VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

PREDIKCE HOMOLOGNÍCH SEKVENCÍ PROTEINŮ

PREDICTION OF HOMOLOG PROTEIN SEQUANCES

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. HANA CHLUPOVÁ
AUTHOR

VEDOUCÍ PRÁCE Ing. TOMÁŠ MARTÍNEK, Ph.D.
SUPERVISOR

BRNO 2015

Abstrakt

Predikce a vyhledávání homologních sekvencí proteinů je jednou z řady důležitých oblastí, které jsou v současnosti řešeny v rámci bioinformatiky. Díky určení homologních sekvencí proteinů k hledanému neznámému proteinu je často možné zjistit jeho strukturu a funkci v organismu. Pro vyhledávání homologních sekvencí se často používají nástroje založené na přímém porovnávání sekvencí, další metody využívají profily vzniklé z vícenásobného zarovnání sekvencí nebo provádějí vyhledávání pomocí konstrukce skrytých Markovových modelů. Nejde však určit univerzální metodu, která by převyšovala ostatní. Pro maximální uspokojení požadavků týkajících se například požadované sekvenční identity a poměru správně nalezených a falešně označených homologů je nutné zvolit vhodný typ metody a jejích parametrů. Tato práce se zabývá vývojem nástroje, který na základě provedené analýzy různých metod a jejich nastavení určí, jakou metodu a její parametry by měl uživatel zvolit. Program pak umožňuje spuštění této metody a zobrazení jejich výsledků.

Abstract

Prediction and searching for homologous protein sequences is one of important tasks which are currently being addressed in the area of bioinformatics. According to the determination of homologous sequences of unknown protein sequence it is often possible to determine its structure and function in the organism. For searching homologous sequences, the most frequently used tools are based on direct sequence comparison, profile comparison or on the use of hidden Markov models. There is no universal method better than all others. To satisfy user`s request on needed sequence identity between domains and error rate between founded true positive and false positive pairs, the selection of proper method and its settings is needed. This work is focused to create tool which will help user to choose the best method and its settings according to his requirements. It was created on the basis of the analysis of method results with different settings. In addition, the implemented application offers the possibility to run this method and show its results.

Klíčová slova

predikce homologních sekvencí, skryté Markovovy modely, HMM, proteiny.

Keywords

prediction of homology sequences, hidden Markov models, HMM, proteins.

Citace

Chlupová Hana: Predikce homologních sekvencí proteinů, diplomová práce, Brno, FIT VUT v Brně, 2015

Predikce homologních sekvencí proteinů

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracovala samostatně pod vedením Ing. Tomáše Martínka, Ph.D.
Uvedla jsem všechny literární prameny a publikace, ze kterých jsem čerpala.

……
Bc. Hana Chlupová
20.5.2015

Poděkování

Chcela by som sa poďakovať za odbornú pomoc a smerovanie pri tvorbe tejto práce Ing. Tomášovi Martínkovi Ph.D. Ďalej by som chcela poďakovať za technologickú podporu virtuálnej organizácii MetaCentrum.

Acknowledgment

Access to computing and storage facilities owned by parties and projects contibuting to the National Grid Infrastructure MetaCentrum, provided under the programe ”Project of Large Infrastructure for Research, Development, and Innovations” (LM2010005), is greatly appreciated.

© Hana Chlupová, 2015
Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah 1
1 Úvod 3
2 Princípy molekulárnej biológie 4
2.1 Centrálna dogma molekulárnej biológie 4
2.2 Štruktúra DNA 5
2.3 Uloženie informácií DNA 6
2.4 Proteíny a ich štruktúra 6
2.4.1 Proteínová štruktúra 6
2.4.2 Proteínové domény 7
2.4.3 Konzervované a promiskuitné domény 7
2.4.4 CD-Search 8
2.5 Homológne sekvencie proteínov 8
3 Vyhľadávanie homológnych sekvencií 10
3.1 Dôležité pojmy 11
3.1.1 E-value 11
3.1.2 Formát FASTA 11
3.1.3 Formát PDB 12
3.2 BLAST 12
3.3 PSI-BLAST 13
3.4 CS-BLAST a CSI-BLAST 14
3.5 HMMER 14
3.5.1 HMM 14
3.5.2 Profilové HMM 15
3.5.3 Vyhľadávanie homológnych sekvencií pomocou HMMER 16
3.6 SAM 17
3.7 COMPASS 17
3.8 PROCAIN 18
3.9 HHsearch 18
3.10 HHblits 19
3.11 Metodika testovania nástrojov pre vyhľadávane homológov 19
4 Návrh nástroja na vyhľadávanie homológnych sekvencií proteínov 21
4.1 Princíp činnosti nástroja 21
4.2 Posúdenie úspešnosti zvolených metód 22
4.2.1 Tvorba dátovej sady 22

4.2.2 Štruktúrne zarovnanie 23
4.2.3 Požiadavky na metódy vyhľadávajúce homológne proteíny a ich parametre 24
4.2.4 Postup pri vyhľadávaní možných homológnych proteínových párov 25
4.2.5 Postup vyhodnocovania výsledkov metód 26
4.2.6 ROC krivka 27
4.2.7 Klasifikácia výsledkov metódy 29
4.2.8 Klasifikácia výsledkov metódy pre zadanú oblasť sekvenčnej identity 31
4.3 Návrh a implementácia užívateľskej aplikácie 32
4.3.1 Načítanie zdrojových dát 32
4.3.2 Výber optimálnej metódy vyhľadávania 32
4.3.3 Vyhľadanie homológnych párov 35
4.3.4 Zobrazenie výsledkov 35
5 Meranie úspešnosti metód na zvolenej dátovej sade 36
5.1 Vyhodnotenie na základe typov metód 36
5.1.1 BLASTp 36
5.1.2 PSIBLAST 37
5.1.3 PHMMER 38
5.1.4 JACKHMMER 38
5.1.5 CSBLAST 40
5.1.6 CSIBLAST 40
5.1.7 HHBLITS 41
5.1.8 SAM 42
5.1.9 COMPASS 43
5.2 Vyhodnotenie na základe sekvenčnej identity 44
5.2.1 Namerané najlepšie výsledky podľa sekvenčnej identity 51
6 Testovanie aplikácie a jej ovládanie 53
7 Záver 55

1 Úvod

Jednou z najdôležitejších oblastí výskumu pre pochopenie biologických procesov v živých organizmoch je štúdium proteínov. Využitie má v mnohých oblastiach, či už v medicíne, molekulárnej biológii a genetike, alebo v nanotechnológiach atď. V dnešnej dobe je známe veľké množstvo proteínov a ich počet každoročne rýchlo narastá. A preto narastá aj množstvo proteínov, pri ktorých nepoznáme ich funkciu. Experimentálne zisťovanie však býva často náročné nielen na čas, ale aj na peniaze. Jednou z možností ako teda zistiť funkciu proteínu, je nájdenie podobných (homológnych) proteínov, homológne proteíny mávajú rovnakú alebo podobnú funkciu. Vďaka zisteniu homológie s niektorým zo známych proteínov môžeme teda odvodiť funkciu novo nájdenému proteínu.

Určovanie homológnych sekvencií sa používa aj na modelovanie štruktúry proteínov. Ak je podobnosť sekvencií dvoch homológnych proteínov dostatočne vysoká, je možné na základe známej štruktúry proteínu zostaviť štruktúru druhého.

V dnešnej dobe existuje už veľa nástrojov na vyhľadávanie homológnych sekvencií, niektoré z nich sú založené na priamom porovnávaní zadanej sekvencie so sekvenciami z databázy, iné využívajú profily vzniknuté z viacnásobného zarovnania sekvencií a ďalšie prevádzajú vyhľadávanie pomocou konštrukcie skrytých Markovových modelov. Pri vyhľadávaní je možné využiť aj vlastnosti konzervácie určitých domén v rámci rodín proteínov.

Táto práca sa zaoberá problematikou vyhľadávania homológnych sekvencií proteínov. V tejto oblasti existuje množstvo rôznych metód, ale každá pracuje na mierne inom princípe a dáva hodnotné výsledky len pre istú špecifickú oblasť dát. Hlavným cieľom práce bolo nájsť riešenie, vyvinúť aplikáciu, ktorá bude na základe uskutočnených testov pomáhať užívateľovi rozhodnúť sa, akú metódu má v danom prípade použiť. Riešenie bude teda založené na využití výsledkov porovnania metód a podľa ich úspešnosti pri hľadaní párov s danou sekvenčnou identitou bude pomáhať užívateľovi vybrať nie len správnu metódu, ale aj jej nastavenie s odpovedajúcimi parametrami tak, aby splnili požadované kritériá. Určenie ideálneho nastavenie prebehne podľa vyhodnotenia úspešnosti hľadania metód podľa typu vstupu a požiadaviek. Výsledok tejto práce bude aplikácia, ktorá sama rozhodne o type použitej metódy a jej nastaveniach tak, aby výsledkom bola sada homológnych párov proteínov s parametrami, ktoré užívateľ požadoval.

V úvode práce (kapitola 2) je stručné načrtnutie problematiky z biologického hľadiska. Teda princípy molekulárnej biológie ako je štruktúra a uloženie informácie v DNA, proteíny a ich štruktúra. Následne v kapitole 3 sú popísané v dnešnej dobe asi najpoužívanejšie metódy vyhľadávania homológnych sekvencií proteínov. V kapitole 4 je popísaný návrh samotného nástroja vrátane popisu metodiky pri hodnotení výsledkov testovaných metód, ktorý slúži ako zdroj informácií pre aplikáciu, podľa ktorej sa rozhoduje o type a parametroch použitej metódy. V kapitola 5 je potom

samotné zhrnutie výsledkov testovania existujúcich metód na testovacej sade dát. Posledná kapitola 6 popisuje testovanie implementovanej aplikácie spolu s postupom jej ovládania.

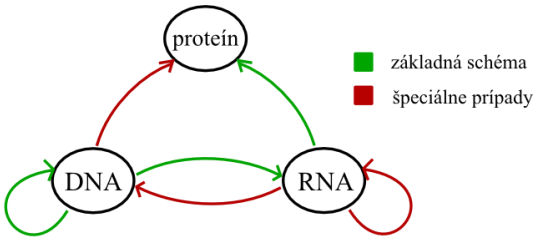
2 Princípy molekulárnej biológie

Molekulárna biológia je relatívne mladý vedný obor, ktorý skúma molekulárny vzťah štruktúry a interakcií biologických makromolekúl k funkciám a vlastnostiam živých sústav [1]. Jej základy boli položené v tridsiatych rokoch minulého storočia a názov jej dal Warren Weaver.

Molekulárna biológia je založená na vedomostiach z oblasti genetiky a biochémie. Jej úlohou je pozorovať zmeny v molekulárnej stavbe organizmov a účinky týchto zmien na život daného organizmu. Zaoberá sa vzťahmi medzi bunkovými systémami, ako napríklad DNA a RNA, ďalej skúma, ako sú tieto vzťahy regulované. Molekulárna biológia sa snaží vysvetliť princípy genetickej regulácie a genetického kódu.

2.1 Centrálna dogma molekulárnej biológie

Keďže molekulárna biológia sa zaoberá štúdiom molekúl, zaujímajú nás predovšetkým procesy translácie, transkripcie a replikácie genetického materiálu [2]. Je zobrazená na obrázku 2.1.



Obrázok 2.1.: Centrálna dogma molekulárnej biológie

Replikácia predstavuje prepis DNA na DNA, kedy sa vytvára komplementárne vlákno k vláknu pôvodnému. Helikázy, špeciálne enzýmy, pri tomto procese rozmotávajú dvojzávitnicu, následne sa pomocou polymerázy pripojujú potrebné bázy, ktoré sú spojené pomocou ligázy. Po dokončení máme dve identické dvojzávitnice.

