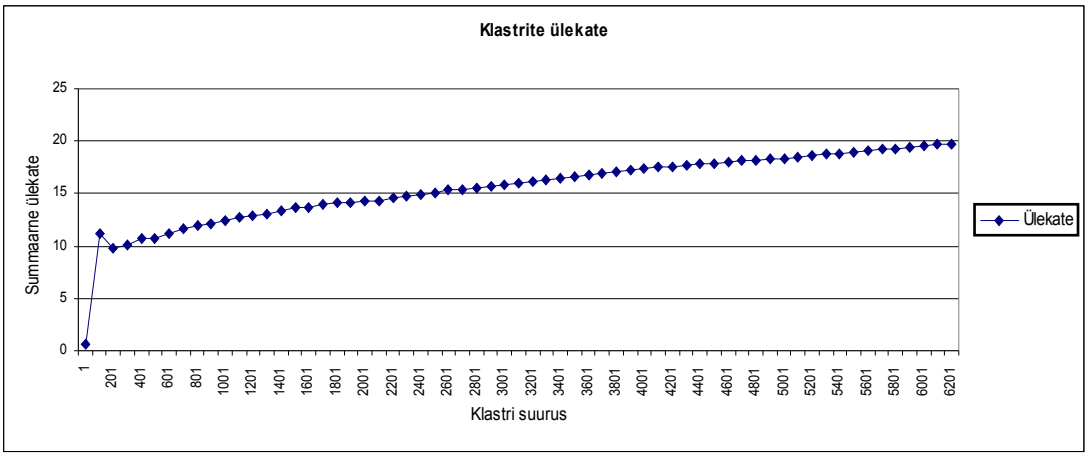
Huvitav on märkida, et vähema hulga klastrite (1 kuni 1000) korral käituvad mõlemad meetodid vastupidiselt vastavalt sisestatud andmetele. Olgu päringugeenid kujul m x n, kus m on ridade (st. päringute) arv ja n geeniproduktide arv selles reas. Ilmneb, et juhul m < n osutub efektiivsemaks meetod 1; juhul m > n on kiirem meetod 2. Juhul m = n osutus samuti kiiremaks meetod 2, kuid see vahe ei olnud enam nii tähelepanuväärne, kui eelmistel juhtudel.

4.3.4 Meetodite efektiivsuse tõstmine

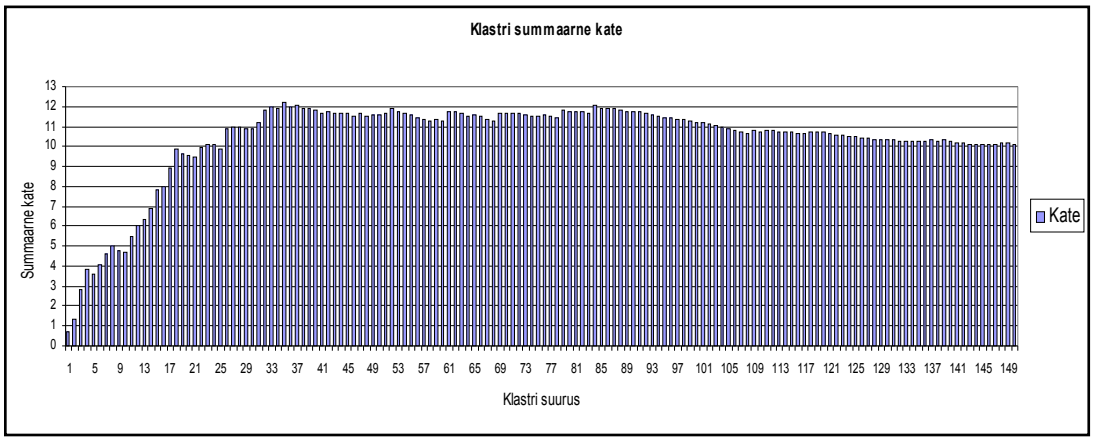
Vastavalt ülesande püstitusele leida olulisis seoseid geeniklastrite ja GO terminite vahel, on kaks viisi meetodite 1 ja 2 efektiivsuse tõstmiseks. Mõlema viisi testimiseks kasutan esimest rida failist *klastrid.txt*, mille järgi saab vaadelda 6221 erineva suurusega geeniklastrit. Kuna kõik read nimetatud failis on moodustatud samal põhimõttel, üldistab konkreetse rea valik ka kõiki teisi ridu.

4.3.4.1 Klastri suurus

Ideaalne ühisosade ja seeläbi oluliste seoste otsimine hõlmab kõikvõimalike suurustega klastrite võrdlemist GO termini geenidega. „Laiahaardelist" päringut üle võimalike, suuruse poolest erinevate klastrite iseloomustavad joonised 5 ja 6.



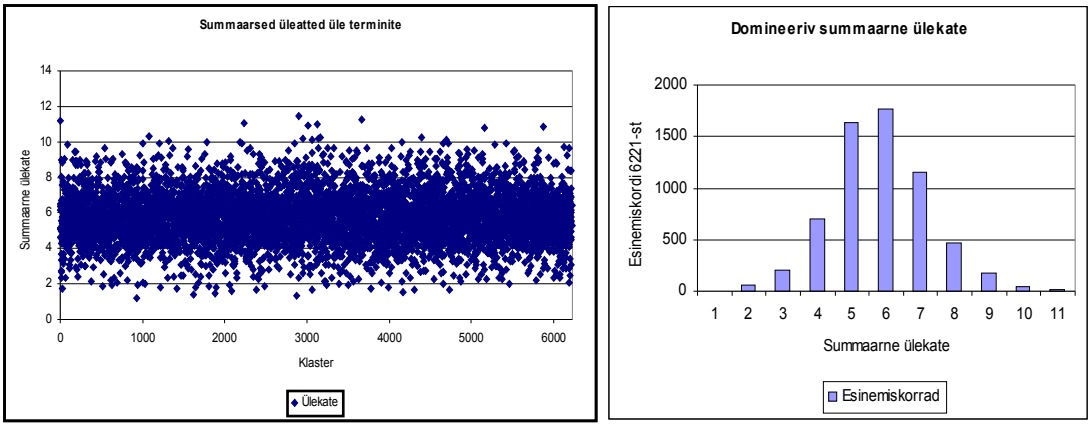
Joonis 5. Klastrite suurusega 1-6221 summaarsed suhtelised ülekatted kõikide GO terminitega.



Joonis 6. Klastrite suurusega 1-150 summaarsed suhtelised ülekatted kõikide GO terminitega.

Ilmneb, et summaarsed suured ülekatted tekivad esiti klastrite suurusega 26-105 puhul. Summaarsete ülekatete statistiline mood selles vahemikus on 11,69.

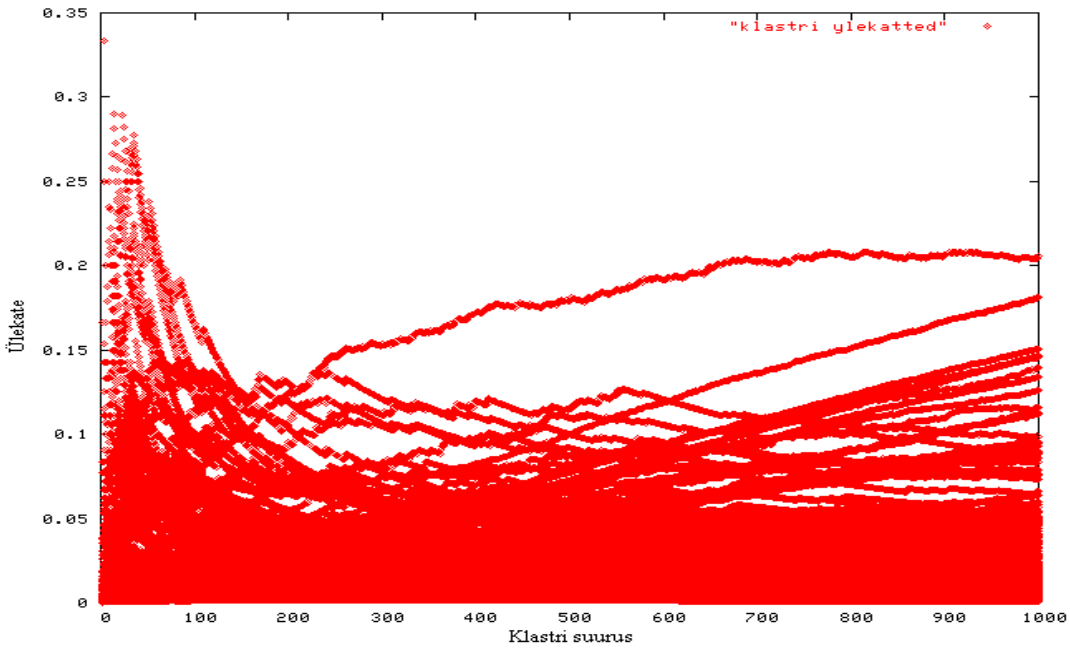
Et näidata suhteliste ülekatete koonduvust sama suurusega klastrite korral, tegin eksperimendi kõigi 6221-e klastriga võttes klastri suuruseks 100. Iga selle klastri kohta leidsin summaarse suhtelise ülekatte kõikide GO kategooriatega. Selgus, et sellise klastrisuuruse korral koonduvad summaarsed ülekatted väärtuste 5 ja 6 ümber (joonis 7). Hajuvust domineeriva ülekatte ümber seletab geeniontoloogia hierarhilisus – GO kategooriaga suure ülekatte andev klaster teeb seda ka konkreetse kategooria lähimate eellaste ja järglaste korral.