Transkripcia prebieha v smere z DNA na RNA, z DNA sa transkripciou vytvorí pre-mRNA, ktorá obsahuje introny (nekódujúca časť) a exony (kódujúca časť), toto platí pre všetky eukaryoty. Pre zaujímavosť, len 1,5 % ľudskej DNA tvoria gény kódujúce proteíny [3]. Po prevedení zostrihu sa z pre-mRNA stane mRNA, ktorá už neobsahuje žiadne introny a už nie je uložená v jadre, ale v cytoplazme.

Translácia potom predstavuje prevod mRNA na proteín. Poradie nukleotidov v mRNA je prevedené na poradie aminokyselín v polypeptidovom reťazci. Jednotlivé aminokyseliny sú

prenášané tRNA, druh aminokyseliny je určený kodónom (teda trojicou nukleotidov) mRNA a komplementárnym antikodónom tRNA.

Okrem základných procesov existujú aj špeciálne prípady prevodu a to:

- reverzná transkripcia, teda prevod RNA do DNA, príkladom sú niektoré vírusy, ktoré vedia zapísať svoju RNA do DNA hostiteľovej bunky,
- RNA na RNA, príkladom sú vírusy, ktoré vedia zapísať svoju RNA do inej RNA,
- DNA na proteín, tento prípad je možný len v laboratórnych podmienkach.

2.2 Štruktúra DNA

DNA alebo deoxyribonukleová kyselina je nositeľkou genetickej informácie v každom jedincovi. Jej úlohou je delenie, riadenie rastu a obnova buniek. Prvýkrát bola popísaná v roku 1953 Jamesom D. Watsonom a Francisom Crickom [3].

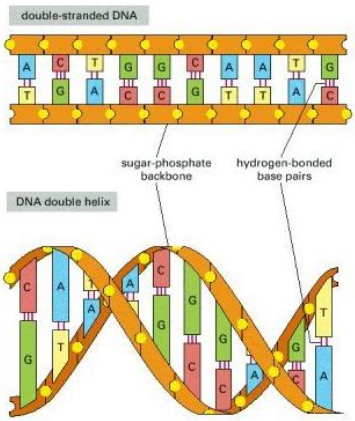
Základnou zložkou DNA sú nukleotidy, tie pozostávajú z troch hlavných zložiek:

1. molekuly deoxyribózy,
2. fosfátového zvyšku kyseliny fosforečnej,
3. dusíkatej bázy.

Prvé dve zložky tvoria pentózafosfátovú kostru a majú za úlohu udržať bázy v správnej polohe a vzdialenosti. Dusíkaté bázy delíme na dve skupiny:

- puriny: **a**denín a **g**uanín
- pyrimidiny: **c**ytozín a **t**ymín, prípadne **u**racil

Tieto dusíkaté bázy tvoria štruktúru DNA a spojujú vlákna dvojzávitnice DNA, tá je zobrazená na obrázku 2.2. Pomocou dvoch vodíkových väzieb sa spája adenín s tymínom (v RNA s uracilom) a pomocou troch vodíkových väzieb guanín s cytozínom. Iné kombinácie sú nevýhodné. Toto párovanie nazývame Watson-Crickovo párovanie báz [4]. Pričom je zachovaná komplementarita báz. Poradie dusíkatých báz je podstatné pre prenos genetickej informácie. Dedičnosť je založená na génoch, pričom gén je sekvencia nukleotidov, obsahujúca informáciu pre syntézu istého produktu [1].



Obrázok 2.2.: Štruktúra DNA [3]

DNA je uložená v jadre eukaryotu a voľne v cytoplazme prokaryotu. Primárna štruktúra určuje poradie aminokyselín v bielkovinách. DNA sa bežne vyskytuje ako tzv. pravotočivá dvojzávitnica, kedy sa dve vlákna DNA točia okolo spoločnej osi. Veľmi dôležitou vlastnosťou DNA je aj to, že je smerovaná, rozlišujeme dva konce podľa ukončenia reťazca. Ak je zakončená fosfátom, ide o 5' koniec, ak OH skupinou, ide o 3' koniec. Jedna závitnica je teda v smere 3'->5', druhá v smere 5'->3' a vzájomne sú antiparalelné. Nukleotidy podľa dohody zapisujeme vždy v smere 5'->3'[3].

2.3 Uloženie informácií DNA

Genetická informácia je informácia uložená v sekvencii nukleotidov a nachádza sa vo všetkých živých organizmoch, ale aj vo vírusoch. Jej uloženie a možnosť prenášať ju neporušenú na potomkov je jednou zo základných vlastností života. Je v nej totižto zakódovaná informácia o poradí aminokyselín v bielkovinách. Uloženie DNA záleží od typu organizmu, môže byť voľne v cytoplazme, v jadre, ale aj v mitochondriách a plastidoch. Kompletnú genetickú informáciu daného organizmu potom nazývame **genóm** [3].

Genetický kód sa skladá z kodónov, čo sú tri po sebe nasledujúce bázy, keďže poznáme 4 bázy, máme až 4^3 možných kodónov. Kodón je následne preložený na jednu z dvadsiatich aminokyselín, pričom viacero kodónov môže kódovať jednu aminokyselinu. Niektoré kodóny majú špeciálny význam, rozlišujeme tzv. štart a stop kodóny, medzi ktorými sa nachádza čítací rámec.

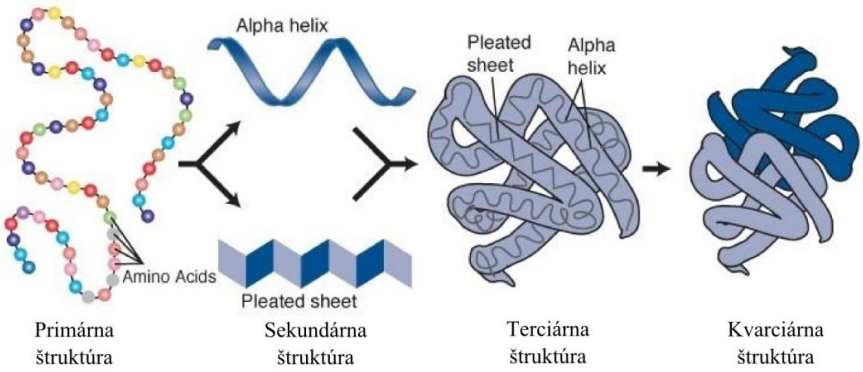
Ľudská molekula DNA obsahuje približne 3 G báz, každá báza môže byť zakódovaná na 2 bity, na uloženie celého genómu teda potrebujeme 750 MB. Pri ukladaní DNA do databázy stačí uložiť len jeden z reťazcov, najčastejšie v smere 5'->3' a ďalší reťazec sa dopočíta na základe komplementarity. Je dôležité myslieť na to, že ak chceme DNA analyzovať, musíme brať do úvahy obe vlákna[4].

2.4 Proteíny a ich štruktúra

Proteíny sú polypeptidové polyméry skladajúce sa z lineárne usporiadaných aminokyselín a sú jedným z najdôležitejších prvkov každej bunky. Majú viacero funkcií od výstavby bunky až po prepravu materiálu medzi jednotlivými bunkami.

2.4.1 Proteínová štruktúra

Proteínová štruktúra je definovaná v štyroch úrovniach. Primárna štruktúra odkazuje na lineárnu sekvenciu aminokyselín v polypeptidickom reťazci. Sekundárna štruktúra popisuje usporiadanie primárnych sekvencií do vzorov (α-helix a β skladaný list a mostík). Terciálna štruktúra je trojdimenzionálne usporiadanie, vznikajúce pri balení sekundárnej štruktúry do domény. Kvarciárna štruktúra je spojenie polypeptidových reťazcov pri vytváraní funkčného proteínu [4]



Obrázok 2.3 – Štruktúra proteínov [3]

2.4.2 Proteínové domény

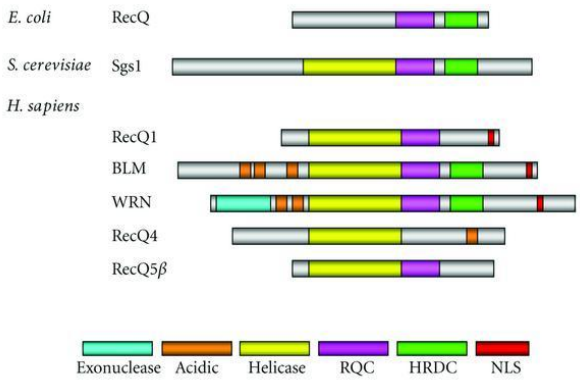
Proteínová doména je konzervovaná časť proteínu, ktorá sa môže vyvíjať, fungovať a existovať nezávisle na zvyšku proteínového reťazca. Každá doména vytvára kompletnú trojdimenzionálnu štruktúru a môže byť samostatne stabilná a zbalená. Väčšina proteínov pozostáva z niekoľkých štrukturálnych domén a jedna doména sa môže vyskytovať v rôznych proteínoch.

V molekulárnej biológii domény fungujú ako stavebné prvky, pri spojení viacerých domén teda vznikajú proteíny s rozličnými funkciami. Domény často vytvárajú samostatné funkčné jednotky. Keďže sú takto nezávislé, v genetickom inžinierstve sa používajú vo výskume pri zostavovaní nových proteínov, tzv. proteínových chimér. Domény majú rôznu dĺžku od 25 až po 500 aminokyselín [4].

2.4.3 Konzervované a promiskuitné domény

Konzervované domény sú také domény, ktoré obsahujú konzervované sekvenčné motívy s rôznou funkciou. Tie umožňujú detekciu domény v polypeptidických sekvenciách. Je možné ich tiež definovať ako jednotky, ktoré sa opakujú počas molekulárnej evolúcie.

Pri hľadaní homológnych proteínov k zadanej sekvencii neznámeho proteínu je možné využiť konzervované domény na určenie rodiny, do ktorej by mohli patriť. Z obrázku 2.4 je zrejmé, ako členovia rodiny zdieľajú spoločné konzervované domény.



Obrázok 2.4. – Zarovnanie proteínov podľa konzervovaných domén [5]

Domény sa líšia v ich náchylnosti na vytváranie viacdoménových proteínov. Zatiaľ čo niektoré domény sa môžu vyskytovať len v určitých kombináciách a pozíciách, iné sa môžu vyskytovať v rôznych doménových architektúrach proteínov. Tie sa potom nazývajú mobilné, alebo promiskuitné a je pri nich udaný stupeň mobility. Tieto domény majú významnú úlohu v interakcii medzi doménami proteínov a pri prenosoch signálov bunkou [6].

Existuje mnoho doménových databáz, ktoré umožňujú detekciu proteínových domén ako napríklad **Pfam** [7] alebo **CDD** [8]. Databázy používajú metódy pracujúce so sekvenciami alebo štruktúrami, ktoré rozpoznávajú oblasti v proteínoch patriace do určitej doménovej rodiny. Na vyhľadanie konzervovaných domén bolo vyvinutých niekoľko metód, asi najpoužívanejšou je CD-Search, ktorá spolupracuje s databázou CDD.

2.4.4 CD-Search

Conserved Domain Search (CD-Search) je nástroj pre detekciu štrukturálnych a funkčných domén v proteínových sekvenciách. Využíva heuristické funkcie známe z nástroja BLAST na vyhľadávanie doménových profilov. Nájdené domény zo zadanej sekvencie môžu byť zobrazené ako špeciálne značky s názvom domény pozdĺž sekvencie. Je možné zobraziť aj viacnásobné zarovnania s doménovo odpovedajúcimi sekvenciami z databázy.

Vstupom metódy je sada sekvencií, v ktorých sú vyhľadávané známe domény. Modely domén sú získavané z doménových databáz (napr. CDD), tie sú skonvertované na pozične špecifické skórovacie matice. Zadané sekvencie sú potom s využitím týchto matíc preskenované algoritmom RPS-BLAST (variant PSI-BLAST).

CD-Search používa rovnaké ohodnotenie nájdených zarovnaní ako BLAST. Spolu s nájdenými zarovnaniami vracia ich hodnotu E-value (viď 3.1.1) [9].

2.5 Homológne sekvencie proteínov

Pojem biologická homológia je používaný napr. pre segmenty, ktoré sú založené na spoločnom pôvode, alebo zdieľajú podstatnú podobnosť. Na základe štruktúrnej podobnosti segmentov môžeme odvodiť vývojový pôvod organizmov.

Za homológne proteíny sa považujú tie, ktoré majú sekvencie, pri ktorých očakávame, že zdieľajú rodový pôvod, a teda boli odvodené z jednej pôvodnej sekvencie. Homológiu sekvencií rozdeľujeme na dva druhy – *orthológne* a *paralógne* [4].

- Orthológne sekvencie sú homológne sekvencie v rôznych druhoch, ak vznikli z génu spoločného predka pri *speciácii* (jeden druh sa rozdelil do dvoch, v oboch sa nachádza kópia rovnakého génu).
- Paralógne sekvencie sú homológne sekvencie, ktoré vznikli pri génovej duplikácii.

Pri určovaní homológie proteínov sa používajú aj pojmy identita a podobnosť:

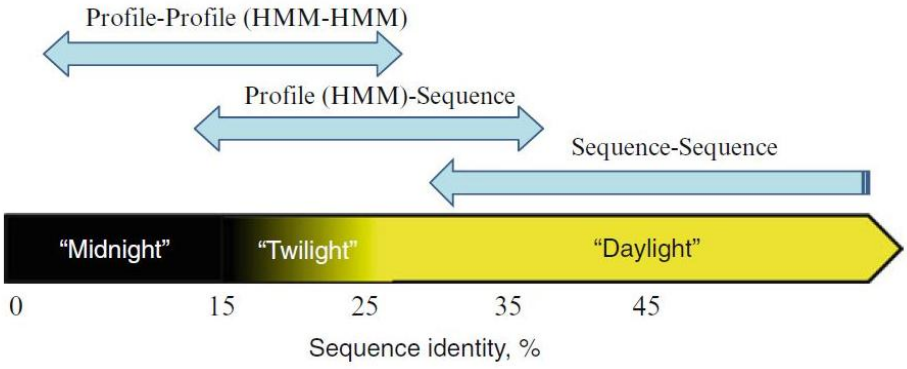
- Identita (sekvenčná) znamená, že dve sekvencie sú úplne rovnaké, prípadne uvádzame v percentách, na koľko sú dve sekvencie identické (napr. ATCG a ATCG sú identické na 100 %, kým ATCC a ATCG sú identické iba na 75 %).
- Podobné sekvencie zdieľajú podobné biochemické vlastnosti. Potom homológne páry sú štruktúrne a funkčne príbuzné. Pri konštrukcii zarovnania pre homológne sekvencie môže prísť aj k párovaniu aminokyselín, ktoré nie sú identické, ale patria do rovnakej skupiny [4].

3 Vyhľadávanie homológnych sekvencií

V závislosti na evolučnej vzdialenosti medzi proteínmi sú používané rôzne zložité metódy vyhľadávania ich homológií. Tieto môžeme rozdeliť do niekoľkých skupín, podľa toho, ako komplexná je informácia o proteínoch, s ktorými pracujeme:

- zarovnanie páru sekvencií,
- zarovnanie profil – sekvencia a skryté Markovove modely (HMM),
- zarovnanie profil – profil a HMM – HMM.

Ak hľadáme blízke homológie, teda také, ktoré majú vysokú sekvenčnú identitu, malo by byť dostatočné aj najjednoduchšie zarovnanie párov sekvencií. Ak ide o vzdialenejšie príbuzné sekvencie, je vhodné vytvoriť HMM profil a ten použiť pre vyhľadávanie [10]. Toto je znázornené na obrázku 3.1.



Obrázok 3.1. – Porovnanie skupín metód podľa identity sekvencií s ktorou pracujú [10].

Na hľadanie príbuzných proteínov sa často využívajú proteínové databázy, ktoré obsahujú rôzne vzorky proteínov. Nájdenie homológnych sekvencií, ktoré sú známe, môže umožniť určenie štruktúry alebo funkcie neznámeho proteínu.

Asi najpoužívanejšou sekvenčnou databázou je Non-Redundant databáza (NR), ktorá je spravovaná organizáciou National Centre for Biotechnology Information (NCBI) a je zacielená na jej vyhľadávač BLAST. Preto používa formát Protein Data Bank (PDB [11]), ale je možné s ňou pracovať aj vo formáte FASTA. Údaje v databáze sú vždy unikátne. Úplne identické sekvencie tvoria len jeden záznam, obsahujúci záznamy zo všetkých čiastočných databáz, ako je napríklad UniProt a Swisprot [12].

Na vyhľadávanie s profilmi alebo HMM profilmi sa často používa databáza Pfam proteínových rodín. Tá obsahuje anotácie a viacnásobne zarovnané sekvencie, ktoré sú generované HMM. Pfam-A je manuálne spravovaná databáza a obsahuje okolo 10 000 záznamov. Tieto záznamy nepokrývajú všetky známe proteíny, preto vznikla databáza proteínových rodín odvodených algoritmom, Pfam-B, ale tá je považovaná za menej kvalitnú.

Na kategorizáciu známych proteínov bola vytvorená hierarchia Structural Classification of Proteins (**SCOP** [13]), ktorá zahŕňa proteíny a ich domény, ktoré je možné zaradiť do spoločných

kategórií. Táto klasifikácia tvorí strom so 7 poschodiami: najvyššie je koreňový uzol, ktorý obsahuje 11 základných tried (*classes*), tie sa ďalej členia na foldy (*folds*), tie následne na nadrodiny (*superfamilies*), rodiny (*families*), proteíny (*proteins*) a tie nakoniec na domény. SCOP je používaná aj na kontrolu nájdených homológnych sekvencií na falošné zhody. Pritom sa berie do úvahy, či nájdené homológne páry patria do rovnakej nadrodiny alebo rodiny.

V ďalšom texte budú popísané rôzne vyhľadávacie nástroje, pričom väčšina je dostupná ako voľne stiahnuteľný softvér, alebo ako webový server.

3.1 Dôležité pojmy

3.1.1 E-value

Expected value (E-value) je hodnota, ktorá vyjadruje počet rôznych zarovnaní sekvencií so skóre zhodným alebo vyšším než získané skóre na základe Smith-Watermanovho algoritmu (*raw score*), ktorých výskyt je očakávaný pri náhodnom vyhľadávaní v databáze. Je možné ho interpretovať aj tak, že overuje vierohodnosť nájdenej zhody. Znižuje sa exponenciálne s hodnotou základného skóre. Nízke hodnoty (pod 10^{-8}) vypovedajú, že nájdená zhoda je homológiou. Hodnoty nachádzajúce sa v rozmedzí 10^{-8} až 10^{-4} označujú množinu pravdepodobných homológií a hodnoty nad 10^{-4} naznačujú, že o homológiu nejde.

Pre výpočet E-value sa používa nasledujúci vzťah (1):

$$E = Kmne^{-\lambda S} \qquad (1)$$

V tomto vzorci koeficient K určuje mierku priestoru pre vyhľadávanie a koeficient λ mierku skórovacieho systému. Hodnoty m a n udávajú dĺžky oboch porovnávaných sekvencií a hodnota S odpovedá základnému skóre pre zarovnanie dvoch sekvencií [14].

3.1.2 Formát FASTA

Formát FASTA je textový formát, reprezentujúci nukleotidové alebo peptidové sekvencie, v ktorom sú nukleotidy alebo aminokyseliny reprezentované jednoduchým písmenovým kódom [14].

Sekvencia zapísaná vo formáte FASTA začína symbolom “>” nasledovaným jednoduchým popisom sekvencie na prvom riadku. Na ďalších riadkoch sú dáta vlastnej sekvencie. Na jednom riadku by nemalo byť viac ako 60 znakov. Sekvencia končí, ak je na ďalšom riadku znova znak “>”, začínajúci ďalšiu sekvenciu. Príklad proteínu uloženého vo FASTA formáte vyzerá nasledovne:

```
>tr|Q988A4|Q988A4_RHILO Mll6828 protein - Rhizobium loti (Mesorhizobium loti).
MVTHPSLVLAATIALAVVLGAISIADFRRQIIPDGLNLALAGIGLSYQLAADADAMPQRL
LFAAATFAAAWLLRRGHFLMTGRIGLGLGDVKMLAAASCWISPLLLPVLLFIASASALLF
VGGQVVATGPAAARARVAFGPFIAIGLGASWALEQFAGLDMGLL
```

3.1.3 Formát PDB

Formát PDB [11] je jednoduchý textový formát, ktorý popisuje 3-dimenzionálne štruktúry molekúl. Sú v ňom uložené informácie o anotácii proteínov, ďalej obsahuje aj popis sekundárnej štruktúry, súradnice atómov a ich prepojenie. Každý riadok v súbore obsahuje na začiatku kľúčové slovo popisujúce typ informácie na riadku.

Medzi tie základné patria informácie o autorovi a názve záznamu (značky *HEADER*, *TITLE*, *AUTHOR*), popis anotácie (značka *REMARK*) či odkaz na záznam do proteínovej databázy (*DBREF*). Tento záznam obsahuje názov proteínu a názov reťazca, ktorého sekvencia je ďalej v súbore popísaná. Ďalšie dôležité informácie z pohľadu metód pre vyhľadávanie homológov sa skrývajú na riadkoch, ktoré začínajú značkou *SEQRES*. Na tých je okrem označenie reťazca, ku ktorému patrí, uvedená aj sekvencia aminokyselín.

3.2 BLAST

Basic Local Alignment Search Tool (BLAST) je jeden z najrozšírenejších nástrojov na porovnávanie sekvencií. Umožňuje zistiť, ktorému známemu proteínu je zadaná sekvencia podobná. Porovnáva nukleotidové alebo proteínové sekvencie so sekvenčnými databázami a ohodnocuje nájdené zhody. Je dobre využiteľný pre odvodenie funkcie a evolučných vzťahov medzi sekvenciami, ale aj pre nájdenie génových rodín [15].

Algoritmus BLAST bol vyvinutý Stephenom Altschulom v roku 1990. Ide o heuristický algoritmus, ktorý je rýchlejší než predtým používané prístupy ako bolo zisťovanie optimálneho globálneho zarovnania sekvencií (algoritmus Needleman-Wunsch [16]) alebo lokálneho zarovnania (algoritmus Smith-Watermanov [17]). Tieto staršie prístupy sú v porovnaní s algoritmom BLAST síce pomalšie, na druhej strane vždy nájdu optimálne zarovnanie medzi dopytovanou sekvenciou a sekvenciami z databázy.

Základný variant BLAST umožňuje porovnávanie sekvencie so sekvenciou, alebo sekvencie so sekvenčnou databázou. Základnou myšlienkou je využívanie párov sekvencií s vysokým ohodnotením (High Scoring Segment Pairs – **HSP**). Vstupom je najčastejšie sekvencia vo formáte FASTA (viď kapitola 3.1.2) alebo GenBank a váhová matica, výstupom je grafické zobrazenie vo formáte tabuľky, ktorá obsahuje identifikátory nájdených sekvencií, ich zarovnanie a skóre. Algoritmus BLAST sa skladá z troch fáz [18]:

- Osievanie – hľadaná sekvencia je rozdelená na časti (slová) veľkosti *w*. Pre *w* platí, že čím má menšiu hodnotu, tým viac výskytov je nájdených a algoritmus má vyššiu citlivosť, ale narastá doba výpočtu. Rozdelenie na slová prebehne až po tom, čo sú odstránené časti DNA, ktoré sú málo vypovedajúce (napr. repetície, nejasné oblasti atď.),

– pre každú časť je nájdená množina alternatívnych slov. Slovám je pridelené skóre podobnosti, najčastejšie maticou BLOSUMxx [19]. Do tabuľky podobnosti sú uložené slová so skóre vyšším než zvolený prah T. Pre hodnotu T platí, že čím je hodnota vyššia, tým presnejšie zhody sú nájdené a výpočet je rýchlejší, ale je nájdených menej výskytov.