Joonis 7. Summaarsed suhtelised ülekatted ja nende koondumine 6221 klastriga suurusega 100 korral üle kõigi GO kategooriate.

Kuna ülekatte suurenemine klastri suurenedes on loogiline, võib seoste leidmiseks parim klastri suurus jääda just vahemikku 26-105. Hüpoteesi

kinnitamiseks on joonis 8, kus klastrid suurusega 1-1000 (tegemist on ühe klastrireaga, mida vastavalt ühe komponendi võrra suurendati; esineme rida failist *klastrid.txt*) andsid GO terminitele suuri ülekatteid just selles vahemikus.



Joonis 8. Ülekatted GO kategooriatega klastritega suurusvahemikus 1-1000.

Ei saa mitte märkamata jääda, kuidas üks või mitu hierarhiliselt lähedal asetsevat GO terminit stabiilselt teistest paremaid ülekatteid saavad. Selleks osutub *Component: mitochondrion* (GO:0005739), ontoloogiapuus sügavusel 6 (arvestusega, et juure GO:0003673 *Gene Ontology* sügavus on 1). Alates klastrist suurusega 600 saab häid ülekatteid selle eellane *Component: cytoplasm* (GO:0005737), ontoloogiapuus sügavusel 5. Selgub, et pidev suhtelise ülekatte kasvamine (samal ajal suureneb ka klaster) tekib GO kategooriatega hierarhia tipus. Klaster suurusega 1000 annab suuremaid ülekatteid kui 0,1 kõigi nimetatud termini eellastega ning molekulaarse funktsiooni (GO:0003674) ontoloogiat läbides jõuab juureni. Ühisosi leitakse ka bioloogilise protsessi ontoloogiaga, suuremaid ülekatteid kui 0,1 annavad selle 2 GO kategooriat.

Huvitav oleks teada, kus need 2 kõige selgemat trajektoori näidanud GO kategooriat väiksemate klastrite korral asuvad ehk mis on nende suhtelised ülekatted nö. heade klastrisuuruste 26-105 korral. Selgub, et klastri suurusega 26 korral on ülekate 0,029 ja 105 korral 0,079. See tähendab, et selle GO kategooria ja järjest suureneva klastri ülekate on stabiilselt kasvav. Täpselt

samamoodi käitub ka tema eellane GO:0005737 vaatamata väiksematele ülekatetele.

Vaatame ka neid GO kategooriad, mis kõige suuremaid suhtelisi ülekatteid annavad. Need jäävad nn. hea suurusega klatrite hulka. Suurima ülekatte andis ontoloogiapuus sügavusel 8 olev GO:0017062 - *Process: cytochrome bc(1) complex biogenesis*. Ülekatte suuruse poolest järgnesid GO:0005746 (*Component: mitochondrial electron transport chain*), GO:0006119 (*Process: oxidative phosphorylation*) ja GO:0004129 (*Function: cytochrome-c oxidase activity*) oma kahe eellasega. Et sarnaseid ülekatteid hierarhiliselt lähestikku paiknevad GO kategooriad annavad, kinnitavad ka GO:0006119 kahe järglase (otsese ja selle järglase) leitud ülekatted, mis oma suuruselt pingeritta mahuvad.

Vaatleme lisaks GO kategooriaid klastrisuuruste 200-600 korral, kus on näha selgelt domineerivaid termineid. Neist suurimat ülekatted andva GO termini tuvastasin juba eelpool (GO:0005739), suuruselt järgmine on selle järglane GO:0005740 (*Component: mitochondrial membrane*, hierarhias sügavusel 5). Suure ülekatte poolest järgneb GO:0015980 oma eellasega GO:0006091, mõlemad bioloogilise protsessi ontoloogiast, hierarhias sügavusel vastavalt 6 ja 5. Rakulist paiknemist kirjeldab selles klastrisuuruste vahemikus GO:0019866 (*Component: inner membrane*), hierarhiliselt sügavusel 5.

Klastri suuruse määramine on esimene samm meetodite 1 ja 2 kiiremaks muutmisel.

4.3.4.2 Künnis

Minimaalse ühisosa ehk künnise seadmine on teine võimalus meetodi 2 töökiiruse parandamiseks. Arvutuste käigus selgus, et ainult ühe pärmigeeniga seotud termineid on ligi 1/3 kõikidest pärmi geenidega assotsieeritud terminitest. See aga vähendaks oluliselt ühishulkade otsimisse kaasatud assotsieeritud terminite arvu ja innustab otsima suuremat künnist.

Rahuldava minimaalse ühisosa leidmiseks kasutan ühisosa leidumise tõenäosust ja hüpergeomeetrilist jaotust, mis eeldab, et andmed on lõplikud ja pärast igat valikut (geeni sattumist klastrisse) võimalused järgmiste geenide valikuks muutuvad [VK01]. Leides kõigepealt klastri ja terminiga seotud geenide ühisosad on võimalik arvutada tõenäosus sellise või suurema ühisosa saamiseks. Nimetatud tõenäosus leitakse järgnevalt:

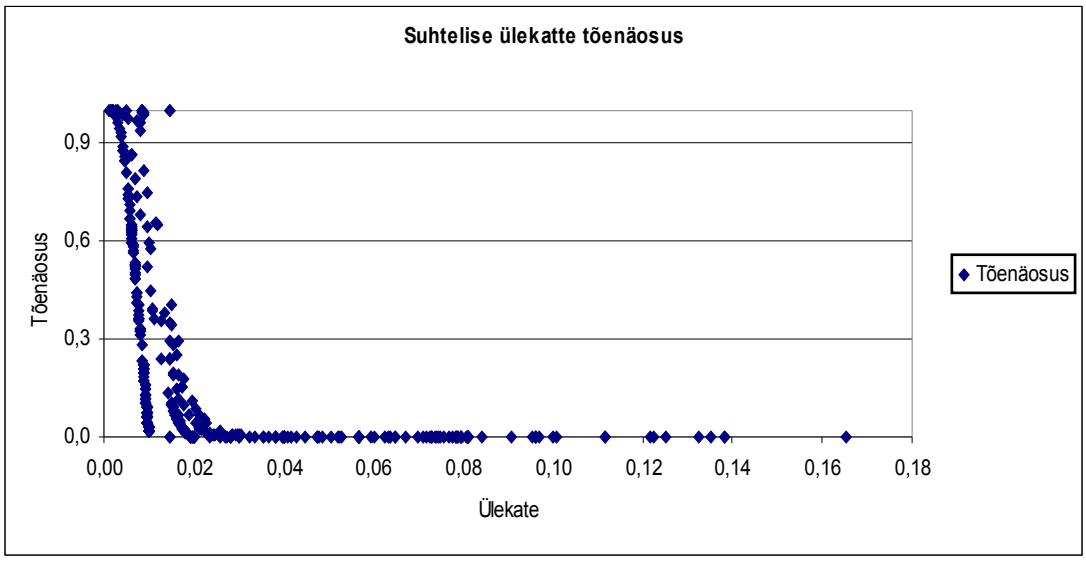
$$p_k = \frac{C_K^k \cdot C_{N-K}^{n-k}}{C_N^n} ,$$

kus n on klastri suurus, k ühisosa suurus, N kogu geenide arv, K terminiga seotud geenide arv. Mida väiksem see tõenäosus tuleb, seda tähenduslikumaks leitud ühisosa ja vastav GO termin geeniklastri jaoks on.