- Rozširovanie – slová uložené v tabuľke podobnosti sú rozšírené na obe strany. Ak je skóre nad požadovanou hodnotou, sekvencia je odovzdaná na výstup. Ak klesne pod zvolený prah, rozširovanie je ukončené, na výstup ide sekvencia nájdená predtým, ako začalo skóre klesať,
  – výstupom tejto fázy je zoznam segmentov s najvyšším skóre.
- Ohodnotenie – vo výsledku nás zaujímajú skupiny konzistentných úsekov, ktoré na seba nadväzujú. Zo zoznamu vyradíme segmenty s nízkou hodnotou *E-value* (viď 3.1.1). Samotné ohodnotenie sa líši podľa typu úlohy [20].

V priebehu rokov bolo okrem základných variant ako je BLASTn a BLASTp, vyvinutých viacero vylepšení algoritmu BLAST, ako sú napríklad PSI-BLAST, PHI-BLAST atď. [20].

3.3 PSI-BLAST

Position-Specific Iterated BLAST je jedným z vylepšení algoritmu BLAST, je oveľa presnejší v hľadaní vzdialene príbuzných proteínov a nových členov z danej rodiny proteínov. Iteratívne vyhľadáva v proteínovej databáze podobné sekvencie k jednej alebo viacerým zadaným proteínovým sekvenciám. Rozdiel medzi PSI-BLAST a BLAST je v type použitej skórovacej matice. BLAST používa preddefinovanú skórovaciu maticu (napr. BLOSUM62), zatiaľ čo PSI-BLAST používa pozične špecifickú skórovaciu maticu (Position Specific Scoring Matrices – **PSSM**), ktorá je odvodená až počas vyhľadávania [21].

PSI-BLAST by mal byť schopný nájsť aj vzdialene príbuzné sekvencie, ktoré môžu byť prehliadnuté pri vyhľadávaní základnou metódou BLAST. Pre odvodenie novej PSSM sa používa viacnásobné zarovnanie párov sekvencií s vysokým ohodnotením (HSP) nájdených v predchádzajúcej iterácii, takto vzniknutá PSSM sa použije v nasledujúcej iterácii hľadania. Pre odvodenie sa vždy použijú iba páry, ktoré majú hodnoty E-value nižší ako zadaný prah (východzia hodnota býva 0,005). Pri prvej iterácii PSI-BLAST, zvykne byť namiesto PSSM použitá štandardná skórovacia matica.

Algoritmus použitý v PSI-BLAST je iteratívny, vyhľadávanie homológov končí v prvej iterácii, kedy sa nenájdu už žiadne nové sekvencie. Toto je možné upraviť tak, že sa vykoná iba zadaný počet iterácií.

Vyhľadávanie sekvencií nástrojom PSI-BLAST prebieha v nasledujúcich krokoch [22]:

1. Vstupom je jedna proteínová sekvencia, ktorá bude porovnaná s proteínovou databázou využitím metódy BLASTp.

2. Algoritmus skonštruuje viacnásobné zarovnanie z významných lokálnych zarovnaní získaných v predchádzajúcom kroku. Z toho je potom skonštruovaná PSSM.
3. Vyhľadanie nových lokálnych zarovnaní v proteínovej databáze pomocou získanej PSSM.
4. PSI-BLAST štatisticky odhadne významnosť nájdených lokálnych zarovnaní medzi pôvodnou sekvenciou a sekvenciami z databázy.
5. Ak je splnená ukončujúca podmienka, teda bol dosiahnutý zadaný počet opakovaní, alebo metóda prestala konvergovať, je algoritmus ukončený.
   Ak nie, pokračuje sa znova od kroku 2.

3.4 CS-BLAST a CSI-BLAST

Context-Specific BLAST (CS-BLAST) bol v roku 2009 vyvinutý ako ďalšie rozšírenie pôvodného algoritmu BLAST. Obdobne *Context-Specific Iterated BLAST* (CSI-BLAST) je kontextovo špecifická analógia nástroja PSI-BLAST [23].

Úlohou CS-BLAST je zostrojiť počiatočnú PSSM za použitia dopytovanej sekvencie a knižnice krátkych profilov. Na vytvorenie tejto knižnice autori zostrojili mnoho viacnásobných zarovnaní sekvencií porovnaním podmnožín sekvencií z celej NR proteínovej databázy, s inou pomocou dvoch iterácií PSI-BLAST. Viacnásobné zarovnania sekvencií sú prevedené na frekvenčné profily aminokyselín, rozdelené do krátkych okien a zoskupené do knižnice profilov.

CS-BLAST určuje pravdepodobnosť výskytu kontextovo špecifickej mutácie, ktorá je závislá na 13 aminokyselinách (hlavné aminokyseliny a 12 susedných). Takto určená pravdepodobnosť je presnejšia ako pravdepodobnosť výskytu podľa substitučnej matice, čím sa zlepšuje zarovnanie a vyhľadávanie sekvencií [23].

Pre porovnanie, CS-BLAST nájde až o 50 % viac homológnych proteínov než BLAST pri rovnakom nastavení, pričom zarovnania CS-BLAST sú omnoho citlivejšie a presnejšie predovšetkým pre komplikované zarovnania. CSI-BLAST dosahuje lepšiu citlivosť ako PSI-BLAST a pri dvoch iteráciách CSI-BLAST spravidla nájde viac sekvencií než s piatimi iteráciami PSI-BLAST [24].

3.5 HMMER

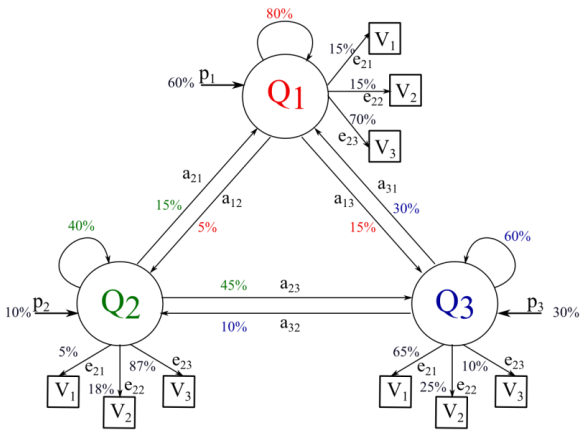
Ako HMMER sa označuje sada nástrojov pre biologickú analýzu sekvencií s využitím profilových skrytých Markovových modelov (Hidden Markov Model – **HMM**).

3.5.1 HMM

Skrytý Markovov model (HMM) je štatistický model pre sekvencie s diskrétnymi symbolmi. Dlho sa používal pri problémoch rozpoznávania reči, neskôr sa začal používať aj v oblastiach bioinformatiky. HMM môže byť zapísaný ako pätica (Q,V,p,A,E), kde:

- **Q** je konečná množina stavov,
- **V** je konečná množina pozorovaných symbolov (abeceda),
- **p** je počiatočná pravdepodobnosť stavu,
- **A** sú pravdepodobnosti prechodov medzi stavmi, pravdepodobnosť prechodu medzi stavmi $s, t \in Q$ označíme ako a_{st},
- **E** je matica pravdepodobností, že daný symbol bude výstupom stavu. Pravdepodobnosť, že výstupom stavu $s \in Q$ bude symbol $x \in V$, označíme ako e_{sx}.

Výstupom HMM sú symboly, ktoré boli výstupmi prechádzaných stavov pri ceste medzi zadaným počiatočným a koncovým stavom. Samotná cesta medzi stavmi je skrytá [25]. Príklad HMM na obrázku 3.2. a popisuje situáciu, kedy máme tri stavy a tri možné symboly, ktoré môžu byť výstupom modelu.



Obr. 3.2. – Príklad HMM s troma stavmi a troma symbolmi s ilustračnými percentuálnymi hodnotami.

HMM môžu byť účinne aplikované na problémy v bioinformatike, ako sú:

- zisťovanie sekundárnej štruktúry proteínov,
- viacnásobné zarovnanie sekvencií,
- hľadanie génov.

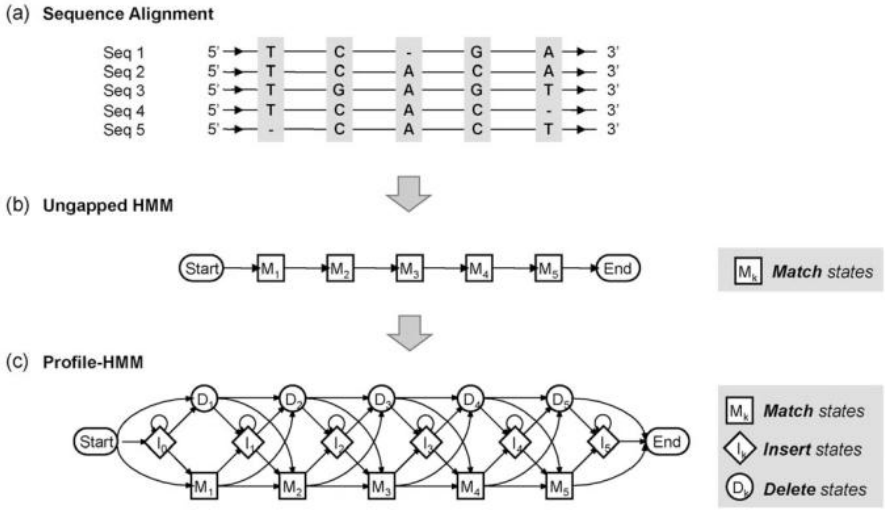
Na nájdenie najpravdepodobnejšej sekvencie symbolov sa najčastejšie používa Viterbiho algoritmus [26], ten je založený na technike dynamického programovania.

3.5.2 Profilové HMM

Ide o štatistický model, založený na viacnásobnom zarovnaní. Tento model zachytáva informáciu o tom, ako často sa v danom stĺpci vyskytuje dané rezíduum. Na rozdiel od klasického HMM sú profilové HMM vždy lineárnou štruktúrou, orientovanou zľava doprava, ktorá neobsahuje žiadne cykly [27]. Profilový HMM používa tri typy skrytých stavov:

- zhoda (match states) M_k – ak aktuálny symbol v sekvencii odpovedá aktuálnemu symbolu,
- vloženie (insert states) I_k – ak symbol nie je prítomný v modeli, len v sekvencii,
- mazanie (delete states) D_k – ak symbol je prítomný v modeli, ale nie je v sekvencii.

Postup pri vytváraní profilového HMM je znázornený na obrázku 3.3. Z neho je zrejmé, že počet stavov *k* odpovedá veľkosti viacnásobného zarovnania. Stavy M_k odpovedajú situácii, kedy porovnávaná sekvencia na *k*-tom mieste odpovedá pôvodnému viacnásobnému zarovnaniu. Ak porovnávaná sekvencia je dlhšia než pôvodné zarovnanie, tak vkladanie nezhodných symbolov je modelované stavmi I_k. Stavy D_k sa vyskytujú, ak je porovnávaná sekvencia kratšia, teda jeden alebo viacero symbolov v zarovnaní je naviac oproti porovnávanej sekvencii [25].



Obrázok 3.3. – Vytvorenie profilového HMM z viacnásobného zarovnania sekvencií. Kde (a) znázorňuje viacnásobné zarovnanie, (b) HMM bez vkladania medzier a (c) odpovedajúce profilové HMM [25].

Profilové HMM sa často používa pre reprezentáciu vzorov v rodine DNA sekvencií alebo proteínov. S databázami založenými na profilových HMM pracuje viacero nástrojov na hľadanie homológnych sekvencií, ako je HMMER (kapitola 3.5.3) a SAM (kapitola 3.6).

3.5.3 Vyhľadávanie homológnych sekvencií pomocou HMMER

Sada nástrojov HMMER obsahuje program na vyhľadávanie homológnych sekvencií proteínov v sekvenčných databázach. Obsahuje aj ďalšie nástroje napr. na vytvorenie viacnásobného zarovnania proteínov, či vytváranie profilových HMM. Navrhol ho Sean Eddy s výskumnou skupinou z Janelia Farm Research, ktorá pôsobí na inštitúte Howarda Hughesa od roku 2010.

HMMER môže pracovať aj s jednoduchými sekvenciami, ale jeho prednosťou je práca s viacnásobným zarovnaním sekvencií nad rodinou sekvencií. Pri tom si vytvára HMM profil dotazu, ktorý priraďuje pozične špecifický skórovací systém pre substitúcie (zhodu), vloženia a vymazania [28]. Pri určovaní ako sa zadaná sekvencia alebo profil zhoduje s existujúcim HMM profilom sa počíta skóre nazývané ako *log-odds ratio* [29] najpravdepodobnejšieho zarovnania pomocou Viterbiho algoritmu. V novej verzii HMMER3 sa namiesto neho používa *Forward-Backward* algoritmus [4].

HMMER vznikol, aby nahradil staršie metódy ako BLASTp a PSI-BLAST pri hľadaní zadanej sekvencie v databáze. Oproti nim je presnejší a dokáže nájsť aj vzdialenejšie homológy. V prvej verzii HMMER však boli problémy s výpočtovou náročnosťou, ktorá spôsobovala, že výpočet bol až stokrát pomalší než pri metóde BLAST. Nová verzia HMMER3 sa jej už rýchlosťou vyrovná [30].

Sada HMMER zahŕňa tieto programy na vyhľadávanie sekvencií proteínov v databáze:

- phmmer – vyhľadáva zadanú sekvenciu v sekvenčnej databáze (ako BLASTp),
- jackhmmer – iteratívne vyhľadáva sekvenciu v sekvenčnej databáze (ako PSI-BLAST),
- hmmsearch – vyhľadáva profilovú HMM v sekvenčnej databáze,
- hmmscan – vyhľadáva sekvenciu v databáze profilov HMM.

3.6 SAM

Sequence Alignment and Modeling system (SAM) je súbor softwarových nástrojov na vytváranie, úpravu a používanie skrytých Markovových modelov na analýzu sekvencií. Základom metódy je iteratívny algoritmus, ktorý vytvára HMM z jednej proteínovej sekvencie alebo zarovnania použitím iteratívneho prehľadávania databázy. Táto metóda je veľmi podobná metóde HMMER.

Postup pri vyhľadávaní homológnych sekvencií sa skladá z niekoľkých krokov. Najprv sa generuje HMM model. Ten môže byť vytrénovaný automaticky, ak mu je zadaná skupina príbuzných sekvencií, alebo je možné ho vytvoriť z viacnásobného zarovnania. S jeho pomocou je potom prehľadaná zadaná databáza a nájdené potenciálne homológne sekvencie sú použité na vytvorenie novej databázy homológov. Z najlepšie ohodnotených je potom vytvorený v niekoľkých krokoch nový HMM model. Pri ohodnotení sa opäť využíva skóre *log-odds ratio*. Novo vytvorený HMM model je nakoniec použitý na opätovné prehľadanie databázy a získanie hľadaných homológnych sekvencií [31].

Na rozdiel od metódy HMMER, ktorá pri trénovaní modelov využíva metódu diskriminácie maxima (*maximum discrimination*), sa v algoritme SAM využíva metóda maximalizácie očakávania (*expectation maximization*). Pri tom sa celá trénovaná množina sekvencií zarovná k existujúcemu modelu a je generovaný nový model, ktorý kombinuje informácie zistené pri tomto zarovnaní. SAM využíva pri práci s modelom *Forward-Backward* algoritmus [31].

3.7 COMPASS

Comparison of Multiple Protein Alignments with Assessment of Statistical Significance (COMPASS) je metóda na detekcie vzdialených homológnych sekvencií a vytváranie lokálnych zarovnaní založených na porovnávaní profilov vytvorených z viacnásobných zarovnaní sekvencií. Metóda vytvára HMM profily zo zadaných viacnásobných zarovnaní, ktoré medzi sebou zarovnáva a analyticky určuje hodnoty E-value pre nájdené podobnosti [32].

Prvá fáza metódy pozostáva z konštrukcie profilov zo zarovnania. Je nutné vyrovnať sa s redundanciou medzi podobnými sekvenciami, to sa robí pomocou metódy výpočtu pozične špecifických nezávislých počtov (*PSIC* [33]). Pri vytváraní lokálneho zarovnania sú vypúšťané stĺpce s vysokým obsahom medzier, ktoré spôsobujú problémy. Skórovací systém pre ohodnotenie zhôd dvoch vytvorených profilov je založený na výpočte skóre *log-odds ratio*. Zmena mierky u získaných pozičných skór umožňuje nájdenie dlhších lokálnych zarovnaní.

V ďalšej fáze dochádza ku konštrukcii samotného lokálneho zarovnania pomocou algoritmu Smith-Waterman. Štatistická významnosť zarovnania je ohodnotená hodnotou E-value [34].

3.8 PROCAIN

PROCAIN je metóda porovnávania založená na viacnásobnom zarovnaní sekvencií, ktorá využíva kombináciu vertikálneho a horizontálneho kontextu zarovnania. Patrí do metód založených na porovnávaní profilov. Pri vertikálnom kontexte sa uvažujú obmedzenia substitúcií na určitých pozíciách v sekvencii a pri horizontálnom kontexte sa berú do úvahy vzory zložené z rezíduí na viacerých pozíciách. Hoci je PROCAIN založený na jednoduchej metóde porovnávania profilov a jednoduchého merania podobnosti vzorov pre horizontálne zarovnania, jeho výsledky pri hľadaní homológnych sekvencií sú porovnateľné s metódami založenými na HMM a predikcii sekundárnej štruktúry [35].

Vstupom metódy je viacnásobné zarovnanie. PROCAIN prevádza hľadanie v profilovej databáze, vytvára zarovnanie profilov a vracia významné podobnosti. Zarovnanie profilov je založené na skóre podobnosti medzi pozíciami porovnávaných viacnásobných zarovnaní sekvencií. Výpočet skóre zahrňuje štandardný výpočet skóre ako u metódy COMPASS kombinovaný s ďalšími hodnotami, ktoré reflektujú lokálnu podobnosť v sekundárnej štruktúre konzerváciou aminokyselín a sekvenčných motívov. Výsledné pozičné skóre je použité pre konštrukciu optimálneho lokálneho zarovnania dvoch profilov pomocou algoritmu Smith-Waterman.

3.9 HHsearch

HMM-HMM Search (HHsearch) je metóda, ktorá namiesto porovnania profilov (často používané), pracuje s porovnaniami HMM profilov – pri ktorom sa vyhodnocuje sekundárna štruktúra proteínov. Na rozdiel od ostatných metód sa nespolieha len na hodnotu E-value, pre ktorú bolo dokázané, že môže byť veľmi nespoľahlivá [36], ale používa aj pravdepodobnosť, že každá zhoda bude správne určená (true positive). Táto pravdepodobnosť je založená na rozdelení skóre pre negatívne a homológne doménové páry. Tieto skóre boli získané pri porovnaní každého zarovnania s každým z databázy SCOP, teda vychádzajú z reálnych dát.

Pri určovaní homológnych párov sa považuje za rozhodujúce, či sú z rovnakej nadrodiny alebo majú aspoň štrukturálne skóre MAXSUB [37] väčšie ako 0,1. Pri výpočte sa pre dopytovanú sekvenciu alebo zarovnanie predpovedá sekundárna štruktúra (α-helix, β-skladaný list a β-mostík) s dôveryhodnosťou v intervale 0 až 9. Na to sa používa program HHpred.

HHsearch podporuje lokálne aj semi-globálne zarovnanie.

3.10 HHblits

HMM-HMM based lightning fasta iterative sequence search (HHblits) je metóda založená na metóde HHsearch, ktorú rozširuje o iteratívne sekvenčné vyhľadávanie. Je rýchlejšia, pretože využíva predfiltrovanie zarovnania profilov, čím redukuje počet kompletných HMM-HMM zarovnaní rádovo na tisíce. Vďaka tomu je rýchlejšia než PSI-BLAST a citlivá ako HHsearch [38].

Použitá sekvenčná databáza je rozdelená na skupiny sekvencií, ktoré sú zarovnané skoro celou svojou dĺžkou. Pre tieto skupiny sú vygenerované viacnásobné zarovnania sekvencií a HMM profily. Ďalším krokom je potom prevedenie dopytovanej sekvencie alebo viacnásobného zarovnania na HMM profil. Nasleduje hľadanie v databáze HMM profilov. Z nájdených HMM profilov, ktoré majú hodnotu E-value nižšiu než zadaný prah, sa vytvára spoločne s dopytovaným HMM profilom nové (viacnásobné) zarovnanie. Z výsledného zarovnania je zostrojený znova HMM profil, ktorý sa použije v ďalšej iterácii metódy.

Základom rýchleho spracovania je predspracovanie, ktoré sa prevádza tak, že každý stĺpec HMM profilu je diskretizovaný na jeden znak z abecedy 219 znakov, kde každý znak reprezentuje jeden typický profil stĺpca. Týmto spôsobom vznikne rozšírený sekvenčný profil. Predfiltrovanie sa potom skladá z dvoch krokov: prvým je lokálne zarovnanie bez medzier medzi rozšíreným profilom z dopytu a rozšírenými profilmi z databázy a druhým je lokálne zarovnanie s medzerami.

3.11 Metodika testovania nástrojov pre vyhľadávane homológov

V mnohých publikáciách sa autori zameriavajú na porovnanie metód pre hľadanie homológov. Základom pre toto porovnanie je výber štatisticky odpovedajúceho zastúpenie vzorkou dát (proteínov) z celkovej sady dát (databázy), ktorá je použitá ako vstup pre jednotlivé metódy. Táto vzorka by mala obsahovať odpovedajúce zastúpenie proteínov podľa dĺžky sekvencie, sekvenčnej podobnosti a ďalších kritérií.

Dôležitým krokom je určenie skutočnej príbuznosti medzi proteínmi. S pomocou tejto znalosti je možné ohodnotiť výstup metódy pre vyhľadanie homológov. Podľa tohto ohodnotenie potom môžu

byť metódy porovnané. Určenie skutočnej homológie medzi proteínmi môže prebiehať napríklad na základe informácií získaných z existujúcich klasifikácií proteínov (teda založené na referenčnom modeli). Ďalšou možnosťou je určenie homológie na základe výpočtu použitia jedného zo známych algoritmov pre výpočet štruktúrnej podobnosti (budú ďalej v podkapitole 4.2.2), ktorých výsledkom je skóre, podľa ktorého je možné s určitou presnosťou určiť príbuznosť nájdených párov. Tento prístup je teda nezávislý na referenčnom modeli.

Metódy sú teda porovnávané podľa ich úspešnosti nájsť skutočne homológne páry, ktoré sú určované podľa niektorého z prístupov popísaných vyššie. Základom je vždy určenie štruktúrnej podobnosti, ktorá je jedným z hlavných ukazovateľov homológie proteínov.