Künnise leidmise eksperimenti võtsin klastri suurusega 100, mis on niiöelda hea suurusega klaster. Kuna klastriread failis *klastrid.txt* on genereeritud reaalsete andmete põhjal, iga rida samal põhimõttel, siis sobib sealt katse jaoks võtta mistahes klaster. Võtsin selleks kõige esimese. Valitud klastrit võrdlesin kõikide terminitega seotud geenihulkadega ning leidsin ühisosad ja vastavad ülekatted. Hulkade A ja B ülekatteks $C_{A,B}$ nimetatakse suurust

$$C_{A,B} = \frac{|A \text{ and } B|}{|A \text{ or } B|} = \frac{|A \text{ and } B|}{|A| + |B| - |A \text{ and } B|} .$$

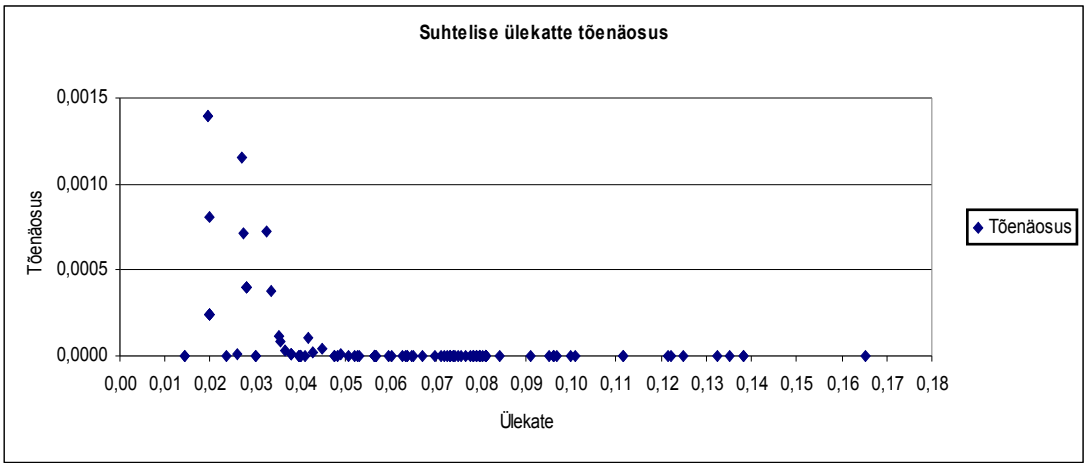
Lisaks leiti GO terminiga seotud geenide arv ja kogu geenide arv. Nende andmete põhjal arvutas teek *statistics* vastavate ühisosade esinemise tõenäosused.



Joonis 9. Ülekatted ja nende leidumise tõenäosus klastrisuuruse 100 korral.

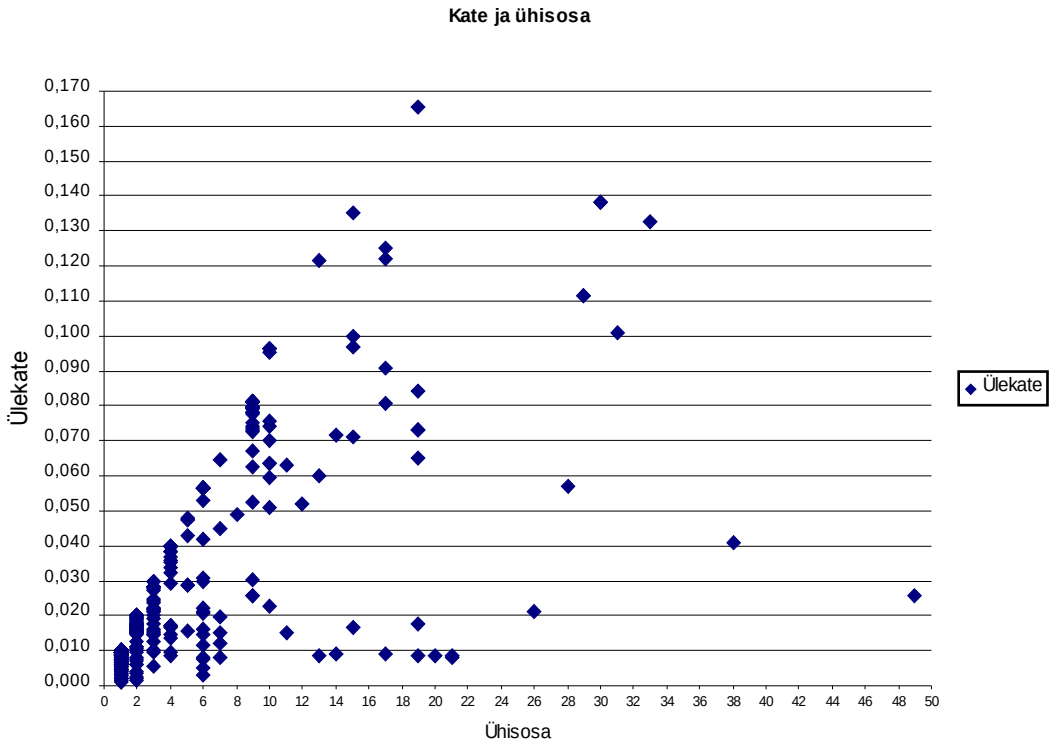
Selgub, et ühtlaselt võrdse tõenäosusega leidub suhtelisi ülekatteid suurusega 0,005 kuni 0,012. Võiks arvata, et ülekaalus on suure leidmistõenäosuse ja väikese ülekattega GO terminite ja otsinguklastri paarid. Ometi leitakse palju just paare keskmise suurusega ülekatetega, mille leidumine on vähetõenäoline. Ülekatteid, mille konkreetsetel juhtudel leidumise tõenäosus oli vähem kui 0,15, leiti kokku 307 juhul 460-st. Neist omakorda 131-e leidumistõenäosus oli alla sajandiku ja 114-ne tõenäosus alla tuhandiku.

Sobiva tõenäosuse valik on subjektiivne. Mina valisin rahuldavaks tõenäosuseks 0,001 ja sellest väiksemad suurused. Vastavalt sellele saab heaks katteks nimetada ülekatet suurusega 0,033 ja sellest suuremaid ülekatteid (joonis 10).



Joonis 10. Suhtelised ülekatted ja nende leidumistõenäosus klastrite suurusega 100 korral.

Künnise valimiseks viisin kokku vastavad ülekatted ja ühisosad (joonis 11).



Joonis 11. Ülekatetele vastavad ühisosad üle GO terminite klastrisuuruse 100 korral.

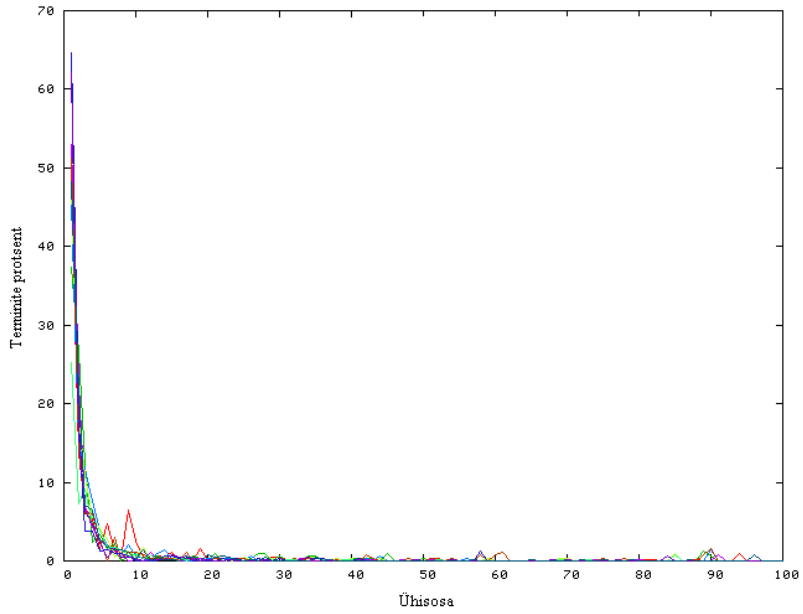
Selgub, et leitud heale ülekattele vastavad ühisosad, mis on võrdsed ja suuremad neljast. Samas leidub häid ülekatteid kogu ühisosa suuruse vahemikus 4-st 20-ni, mis viitab kohasele künnise valikule just nende seast.
Eelpoolseid järeldusi künnise osas kinnitas samasugune katse veel ühest geenireast moodustunud klastritega.

Võttes aluseks joonise 5 võib arvata, et alates klastritest suurusega 200 on mõistlik võtta uus, suurem künnis. Läbiviidud analoogne eksperiment 200-se suurusega klastri korral andis heaks katteks 0,015 ja sellest suuremad väärtused ning sellele vastavalt minimaalseks ühisosaks 4.

4.3.5 Perli meetodite võimalused ja kokkuvõte

4.3.5.1 Ühisosade jaotumine

Huvitav oleks teada, kuidas jagunevad leitud ühisosad GO kategooriate vahel. Selle eksperimendi tarvis valisin 10 klastrit suurusega 100. Kõiki neid võrreldi GO terminitega ning tulemus on järgmisel joonisel.



Joonis 12. 10 päringuga leitud ühisosad ja neile vastav GO terminite protsent päringuga ühisosa andnud terminitest.

Nagu eeldadagi võis, tekib rohkem väikseid ja vähem suuremaid ühisosi. Huvitavaid tõuse teeb graafik jällegi päris suurte ühisosad puhul. Siin on roll GO kategooriate hierarhilisusel – juurelähedased tipud on seotud kõigi oma järglaste geenidega.

4.3.5.2 Klastri kirjeldamine

Programmi põhieesmärgi – geeniklastrit hästi kirjeldavate GO kategooriate leidmine – illustreerimiseks tegin 2 katset: ühe ja kümne 100-elemendilise klastriga.

1 päring üle GO kategooriate

Tulemus sisaldab termineid, mille suhtelised ülekatted olid võrdsed või suuremad 0,033-st. Tabel 4 sisaldab 10 kõige suurema ülekattega terminit koos vastava ühisosa suurusega.

GO:0006119	ülekate	0.16522	ühisosa	19	oxidative phosphorylation
GO:0005743	ülekate	0.13825	ühisosa	30	mitochondrial inner membrane
GO:0019866	ülekate	0.13825	ühisosa	30	inner membrane
GO:0005746	ülekate	0.13514	ühisosa	15	mitochondrial electron transport chain
GO:0005740	ülekate	0.13253	ühisosa	33	mitochondrial membrane
GO:0015078	ülekate	0.12500	ühisosa	17	hydrogen ion transporter activity
GO:0015077	ülekate	0.12230	ühisosa	17	monovalent inorganic cation transporter activity
GO:0006118	ülekate	0.12150	ühisosa	13	electron transport
GO:0015980	ülekate	0.11154	ühisosa	29	energy derivation by oxidation of organic compounds
GO:0006091	ülekate	0.11154	ühisosa	29	energy pathways

Tabel 4. Suurima suhtelise ülekattega GO kategooriad ühe 100-elemendilise klastriga.

Samavärvilised read tähistavad ühte ontoloogiat. Lisaks moodustavad kõik need ühe ontoloogia terminid konkreetse alampuu. Ei kehti seaduspära, et hierarhias kõrgemal asuvad GO terminid annavad suuremaid ülekatteid. Samas ei saa väita ka vastupidist.

10 päringut üle GO kategooriate

Tulemus sisaldab kõikide terminite suurimat ülekatet kümne päringu puhul ja näitab ära ka päringu esimese geeni numbrilisel, nimeks teisendamata kujul. Tabelis 5 toon ära 10 kõige suurema leitud ülekattega terminit. Päringu number tähendab siinkohal klastri esimesele geenile vastavat numbrit, mis seoses faili *nimed.txt* struktuuriga algab arvust 0.

GO:0007046	ülekate	0.21250	päringul	5	ühisosa	51	ribosome biogenesis
GO:0005730	ülekate	0.19868	päringul	5	ühisosa	60	nucleolus
GO:0042254	ülekate	0.19565	päringul	5	ühisosa	54	ribosome biogenesis and assembly
GO:0006364	ülekate	0.19178	päringul	5	ühisosa	42	rRNA processing
GO:0030154	ülekate	0.16561	päringul	2	ühisosa	26	cell differentiation
GO:0030435	ülekate	0.16561	päringul	2	ühisosa	26	sporulation
GO:0006119	ülekate	0.16522	päringul	0	ühisosa	19	oxidative phosphorylation
GO:0030437	ülekate	0.15232	päringul	2	ühisosa	23	sporulation (sensu Fungi)
GO:0016072	ülekate	0.14634	päringul	5	ühisosa	42	rRNA metabolism
GO:0007151	ülekate	0.14286	päringul	2	ühisosa	21	sporulation (sensu Saccharomyces)

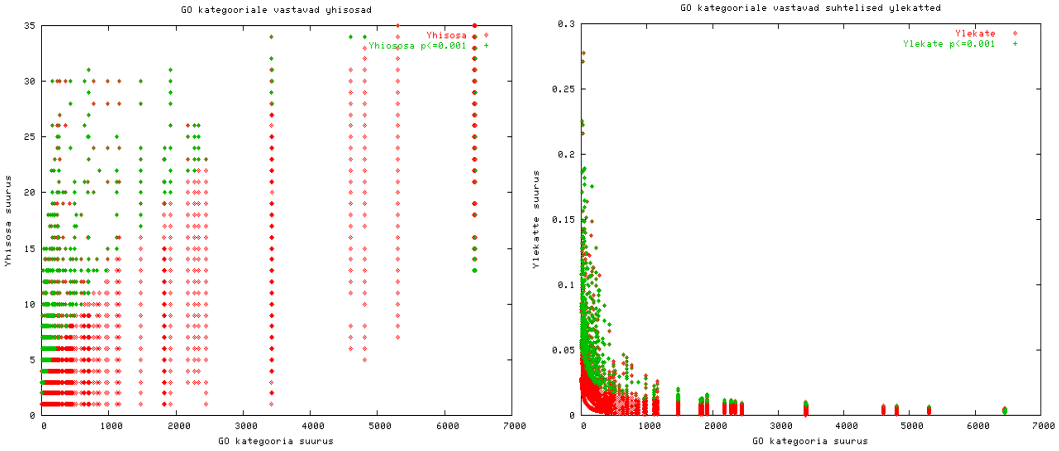
Tabel 5. Suurima suhtelise ülekattega GO kategooriad kümne 100-elemendilise klastriga.

Esiti on näha kohe, kuidas kuues (päringu algusgeen numbrina 5) päring kõige rohkem suuri ülekatteid annab. Lisaks moodustub selgelt kolm alampuud, vastavalt esindatud päringute arvule. Neist 2 asuvad bioloogilise protsessi ontoloogias.

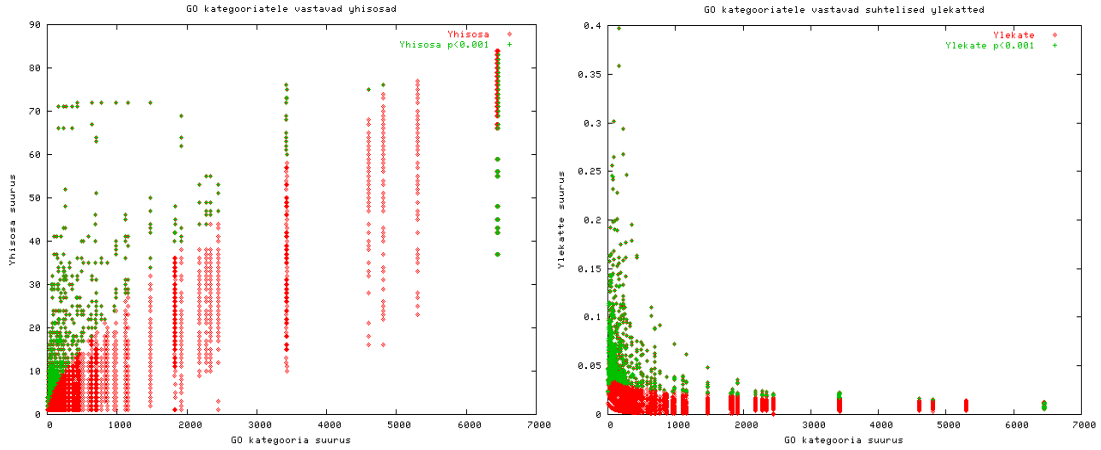
Omamoodi klastrite kirjeldus suurema üldpildi põhjal jäi ka punkti 4.3.4.1 (joonis 8).