4 Návrh nástroja na vyhľadávanie homológnych sekvencií proteínov

V súčasnej dobe existuje množstvo metód pre vyhľadávanie homológnych sekvencií proteínov. Každá ale pracuje na trocha inom princípe a dáva hodnotné výsledky len pre istú špecifickú oblasť dát. Súčasné metódy pracujú predovšetkým s nastavením parametra maximálnej E-value, ktorou je obmedzený počet nájdených uvažovaných homológnych párov. So zväčšujúcou maximálnou hodnotou pribúda počet chybne označených párov. Pre užívateľa je zložité určiť, ako má tento parameter nastaviť. Táto hodnota je navyše závislá na veľkosti databázy používanej pri hľadaní.

Metódy sa líšia aj tom, na vyhľadávanie akej príbuznosti sú cielené. Základné nástroje ako BLASTp alebo PHMMER vyhľadávajú hlavne blízkych homológov, iné metódy cielia aj na homológne proteíny so vzdialenejšou príbuznosťou. K týmto patria hlavne tie iteratívne. Pri nich patrí medzi hlavné parametre, ktoré je nutné určiť, predovšetkým počet iterácií, ktoré môžu previesť. Zatiaľ čo pre hľadanie párov s blízkou príbuznosťou by mal stačiť menší počet iterácií, pre nájdenie vzdialenejších homológov je treba iterácií viacej.

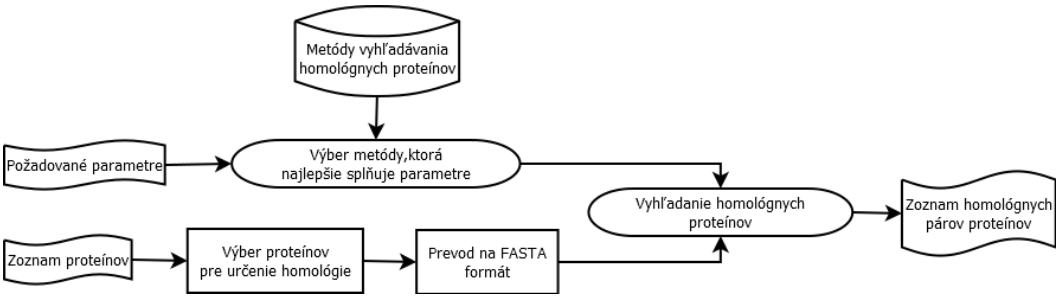
Keďže porovnanie metód na vyhľadávanie homológov bolo vo väčšine prác spracované iba z globálneho pohľadu a nie jednotlivo podľa sekvenčnej identity nájdených párov, hlavným cieľom tejto práce bolo vytvoriť aplikáciu, ktorá bude na základe výsledkov porovnania metód podľa úspešnosti pri hľadaní párov s danou sekvenčnou identitou pomáhať užívateľovi vybrať správnu metódu a jej nastavenie. Aplikácia dokáže metódu aj spustiť a zobraziť jej výstup.

4.1 Princíp činnosti nástroja

Princíp činnosti aplikácie je zobrazený na obrázku 4.1. Ako vstup aplikácie je uvažovaný zoznam proteínov (alebo domén), z ktorých užívateľ vyberie tie, pre ktoré chce nájsť homológne proteíny. Tieto sekvencie proteínov na vstupe môžu mať rôzny formát. Ja som v tejto práci použila dva asi najrozšírenejšie, FASTA a PDB. Aplikácia potom sekvencie proteínov prevedie na jeden spoločný formát FASTA, ktorý bol vybraný z dôvodu, že ho podporujú všetky zvolené metódy.

Potom nasleduje výber optimálnej metódy pre hľadanie homológov na základe toho, akú užívateľ vyberie oblasť sekvenčnej identity, ktorá ho najviac zaujíma a zvolí mieru maximálnej chybovosti. Toto sa prevádza na základe spracovanej analýzy úspešnosti metód, ktorá je popísaná v nasledujúcej podkapitole.

Po zistení optimálnej metódy spolu s nastavením vhodných vstupných parametrov nasleduje spustenie externého nástroja a zobrazenie výsledných nájdených homológnych sekvencií, ktoré boli nájdené k zadanej sekvencii.



Obrázok 4.1. – Princíp činnosti aplikácie

4.2 Posúdenie úspešnosti zvolených metód

Pre porovnanie rozdielnych metód založených na rôznych typoch vyhľadávania bolo nutné stanoviť metodiku, ktorá jasne určí, ktorá metóda je pre zadané parametre najvhodnejšia. K tomu bolo nutné ako prvé zvoliť testovaciu sadu proteínov, nad ktorými bude toto posudzovanie prebiehať. Jej získanie je popísané v podkapitole 4.2.1.

Keďže nás zaujíma aj rozloženie úspešnosti metód podľa sekvenčnej identity proteínov, bolo nutné ju pre každú dvojicu nájsť. V tejto práci som využila sekvenčnú identitu, ktorá bola získaná až po zarovnaní dvojice podľa sekundárnej štruktúry. Tento typ viac odpovedá realite [39], jej význam bol spomenutý aj v podkapitole 3.11. Ďalej už budem túto identitu označovať ako sekvenčnú identitu po štruktúrnom zarovnaní. Popis získavania zarovnania je uvedený v podkapitole 4.2.2.

Ďalším krokom bolo určenie, ktoré zo všetkých možných získaných dvojíc proteínov z tejto sady bude možné považovať za skutočne homológne a tieto páry zaradiť na referenčný zoznam overených homológov. Tak isto bolo nutné získať referenčný zoznam tých dvojíc, pre ktoré je overené, že homológmi nie sú. Pre konštrukciu týchto zoznamov bola využitá hierarchická klasifikácia SCOPe. Postup vytvorenia referenčných zoznamov bude opísaný v podkapitole 4.2.5.

Takto vytvorená sada potom môže byť použitá na klasifikáciu metód a je viac popísaná v podkapitole 4.2.7.

4.2.1 Tvorba dátovej sady

Podmienkou pre dobré otestovanie metód je zvolenie vhodnej dátovej sady proteínov, ktoré budú slúžiť ako vstup pre hľadanie homológnych proteínov. Táto vzorka by mala mať zastúpenie proteínov podľa rôznych kritérií (dĺžka sekvencie, zastúpenie proteínov s danou sekvenčnou identitou apod.) odpovedajúcich zastúpeniu proteínov v prehľadávanej databáze. Rovnako platí aj to, že čím väčšia bude vzorka dát, tým budú výsledky presnejšie.

Pre výber vzorky dát som sa rozhodla využiť niektorú z už existujúcich sad proteínov, ktoré sú klasifikované do skupín podľa ich príbuznosti. Existuje celá rada manuálne alebo automaticky vytvorených klasifikácií proteínov (alebo proteínových domén) do skupín podľa podobnosti

a príbuznosti (PFAM, CATH, SCOP). Keďže sa v tejto práci zameriavam na zisťovanie homológnych párov na úrovni super-rodín, teda aj vzdialenejšej príbuznosti, zvolila som posledne menovanú databázu SCOP a to konkrétne jej rozšírenú verziu 2.05. Celá databáza SCOPe obsahovala v čase vzniku tejto práce 71 015 záznamov (proteínových domén), ktoré sú na najnižšej úrovni zoradené do rodín a super-rodín (viď kapitola 3.).

Množstvo domén z tohto počtu má medzi sebou vysokú sekvenčnou identitu. Pri vytváraní testovacej sady zo všetkých týchto domén by dochádzalo k zbytočnému vyhľadávaniu takmer rovnakých domén. Preto som sa rozhodla zobrať len domény, ktoré sú medzi sebou maximálne z 95% sekvenčne identické, čím v zozname zostalo len 26 949 domén. Ide o podstatný rozdiel, keďže bude nutné namiesto 2 521 529 605 dvojíc, ktoré by vznikli porovnaním každej zo 71-tisíc domén so všetkými ostatnými, určiť podobnosť len u 363 110 826 dvojíc. Hoci potrebujeme výsledky hľadania rozdeliť podľa štruktúrnej identity, týmto obmedzením stratíme časť najpodobnejších párov, ale z dôvodu obrovského nárastu výpočtového času pri zahrnutí všetkých domén sa s touto stratou musíme zmieriť.

4.2.2 Štruktúrne zarovnanie

Na získanie čo najpresnejšieho štruktúrneho zarovnania medzi každým párom proteínov z testovacej sady som využila nástroj TM-align [39]. Jeho algoritmus je založený na použití kombinácie rotačnej matice TM-skóre a dynamického programovania. Oproti starším nástrojom ako sú CE, DALI a SAL je doba konštrukcie zarovnania polovičná až štvrtinová [39].

Nástroj TM-align sa okrem konštrukcie štruktúrneho zarovnania využíva aj na získanie TM–skóre a jeho hodnota popisuje podobnosť dvoch proteínov. TM–skóre je viac citlivé na globálnu topológiu než na lokálne štruktúrne variácie. Oproti iným hodnotám (Z-skóre, RMSD) popisujúcim podobnosť proteínov je normalizovaná tak, že veľkosť skóre nie je závislá na veľkosti proteínu. Pre určenie, či je podobnosť náhodná, alebo je skutočne znakom homológie, je prahová hodnota rovná 0,17. V mojej práci som sa spoľahla na určenie homológnych párov na základe rozdelenia do super-rodín a rodín v hierarchii SCOP a využila som len vyprodukované štruktúrne zarovnanie.

TM-align využíva len súradnice chrbtových C_α atómov zo zadanej proteínovej štruktúry, dá sa však zovšeobecniť na akýkoľvek typ atómu. Pri zarovnaní využíva tri druhy rýchlej identifikácie počiatočného zarovnania:

- Prvý typ vytvára zarovnanie podľa sekundárnej štruktúry dvoch proteínov pomocou dynamického programovania, využíva aj medzery a jednoduché porovnanie (rovnaké – nerovnaké) dvoch sekundárnych štruktúr.
- Druhý typ porovnáva dve štruktúry bez vkladania medzier, vyberá sa zarovnanie s najvyšším TM-skóre.

- Tretí typ je opäť získaný dynamickým programovaním s využitím vkladania medzier, ale skórovacia matica už neobsahuje len dva typy hodnôt, ale kombinuje pol na pol skórovaciu maticu zo štruktúrneho zarovnania (z prvého typu) a maticu vzdialeností získanú pri druhom type zarovnania.

TM-align potrebuje na vstupe súbory obsahujúce popis porovnávaných proteínov vo formáte PDB [11]. Po získaní dokončeného štruktúrneho zarovnania je možné získať viacej vypovedajúcu hodnotu sekvenčnej identity než by bola tá, ktorá by bola získaná priamo porovnaním dvoch sekvencií.

4.2.3 Požiadavky na metódy vyhľadávajúce homológne proteíny a ich parametre

Z pohľadu užívateľa, ktorý sa chce o zvolenom proteíne (alebo doméne) dozvedieť viac pomocou toho, že k nemu nájde homológne proteíny, sú pre vyhľadávajúce metódy najpodstatnejšie tieto ukazovatele:

- **počet nájdených homológnych párov z celkového počtu** – metódy sa líšia v tom, ako sú schopné hľadať aj vzdialenejšie príbuzné proteíny,
- **pomer správne a chybne označených homológov** – existuje mnoho rôznych ukazovateľov úspešnosti pri určovaní homológie, mnou používané ukazovatele *error rate* a *FPR* budú popísané v nasledujúcich častiach práce,
- **čas hľadania,**
- **dostupnosť naformátovaných databáz** – metódy často potrebujú špeciálne naformátované databázy, niektoré dokonca poskytujú nástroje pre vytvorenie vlastnej databázy s pomocou sady proteínov alebo ich zarovnaní.