4.3.5.3 GO kategooriad ja ülekatted

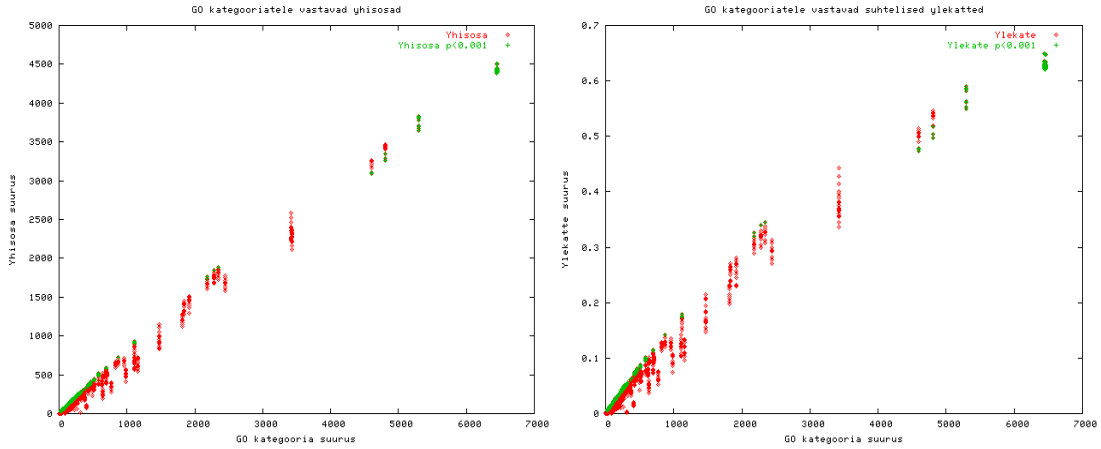
Vaatame klastrisuurusi 35 ja 84, mis on summaarse suhtelise ülekatte lokaalsed optimumid (joonis 6) ning alternatiivina klastrit suurusega 5000. Leian kõigi kolme kohta GO kategooriate ühisosad, suhtelised ülekatted ja vastavad GO kategooriate suurused. Ühisosade leidumise tõenäosuse piirmääraks jätan ühe tuhandiku. Joonised on moodustatud 32000 nullist suurema ühisosaga kategooria ja klastri andmete põhjal.



Joonis 13. Klastrid suurusega 35 ning vastavalt nende nullist suuremad ühisosad ja vastavad ülekatted. Rohelise värviga ühisosad ja ülekatted, mille leidumistõenäosus on väiksem kui üks tuhandik.



Joonis 14. Klastrid suurusega 84 ning vastavalt nende nullist suuremad ühisosad ja vastavad ülekatted. Rohelise värviga ühisosad ja ülekatted, mille leidumistõenäosus on väiksem kui üks tuhandik.



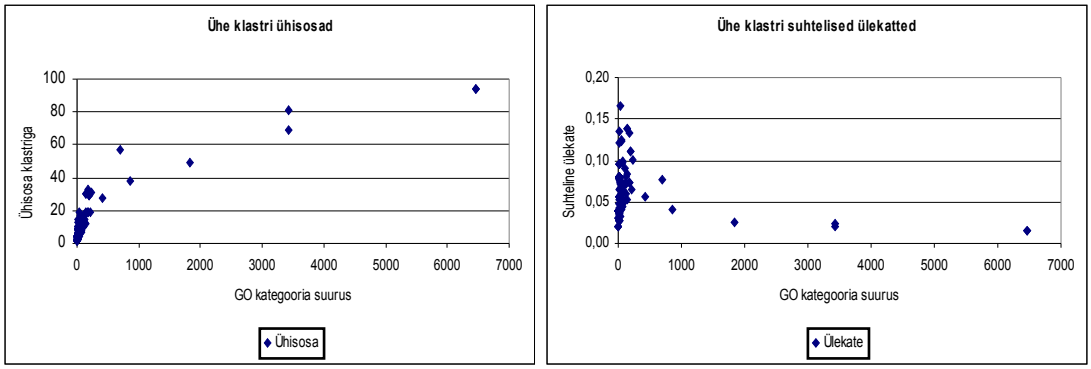
Joonis 15. Klastrid suurusega 5000 ning vastavalt nende nullist suuremad ühisosad ja vastavad ülekatted. Rohelise värviga ühisosad ja ülekatted, mille leidumistõenäosus on väiksem kui üks tuhandik.

Ühisosad joonistel 13, 14 ja 15 kasvavad lineaarselt ja koonduvad kujuteldava telje ümber. Sellise telje põhjustab GO kategooriate suuruste jaotus (joonis 3). Eriti hästi ilmneb see joonisel 15 klastrite suurusega 5000 korral – GO kategooriate suurusega 3420 ja 3430 (leitud seoses joonis 3 genereerimisega) ühisosad koonduvad suurusvahemikku 2000-2500. Seda seletab sellise suurusega GO kategooriate arv, mis on kummalgi juhul 1 (nimetatud suuruspiirkonnas keskmiselt 1,4). Seega päringu suurusega 5000 (see hõlmab endas ligi 80% kõigist pärmi geenidest) ja vaadeldava suurusega kategooriate antavad ühisosad ei saagi tulla väiksemad kui 2199: kategooria suurus miinus

maksimaalne mittetabamine, mis tuleb vastavalt 3420-(6221-5000)=2199 ja 3430-(6221-5000)=2209.

Piir leidumistõenäosusega 0,001 ja suurema tõenäosusega ühisosade vahel kulgeb joonistel 13, 14 ja 15 analoogselt. Ilmne on seegi, et klastri suuruse kasvades kasvavad ühisosad ning koos sellega ühisosade leidumistõenäosus suureneb. Näiteks ühisosa 10 leidmine 500-se GO kategooria puhul on klastrite suurusega 35 korral juba vähetõenäoline, samas kui suurusega 84 klastrite korral on selle tõenäosus suurem kui 0,001. Klastrite suurusega 5000 korral nii väikseid ühisosi ei leitud selles katses üldse, kuigi teoreetiliselt võivad nad olemas olla. Suurte GO kategooriate korral tekib kirjeldatule just vastupidine efekt – vähetõenäolisteks osutuvad väiksed ühisosad. Siin ei saa enam kasutada sama mõõdupuud hea ülekatte leidmiseks, sest nende GO kategooriatega on seotud kõik pärmi geenid. Seega ei oma selles punktis ülekate ja ühisosa tegelikult enam tähtsust.

Kui teha samasugune katse juba eelnevalt vaatluse all olnud konkreetse 100-elemendilise klastriga, saan graafikud (joonis 16), mis järgivad siinseid üldiseid malle.

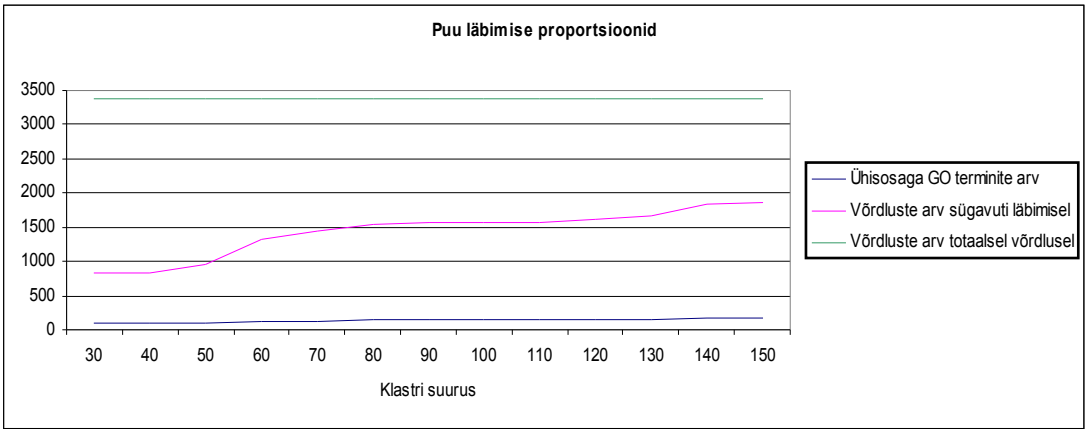


Joonis 16. Konkreetse 100-elemendilise klastri ühisosad ja suhtelised ülekatted leidumistõenäosusega $p<0,001$.

Leitud üldiste mallide järgimist kinnitab ka punktis 4.3.4.2 toodud klastriga suurusega 200 läbi viidud katse tulemuste kõrvutamine joonistega 13 ja 14 – klastri suurenedes hea ülekatte suurus väheneb ja minimaalne ühisosa suureneb.

4.3.5.4 Meetodite erinevusi

Totaalse otsingu korral käiakse läbi kõik 3389 tippu – tipud, mis on assotsieeritud geenidega failist *nimed.txt*. Künnise seadmisel sügavuti läbimisele see arv ilmselgelt väheneb. Sügavuti läbimisega (hierarhiline lähenemine) puus läbi käidavat osa selgitab joonis 17.



Joonis 17. Totaalse võrdlusega ja hierarhilise lähenemisega (künnisega 4) läbi käidav GO kategooriate arv ühe suvalise klastrirea kasvamisel.

Konkreetselt läbi viidavate hulgavõrdluste arv on oluliselt suurem leitud ühisosadega terminite arvust. Hüppeliselt kasvab võrreldavate hulkade arv 50 ja 60 suurusega klastri vahel (21,2 korda), samas kui leitud ühisosaga GO kategooriate arv kasvab vaid 2 korda. Analoogne, kuigi vähem drastiline situatsioon on klastrisuuuste 130 ja 140 vahel. Seega neis vahemikes klastrisse lisandunud geenid suutsid panustada vajalikku minimaalsesse ühisosasse ning seeläbi tuua võrdlusse hulga uusi järglasi, kuid mitte oluliselt suurendada vajaliku künnise ületanud GO kategooriate arvu. Selles klastrisuuruste vahemikus ilmneb ka seos, et vastava arvu GO terminite leidmiseks võrreldakse 10,3 korda rohkem hulki ehk umbes iga kümnes võrdlus ületab künnise 4.
Kõigi testandmeteks olevate 6221 klastri läbi vaatamisel tehakse järgmine arv hulkade võrdlusi (tabel 6).

	Klastrid suurusega 30	Klastrid suurusega 60	Klastrid suurusega 100	Klastrid suuruseg a 130	Klastrid suurusega 150
Totaalne võrdlus	21064306	21064306	21064306	21064306	21064306
Sügavuti läbimine künnisega 4	3220613	5185110	7487389	9008952	9939486

Tabel 6. Kahe meetodi hulgavõrdluste arv 6221 klastri korral.

4.3.6 Päring MySQL andmebaasile

Alternatiivina Perli andmestruktuuridele vaatleme andmete hoidmist MySQL andmebaasis ning sealt päringute koostamisele. Mind huvitava ühisosa leidmiseks on tarvilikud kaks punktis 4.1.2 kirjeldatud tabelit: tabel TERMIN ja tabel ASSOTS. Kuna päringuhulgana kasutasin faili *klastrid.txt*, tuli sealsetele numbriloenditele genereerida vastavad geeniloendid *nimed.txt* järgi. Vajalike andmete kättesaamiseks katsetasin kahte viisi:

Päring 1.

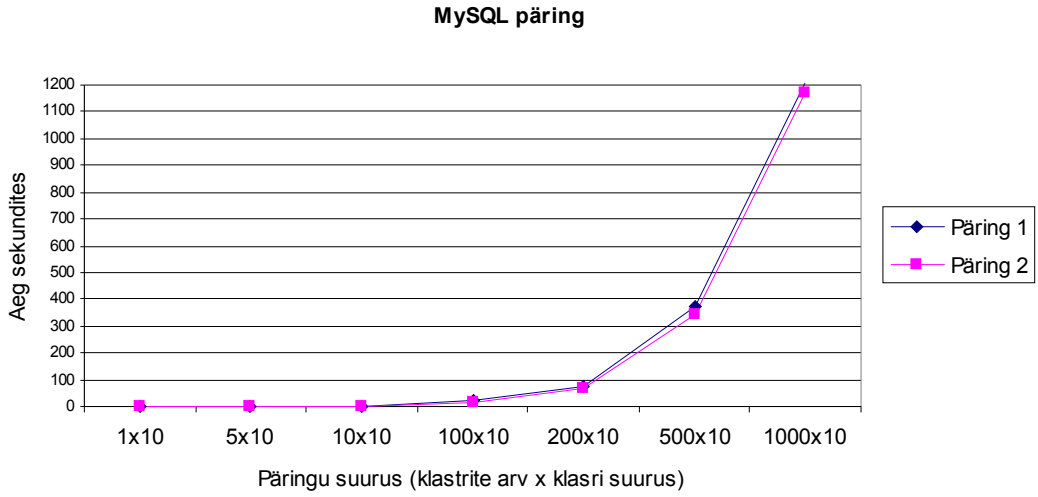
```
SELECT DISTINCT(term)
FROM assots
WHERE geen=$geen;
while (($term) = $sth->fetchrow){

...

SELECT DISTINCT(t_vanem)

FROM termin

WHERE term=$term;

...
}
```

Päring 2.

```
SELECT assots.term as term,termin.t_vanem as t_vanem
```

```
                    FROM assots, termin
WHERE assots.geen=$geen and assots.term=termin.term;
```

Testimise tulemusena kujunes järgnev graafik.

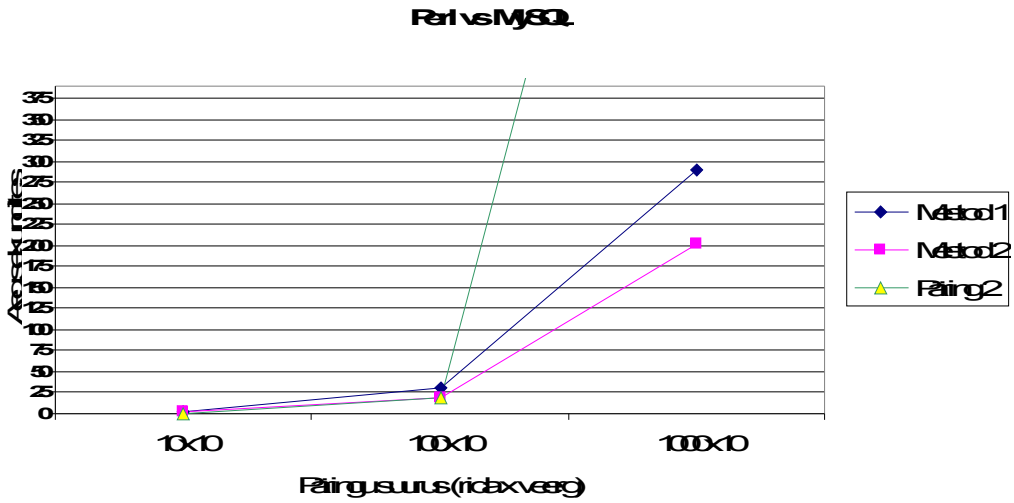


Joonis 18. Kahe MySQL päringu töökiiruse ajaline võrdlus klastrisuurus 10 korral.

Jooniselt 18 on näha, et päringute töökiiruses märkimisväärset erinevust ei ole, kuid stabiilselt on efektiivsem päring 2.

4.3.7 Perl vs andmebaas

Erinevate meetodite testimise tulemusena võib teha järelduse, et antud juhul on efektiivsem kasutada Perli andmestruktuure kui hoida andmeid MySQL andmebaasis. Ka kiiremaks osutunud MySQL päringu kiirus jääb alla mõlemale Perli meetodile, mida illustreerib joonis 19. Kuna andmebaasipäring mahu kasvades ilmselgelt ebaefektiivseks osutub, ei ole testimiseks suuri hulki kaasatudki.