To, aké výsledky metóda pre zadaný vstup vracia, je možné nastaviť pomocou týchto základných parametrov:

- **počet iterácií metódy** – u iteratívnych metód, často ide o maximálny možný počet, ak metóda nedokonverguje k výsledku skôr, než dosiahne zadaný počet iterácií,
- **E-value –** maximálna hodnota E-value, ktorú môže mať pár vrátený metódou,
- **prahová E-value –** u iteratívnych metód značí maximálnu E-value, ktorú môžu mať páry na konci jednej iterácie, ktoré budú zahrnuté do tvorby PSSM matice používanej v nasledujúcej iterácii,
- **pokrytie** – určuje v percentách minimálny pomer zarovnanej časti proteínov voči najdlhšiemu proteínu v zarovnaní,
- **cena za otvorenie medzery** – tam, kde sa dopytovaná sekvencia a nájdená sekvencia nezhodujú v žiadnej z aminokyselín, je často vložená medzera, týmto parametrom je možné ovplyvniť, ako odvážne budú metódy pri rozhodovaní o otváraní medzier,

- **cena za predĺženie medzery** – spravidla menšia ako cena za otvorenie medzery, pretože dĺžka medzery nie je až taká podstatná, dôležité je, že medzera vôbec vznikla,
- **typ použitej skórovacej matice** – väčšina metód používa ako defaultnú maticu BLOSUM62.

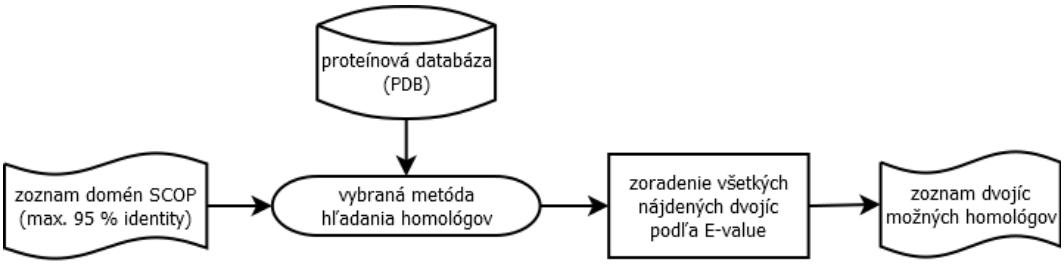
4.2.4 Postup pri vyhľadávaní možných homológnych proteínových párov

Pre spustenie vybranej metódy je nutné vybrať vstup – zoznam sekvencií proteínov, ku ktorým chceme vyhľadať homológne proteíny. V tejto práci som sa zamerala na hľadanie s pomocou sekvencií proteínov v sekvenčných databázach. Tieto databázy musia odpovedať typu sekvencií na vstupe. Ak je na vstupe podmnožina celej hierarchie SCOP, mala by sekvenčná databáza obsahovať minimálne tú istú podmnožinu sekvencií. V mojom prípade som použila databázu Protein Data Bank (PDB), ktorá bola dostupná pre väčšinu testovaných metód a ktorej podmnožinou je databáza SCOP. Teda domény z databázy SCOP sa vyhľadávajú vo viac doménovej databáze z PDB. Ak metóda našla niektorú z domén, ktorá je v PDB, ale nie je v použitom výbere domén SCOP s najviac 95% sekvenčnou identitou, bola odfiltrovaná preč.

Po spustení vybranej metódy sú na výstup vytlačené dvojice možných homológov spolu s niektorými ukazovateľmi, prípadne aj s ich zarovnaním. Najzaujímavejšou z môjho pohľadu bola asi hodnota E-value. Pre zhodnotenie úspešnosti metódy pre hľadanie homológov a vykreslenie ROC krivky (viď ďalšia podkapitola) je nutné tieto dvojice zoradiť práve podľa E-value. Potom dôjde k tomu, že páry, pri ktorých je najväčšia pravdepodobnosť, že budú skutočne homológne, budú na začiatku zoznamu.

Namiesto E-value by bolo možné použiť aj iné ukazovatele, napríklad takmer každá metóda udáva aj svoje špeciálne skóre, podľa ktorého by bolo možné zoznam zoradiť, ale v rámci porovnaní rôznych metód je hodnota E-value asi najviac nezávislá na type metódy.

Na obrázku 4.2. je znázornený diagram postupu pri hľadaní homológnych sekvencií použitý v tejto práci.



Obrázok 4.2. – Diagram postupu hľadania homológnych domén.

4.2.5 Postup vyhodnocovania výsledkov metód

Ak použijeme niektorú z metód pre hľadanie homológnych proteínov a podarí sa získať dvojice, o ktorých metóda tvrdí, že sú homológne, je potrebné nejakým spôsobom overiť, či ide skutočne o pravých homológov. Niekedy sa pre určenie, či je pár proteínov skutočne homológny alebo nie, používa hodnota E-value, ktorú metódy spravidla udávajú pre každú nájdenú dvojicu. Spôsob určenie, či podľa hodnoty E-value ide o homológny pár, je uvedený v podkapitole 3.1.1.

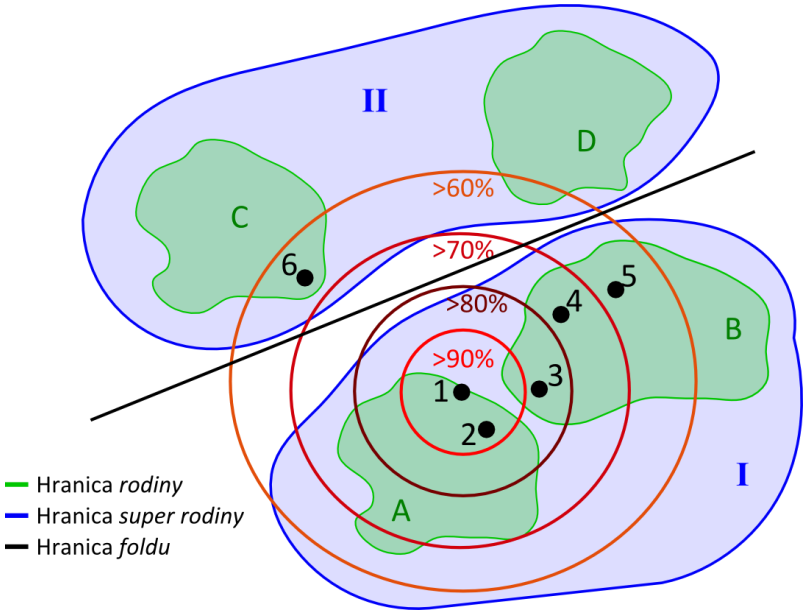
V literatúre je možné nájsť aj iné spôsoby, ktoré sa na overenie homológie používajú. Základom je výpočet rôznych typov skóre, ako napríklad skóre globálnej vzdialenosti (Global Distance Test Total Score – GDT-TS) [40], výpočet Z-skóre v prípade nástroja DALI alebo pomocou TM-skóre získaného nástrojom TM-align (viď 4.2.2).

V tejto práci som sa spoľahla na určenie skutočnej homológie podľa rozdelenie v hierarchii SCOP. Za skutočne homológne považujem tie domény, ktoré patria do rovnakej super-rodiny. Z toho vyplýva, že aj domény, ktoré sú z iných rodín, ale v rovnakej super-rodine, považujem za homológne. Naopak domény, ktoré neležia v rovnakom *folde*, sú označené ako nehomológne. Ich prípadná podobnosť je s najvyššou pravdepodobnosťou len dielom náhody. V prípade zvyšných domén, ktoré síce ležia v rovnakom *folde*, ale v rôznych super-rodinách, nie je homológia jasná. Preto ich v mojom porovnaní nebudem uvažovať. Ak niektorá metóda určí, že dve domény z tejto nejasnej oblasti sú homológne, nebudem už túto dvojicu uvažovať v porovnaní, aby som sa nedopustila chyby pri určovaní skutočnej homológie. To isté platí aj v prípade, keď metóda dvojicu zaradí medzi nehomológne. Tento postup je bežne používaný z dôvodu, že v tejto oblasti je problémové príbuznosť s istotou určiť [40]. A využíva sa pre hľadanie vzdialených homológnych párov (samozrejme vrátane blízkych) a je používaný vtedy, ak nie je špecifikovaná konkrétna oblasť sekvenčnej identity, kde sa má vyhľadávať. Princíp hľadania podľa sekvenčnej identity bude popísaný ďalej.

Hoci je rozdelenie domén v hierarchii SCOP, ktoré bolo prevádzané manuálne, veľmi presné, bolo dokázané, že existujú aj prípady homológnych domén, ktoré ležia v rôznych *foldoch* [40]. K tým patrí napríklad *Rossman doubly-wound sandwich* foldy, *Imunoglobin-like β-sandwich* a *β-propeller* foldy. Hoci existujú tieto výnimky, stále je toto manuálne spravované rozdelenie proteínov veľmi spoľahlivé, a preto som ho využila.

Pri vyhodnocovaní výsledkov sa chcem zamerať na schopnosť metódy určiť homológne páry so zadanou minimálnou sekvenčnou identitou (po štruktúrnom zarovnaní). Táto štruktúrna podobnosť nám pomôže odlíšiť typ príbuznosti, ktorú hľadáme. Ak sa zameriame na najvyššiu štruktúrnu podobnosť, mali by sme dostať najvyššie percento párov tvorených tými najbližšie príbuznými doménami, predovšetkým patriacimi do rovnakej rodiny. Naopak, ak nastavíme minimálny prah, teda sekvenčnú identitu od 0 %, môžeme získať aj tie najvzdialenejšie homológne páry, samozrejme sa tým podstatne zväčší počet chybne nájdených homológnych párov. Toto je znázornené na obrázku

4.3, kde sú domény znázornené na mape takým spôsobom, aby ich vzdialenosť zodpovedala ich sekvenčnej identite po štruktúrnom zarovnaní.



Obrázok 4.3. – Znázornenie domén v priestore podľa ich sekvenčnej identity.

Na obrázku sú znázornené farebnými kružnicami, so stredom v doméne 1, hranice oblastí, ktoré obsahujú domény so sekvenčnou identitou voči doméne 1 minimálne takou, aká je uvedená pri nich. Teda je zrejmé, že napríklad domény 2 a 3 sú voči doméne 1 minimálne z 80 % sekvenčne identické.

Z obrázku je možné odvodiť, že ak zadáme prah minimálnej sekvenčnej identity na 90% vyhľadáme len najbližšie dvojice, ako je tu pár skladajúci sa z domén 1 a 2, pripadne domény 4 a 5 atď. Je zrejmé, že väčšina domén z týchto dvojíc budú pochádzať z rovnakej rodiny. Ak zadáme prah na úrovni 60 %, síce sa nám môže podariť nájsť skutočné homológne páry s doménou 1 ako dvojice s doménami 2, 3, 4 a 5, súčasne sa ale môže stať, že pre doménu 1 nájdeme do páru doménu 6. Tento pár bude, ale označený ako nesprávne určený homológny pár, pretože domény 1 a 6 nepatria do rovnakého *foldu*. Nastavovaním tohto prahu (veľkosti tejto oblasti) je teda možné ovplyvniť, ako veľmi vzdialené príbuzné páry budú nachádzané.

Pre znázornenie úspešnosti jednotlivých metód som sa rozhodla zostaviť pre každú testovanú metódu a jej nastavenie takzvanú ROC krivku, ktorá je popísaná ďalej.

4.2.6 ROC krivka

Receiver Operating Characteristic (ROC) krivka je grafické znázornenie ilustrujúce schopnosť binárneho klasifikačného systému správne zaradiť vstupné hodnoty do tried [41]. Za takýto klasifikačný systém môžeme považovať akúkoľvek z metód pre hľadanie homológov.