Joonis 19. Perli meetodite (totaalne võrdlus ja hierarhiline läbimine) ja MySQL kiirema päringu töökiiruste võrdlus.

Samal ajal ilmnes aga ka fakt, et väiksemate päringuhulkade korral töötab MySQL päring Perli programmidest kiiremini. Antud juhul (joonis 19) kaotab andmebaasipäring oma kiiruses alates sajast päringust suurusega 10. Seega kui mõelda kasutajale, keda huvitab vaid konkreetne geeniklaster, oleks siiski efektiivsem andmete hoidmine andmebaasi kujul. Antud töös on eesmärgiks üldisem pilt, mis saadakse võimalikult paljude andmete analüüsil.

5. Kokkuvõte

Geneetika on pidevalt arenev ja seeläbi täienev teadusharu. Sellele on omased suured andmehulgad, mille kasutamise efektiivsemaks muutmisega tegeleb ka bioinformaatika. Siin on jõutud äratundmisele, et eri liikide geenid ja geeniproduktid on kirjeldatavad ühtse terminoloogiaga. Selliste mitte liigispetsiifiliste terminite väljatöötamise ja omavahel seostamisega tegeleb *The Gene Ontology Consortium,* projekt aastast 1998. Kõigi geeniproduktide kirjeldamiseks on loodud kolm üksteisest sõltumatut ontoloogiat. Kõik geeniproduktid seotakse nende ontoloogiate kategooriatega, mida vajadusel juurde lisatakse, korrigeeritakse või kehtetuks tunnistatakse. Fakt aga on, et kõiki geeniprodukte ei suudeta kohe kategoriseerida. Et geeniprodukti kirjeldada ja vastava GO terminiga annoteerida, võetakse appi geenide väljendumismuster, mis õigete paralleelide tõmbamisel tõstab hüpoteesi sarnaste ekspressiooniprofiilidega geenide funktsionaalse sarnasuse kohta. Siin tuleb bioinformaatika appi erinevate klasterdamistehnikatega ning annab seeläbi ainet geeniklastrite kaudu geenide kirjeldamisele.

Käesoleva töö teoreetilises osas on selgitatud geeniontoloogia olemust, selle arendamist, haldamist ja säilitamist. Lisaks annotatsioonifailide kirjeldused ning lõpuks ontoloogiaid ja annotatsioone koos kasutavate tööriistade ülevaade.

Praktiline osa on püüe arendada kiire ja efektiivne viis geeniklastrite kõige paremini kirjeldavate GO kategooriate leidmiseks, et seeläbi klastrisse kuuluvate tundmatute geenide funktsionaalsust ennustada ja määrata. Selliste andmehulkade puhul osutusid MySQL andmebaasist ja päringutest efektiivsemaks Perli andmestruktuurid. Perli puhul oli lahendusele omakorda mitu lähenemisvõimalust – absoluutne hulkade võrdlemine ja hierarhiline lähenemine. Seoses geeniontoloogia säilitamisel puu-struktuurina osutus samu asju kiiremini leidvaks programm, mis seda hierarhiat kasutas. Nüüd juba reaalseid testandmeid kasutades oli võimalik leida vastuseid küsimustele kui pikk on optimaalne päringuklaster, kui suur on olulisust näitav GO kategooriaga seotud geenide ja päringugeenide suhteline ülekate ja milline võiks seeläbi olla minimaalne aktsepteeritav ühisosa graafi sügavuti läbimisel; kui suure osa ontoloogiast peab läbi käima hierarhilisel lähenemisel ja kuidas seda mõjutab päringuklastri kasvatamine; kas suhtelised ülekatted on juhuslikud suurused või koonduvad need kindla klastrisuuruse korral. Lisaks veel näiteid klastrile hea kirjelduse leida püüdmise ja ühisosade leidmisel GO kategooriate jaotumise kohta.

Loodud programmid on lisaks testandmeteks olnud toidupärmi geenidele rakendatav ka teiste liikide geenidele. Edasine eesmärk oleks tavakasutaja jaoks veebipõhise liidese loomine, mille kaudu leiaks graafilise väljundi ka päringute põhjal genereeritavad joonised.

6. Using gene ontology to efficiently find important relations

Jaanika Luik

Abstract

Genetics is constantly in develop and thereby supplemented discipline. Common is huge amount of genetic data. To promote and improve research on that, bioinformatics deals with different research techniques. One important step is developing gene ontologies by The Gene Ontology Consortium. Those ontologies arrange and hold not species-specific terms in three independent ontologies. Discovered genes and gene products are associated to them, so that every known gene can later be classified. All genes, of course, can not be grouped like this, because their function is unknown. However, bioinformatics tries to do this by using gene expressions. Consistently coexpressed genes under several different conditions may share something common in their regulatory mechanisms and have similarities in their function. Clustering these genes and describing those sets by GO terms may identify functions of unknown genes in specific cluster.

In the theoretical part of this work is explained the character of gene ontology, its development, managing and storing. Further descriptions of annotation files and eventually abstract of tools that use ontologies and annotations.

In the practical part I try to develop a fast and effective way to describe gene clusters by GO terms and thereby to predict and define unknown genes in the cluster. With this kind amount of data programming language Perl compared to MySQL was more efficient. Using Perl there was two ways to solve the raised problem - total search (comparing all sets) and hierarchical approach. Since the structure of gene ontology is a direct acyclic graph, hierarchical approach showed higher speed when comparing sets. Now, using real data for tests, there was possible to answer questions like how long is the optimal gene cluster in query, what is the overlap size of genes associated with GO terms and query genes and what is the appropriate minimal size of intersection between them when using hierarchical approach; how big is the proportion of ontology that must be traversed using hierarchical approach and how it is influenced by enlarging the cluster in query; are overlaps occasional or do they aggregate on certain cluster size. Plus examples of finding good description and how GO categories divide when intersects exists.

Created programs can be used of course with annotation files other species but yeast. Further goal would be building a web based GUI (Graphical User Interface), through what also generated graphs are displayed.

7. Viited

7.1 Kasutatud kirjandus

[ADD04] Fátima Al-Shahrour, Ramón Díaz-Uriarte, Joaquín Dopazo
FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes Bioinformatics Advance Access published on March 1, 2004. Bioinformatics 2004 20: 578-580; DOI: 10.1093/bioinformatics/btg455

[AM01] T.K. Attwood, C.J Miller
Which craft is best in bioinformatics? *Computers and Chemistry*, 2001, 25, 329-339

[BAHI01] Matt Berriman, Martin Aslett, Neil Hall, Al Ivens
Parasites are GO. *TRENDS in Parasitology*, Vol.17 No.10 October 2001

[BSGG+04] Michael Bada, Robert Stevens, Carole Goble, Yolanda Gil, Michael Ashburner, Judith A. Blake, J. Michael Cherry, Midori Harris, Suzanna Lewis
A short study on the success of the Gene Ontology. *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol.1, Issue 2, Feb 2004, 235-240

[CMBB+03] Camon E, Magrane M, Barrell D, Binns D, Fleischmann W, Kersey P, Mulder N, Oinn T, Maslen J, Cox A, Apweiler R.
The Gene Ontology Annotation (GOA) Project: Implementation of GO in SWISS-PROT, TrEMBL, and InerPro. *Genome Research*, Vol.13, Issue 4, April 2003, 662-672

[DB03] Gerard Drewes, Tewis Bouwmeester
Global approaches to protein-protein interactions. *Current Opinion in Cell Biology*, Vol.15, Issue 2, April 2003, 199-205

[DDBM03] Vincent Detours, Jacques E. Dumont, Hugues Bersini, Carine Maenhaut
Integration and cross-validation of high-throughput gene expression data: comparing heterogeneous data sets. *FEBS letters*, Vol.546, Issue 1, 3 July 2003, 98-102

[DKMO+03] S. Drâghici, P. Khatri, R.P. Martins, G.C. Ostermeier, S.A. Krawetz
Global functional profiling of gene expression. *Genomics*, 2003, 81, 98-104

[HGL03] Steffen Hennig, Detlef Groth, Hans Lehrach
Automated Gene Ontology annotation for anonymous sequence data. *Nucleic Acids Research*, 2003, Vol.31, No. 13

[LMG03] Ning Lan, Gaetano T Montelione, Mark Gerstein
Ontologies for proteomics: towards a systematic definition of structure and function that scales to the genome level. *Current Opinion in Chemical Biology*, Vol.7, Issue 1, Feb 2003, 44-54

[TGOC01] The Gene Ontology Consortium
Creating the Gene Ontology Resource: Desing and Implementation. *Genome Research*, Vol. 11, Issue 8, August 2001, 1425-1433

[VBJR+00] J. Vilo, A. Brazma. I. Jonassen, A. Robinson, E. Ukkonen
Mining for putative regulatory elements in the yeast genome using gene expression data*. Proc. of Eighth International Conference on Intelligent Systems for Molecular Biology,* Vol. 8, 2000, 384-394

[VK01] J. Vilo, K. Kivinen
Regulatory sequence analysis: application to the interpretation of gene expression. *European Neuropsychopharmacology*, Vol. 11, 2001, 399-411

[VKKS+03] Jaak Vilo, Misha Kapushesky, Patrick Kemmeren, Ugis Sarkans, Alvis Brazma.
Expression Profiler.
In Parmigiani,G., Garrett,E.S., Irizarry,R. and Zeger,S.L. (eds), The Analysis of Gene Expression Data: Methods and Software, Springer Verlag, 2003, New York, NY.

7.2 URL-id

[URL:AmiGO] AmiGO!
http://www.godatabase.org/cgi-bin/amigo/go.cgi, 18.05.2004

[URL:DAG] Project:GeneOntology:FileList
http://sourceforge.net/project/showfiles.php?group_id=36855, 18.05.2004

[URL:EASE] EASE: the Expression Analysis Systematic Explorer
http://david.niaid.nih.gov/david/ease.htm, 15.03.2004

[UR:EBC] Mari Kelve

EBC: Molekulaarbioloogia sõnaraamat
http://www.tymri.ut.ee/sonastik.html, 25.05.2004

[URL:eGOn] Tool for Genomic Data linked to Gene Ontology (GO): Annotation, Mapping to GO-Structure and Statistical Tests
http://nova2.idi.ntnu.no/egon/, 18.05.2004

[URL:EPGO] Browser and analysis for Gene Ontology
http://ep.ebi.ac.uk/EP/GO/, 18.05.2004

[URL:FatiGO] FatiGO: Data mining with Gene Ontology
http://fatigo.bioinfo.cnio.es/, 15.03.2004

[URL:FunSpec] FunSpec (an acronym for "Functional Specification")
http://funspec.med.utoronto.ca/, 15.03.2004

[URL:GARBAN] Genomic Analysis for Rapid Biological Annotation
http://garban.tecnun.es/garban/home.php, 07.03.2004

[URL:GeneMerge] GeneMerge
http://www.oeb.harvard.edu/hartl/lab/publications/GeneMerge/GeneMerge.html, 15.03.2004

[URL:GenNav] GenNav
http://etbsun2.nlm.nih.gov:8000/perl/gennav.pl, 18.05.2004

[URL:GOannot] GO Annotation Guide
http://www.geneontology.org/GO.annotation.html#file, 23.03.2004

[URL:GOblet] Detlef Groth, *Steffen Hennig*
GOblet service at *MPI for Molecular Genetics*
http://goblet.molgen.mpg.de/, 07.03.2004

[URL:GOformat] File Format Guide
http://www.geneontology.org/GO.format.html#goflat, 23.03.2004

[URL:GoMiner] GoMiner
http://discover.nci.nih.gov/gominer/index.jsp, 15.03.2004

[URL:GOslim] GO Slims
ftp://ftp.geneontology.org/pub/go/GO_slims/, 23.03.2004

[URL:GoSurfer] GoSurfer
http://biosun1.harvard.edu/complab/gosurfer/, 15.03.2004

[URL:MAPPFinder] MicroArray Pathway Profiler
http://www.genmapp.org/MAPPFinder.html, 15.03.2004

[URL:MatchMiner] MatchMiner

http://discover.nci.nih.gov/matchminer/html/index.jsp, 15.03.2004

[URL:MGI] MGI GO Browser http://www.informatics.jax.org/searches/GO_form.shtml, 18.05.2004

[URL:MySQL] GO Database Schema http://www.godatabase.org/dev/sql/doc/godb-sql-doc.html, 20.03.2004

[URL:Onto] Intelligent Systems and Bioinformatics Laboratory http://vortex.cs.wayne.edu/projects.htm, 18.05.2004

[URL:Ontologizer] P. N. Robinson, A. Wollstein, U. Boehme, B. Beattie Ontologizer http://www.charite.de/ch/medgen/ontologizer/, 07.03.2004

[URL:OntologyTraverser] Chad Shaw, Andrew Young, Nathan Whitehouse OntologyTraverser http://franklin.imgen.bcm.tmc.edu/rho/services/index.jsp?page=OntologyTraverser, 20.03.2004

[URL:QuickGO] QuickGO GO Browser http://www.ebi.ac.uk/ego/, 18.05.2004

[URL:TAIR] TAIR Keyword Search and Browse http://www.arabidopsis.org/servlets/Search?action=new_search&type=keyword, 18.05.2004

[URL:TermFinder] SGD Gene Ontology Term Finder http://genome-www4.stanford.edu/cgi-bin/SGD/GO/goTermFinder, 18.05.2004

[URL:TermMapper] SGD Gene Ontology Term Mapper http://genome-www4.stanford.edu/cgi-bin/SGD/GO/goTermMapper, 18.05.2004

8. Lisad

8.1 Lisa 1. CD sisu

1) abaas.pl – andmete lugemine annotatsioonifailist ja sisestamine MySQL tabelisse ASSOTS,
2) ab_yhisosa_1paring.pl – ühisosa leidmine MySQL andmebaasist ühe päringuga (päring 2),
3) ab_yhisosa_2paringut.pl - ühisosa leidmine MySQL andmebaasist kahe päringuga (päring 1),
4) all_assots.pl – leiab kõik GO terminitega seotud geenid, loeb need kokku; kitsendatud variant *k6ik_geenid.pl* –st,
5) all_assots.storable – *hash*i ALL_ASSOTS hoidmiseks teegi *lib.storable* abil,
6) assots.storable – *hash*i ASSOTS hoidmiseks teegi *lib.sotrable* abil,
7) geenid.pl – loeb ontoloogia- ja annotatsioonifailid, leiab iga GO terminiga otseselt seotud geenid ja salvestab need *assots.storable*-s *hash*ina,
8) go.storable – *hash*i GO hoidmiseks teegi *lib.sotrable* abil,
9) goparent.pl – loeb ontoloogiafaili, leiab iga GO termini kõik vanemad ja nende kaugused sellest terminist; täidab MySQL tabeli TERMIN,
10) k6ik_geenid.pl – leiab kõik iga GO terminiga seotud geenid ja salvestab need *all_assots.storable*-s *hash*ina; lisaks terminiga seotud geenide arv ja terminite arv kokku,
11) klastrid.txt – numbriline tekstifail, mis sisaldab 6221 klastrit suurusega 6221,
12) lib.storable.pl – *teek lib.storable,*
13) nimed.txt – numbrid failist *klastrid.txt* ja neile vastavad geenid,
14) nr_rida.storable – *hash*i NR_RIDA hoidmiseks teegi *lib.sotrable* abil,
15) nrid_nimedeks.pl – teisendab *klastrid.txt* nimede kujule,
16) tee_vordluslist.pl – teeb *hash*ile ALL_ASSOTS vastava numbrilise *hash*i NR_RIDA ja salvestab selle *nr_rida.storable*-s,
17) total_search.pl – ühisosade leidmine totaalse võrdelmisega, leiab sama ühisosaga terminite arvu ja nende protsendi kõigist leitud ühisosaga terminitest, maksimaalse ühisosaga termini, suhtelised ülekatted päringute kaupa ja üle päringute,
18) tree_search.pl – ühisosade leidmine sügavuti läbimisel;
19) yle_klastrite.pl – leiab suhtelised ülekatted erinevate klastripikkuste korral.

☑