Počas binárnej klasifikácie rozdeľujeme vstup do dvoch tried, kde výstup je buď pozitívny (*positive* **P**) alebo negatívny (*negative* **N**). Z binárnej klasifikácie potom máme štyri možné výstupy:

- Ak je výstup predikcie pozitívny:
  - a hodnota je v skutočnosti naozaj pozitívna, ide o výsledok *true positive* (**TP**).
  - a hodnota je v skutočnosti negatívna ide o *false positive* (**FP**).
- Ak je výstup predikcie negatívny:
  - a hodnota je v skutočnosti naozaj negatívna, ide o výsledok *true negative* (**TN**).
  - a hodnota je v skutočnosti pozitívna ide o *false negative* (**FN**).

ROC krivka je vytvorená vynesením pomeru hodnoty *true positive rate* párov voči hodnote *false positive rate*.

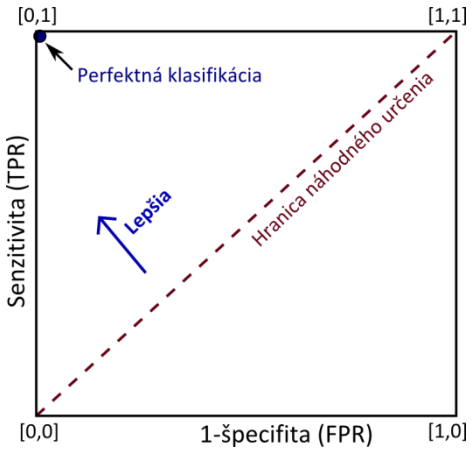
Hodnota *true positive rate* (**TPR**) definuje, koľko správnych výsledkov sa našlo zo všetkých možných správnych výsledkov počas vyhodnocovania. Označujeme ju aj výrazom senzitivita, alebo *hit rate* a vyjadruje sa ako:

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}$$

Hodnota false positive rate (**FPR**) definuje, koľko sa vyskytlo nesprávne pozitívnych výsledkov medzi všetkými možnými negatívnymi výsledkami počas vyhodnocovania. FPR označujeme aj ako *fall-out* alebo doplnok špecificity a vyjadruje sa takto:

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} = fall_out = 1 - \check{s}pecificita$$

ROC priestor je definovaný pomocou hodnôt FPR na ose x a hodnôt TPR na ose y a znázorňuje kompromis medzi ziskami (TP) a stratami (FP). Každý výsledok predikcie reprezentuje jeden bod v ROC priestore. ROC priestor je zobrazený na obrázku 4.4.



Obrázok 4.4. – ROC priestor (ROC space).

Najlepší možný výsledok sa zobrazuje v najľavejšom hornom rohu ROC priestoru (súradnice [0,1]) a odpovedá 100 % senzitivite a 100 % špecificite. ROC priestor rozdeľuje uhlopriečka, ktorá označuje hranicu, kde sú výsledky klasifikačnej metódy lepšie, než keby sme náhodne vyberali, ktorý vstup zaradíme do ktorej triedy, a kde horšie. Najlepšiu klasifikáciu predstavujú body, ktoré sú v definovanom priestore najďalej od tejto uhlopriečky. O bodoch, ktoré ležia na rovnakej priamke, ktorá tvorí rovnobežku s uhlopriečkou, môžeme povedať, že predstavujú rovnako dobré výsledky

klasifikácie. Tieto pravidlá je možné aplikovať aj na priestor pod uhlopriečkou s tým, že je nutné výsledok klasifikácie pred použitím znegovať.

V mojej práci sa budú pre hodnotenie metód uvádzať najmä grafy, kde na ose x bude pomer nájdených FP párov vzhľadom k celkovému počtu FP párov v testovanom vzorku dát a na ose y pomer nájdených TP párov vzhľadom k celkovému počtu TP párov. Oba pomery sú zobrazované v percentách. Krivky v tomto typu grafu sú v literatúre taktiež často označované ako ROC krivky.

Pre určenie úspešnosti metódy podľa pomeru TP a FP párov je použitá miera chybovosti, ktorú budem označovať ako *error rate* a ktorej výpočet je vyjadrený vzorcom (2) a je udávaná v percentách.

$$error\ rate = \frac{\frac{počet\ FP}{počet\ položiek\ v\ referenčnom\ zozname\ FP\ \ párov}}{\frac{počet\ TP}{počet\ položiek\ v\ referenčnom\ zozname\ TP\ \ párov}} * 100 \qquad (2)$$

Krivky zobrazujúce vybrané hodnoty tejto miery sú pridané do grafu s krivkami jednotlivých metód, aby bolo možné ich úspešnosť jednoducho vizuálne porovnať.

4.2.7 Klasifikácia výsledkov metódy

Výstupom spracovávanej metódy sú dvojice, ktoré táto metóda označila ako homológne. Naopak o tých dvojiciach, ktoré metóda nenašla (nevrátila ako výstup), môžeme povedať, že ich metóda označila za nehomológne, alebo že ich nevie zaradiť. Na základe referenčných zoznamov dvojíc zostrojených podľa hierarchie SCOP, v ktorých sú dvojice podľa nás určite homológnych, určite nehomológnych a tých dvojíc, pri ktorých to nevieme určiť, môžeme každú dvojicu zaradiť do jednej z nasledujúcich piatich skupín:

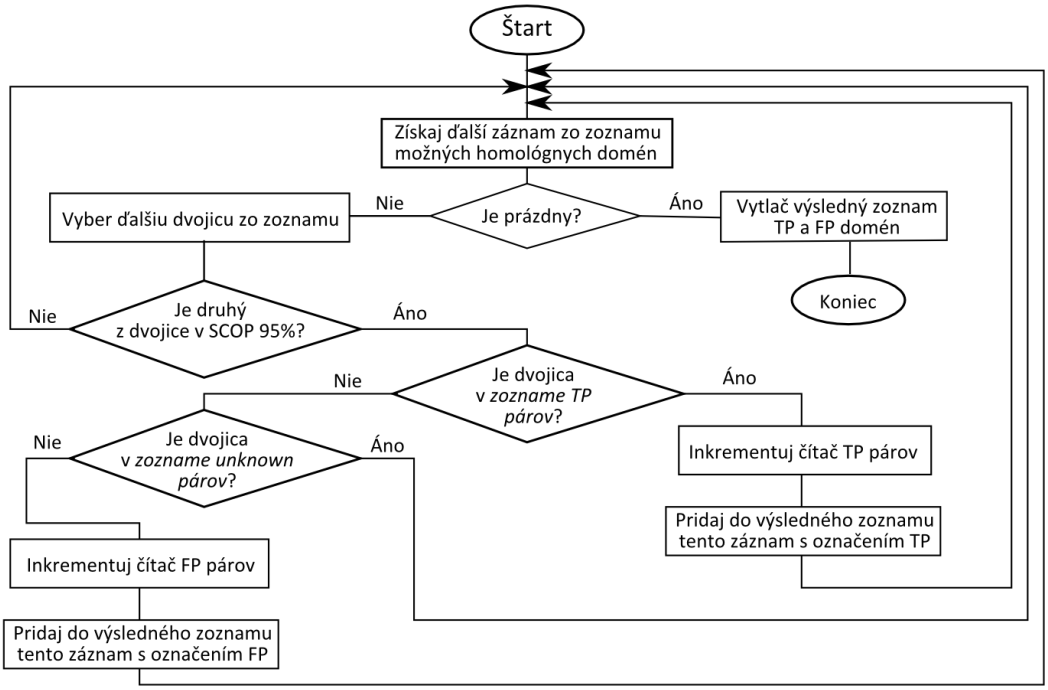
- ***true positive*** – (ďalej označované len ako *zoznam TP párov*) metóda označila dvojicu ako homológnu a súčasne je v našom zozname určite homológnych párov,
- ***false positive*** – metóda síce označila dvojicu ako homológnu, ale táto nie je ani v *zozname TP párov* ani v zozname párov, pre ktoré nevieme o homológii rozhodnúť (ďalej označované len ako *zoznam UN párov*),
- ***true negative*** – metóda túto dvojicu nenašla a súčasne nie je ani v *zozname TP párov* ani v *zozname UN párov,*
- ***false negative*** – metóda túto dvojicu nenašla a súčasne je v *zozname TP párov,*
- ***unknown*** – všetky ostatné prípady (dvojica je v *zozname UN párov*).

Spôsob vyhodnotenia by mal byť zrejmý z obrázku 4.5., na ktorom je znázornené určovanie tried, do ktorých spadajú výsledky získané z metód pre hľadanie homológnych párov s využitím referenčných zoznamov získaných z klasifikácie SCOP.

		Referenčné rozdelenie		
	Celkový počet	Pozitívne (homológny pár)	Neznáme (nedá sa s istotou určiť)	Negatívne (nehomológny pár)
Výstup metódy	Pozitívny (nájdený ako homológny pár)	True positive (TP)	Unknown	False positive (FP)
	Negatívny (nenájdený pár)	False negative (FN)	Unknown	True negative (TN)
		True positive rate (senzitivita) $= \frac{TP}{TP+FN}$		False positive rate (1-špecificita) $= \frac{FP}{FP+TN}$

Obrázok 4.5. – Klasifikácia výsledkov do tried.

Postup určovania skupiny, ktorý sa prevádza pre každú nájdenú možnú dvojicu homológnych domén je znázornený na obrázku 4.6. pomocou diagramu.



Obrázok 4.6. – Diagram procesu vyhodnotenia výsledkov metód pre hľadanie homológnych proteínov.

Vstupom tohto procesu je zoznam nájdených párov, ktoré vybraná metóda určila ako homológne (viď obrázok 4.2). Ak druhá doména z nájdeného páru nie je súčasťou sady SCOP 95%, je tento pár zahodený a nie je ďalej skúmaný. Prvú doménu nie je nutné kontrolovať preto, že je to tá, ktorá bola použitá pre vyhľadávanie danou metódou a je vyberaná zo sady SCOP 95%. Pri nájdení páru v referenčnom zozname TP párov (respektíve FP párov), je okrem jeho označenia ako TP (respektíve FP), získaná aj jeho sekvenčná identita po štruktúrnom zarovnaní, ktorá je pred-počítaná pre každý pár v referenčných zoznamoch. Táto identita bude neskôr využitá pri určovaní úspešnosti vyhľadávania metódy pre zadanú oblasť identity.

Výstupom tohto procesu prevádzaného pre každú metódu a všetky vybrané hodnoty jej nastavení je takzvaný ROC súbor. Tento ROC súbor obsahuje všetky nájdené TP a FP dvojice danou metódou, ktoré sú umiestnené vždy na samostatnom riadku, spolu s názvami domén, nájdenou sekvenčnou identitou a dopredu vyrátaným pomerom *error rate*. Riadky v tomto súbore sú zoradené podľa hodnôt E-value. Tento ROC súbor slúži pre vykreslenie ROC kriviek a tiež pre určovanie najlepšieho nastavenia metódy (viď 4.3.2).

Ako bolo uvedené vyššie, z ROC súborov sa zisťujú vhodné nastavenia metódy vrátane nastavenia maximálnej hodnoty E-value. Táto hodnota je lineárne závislá na veľkosti priestoru pre vyhľadávanie, teda je závislá na veľkosti prehľadávanej databázy. V tejto práci bola použitá sekvenčná databáza PDB (vo väčšine prípadov). V prípade použitia inak veľkej databázy (z pohľadu počtu záznamov) bude nutné na základe vzorca (1) v kapitole 3.1.1. túto hodnotu vynásobiť konštantou, ktorá udáva pomer medzi počtom záznamov vo vybranej databáze vzhľadom k počtu záznamov v databáze použitej v tejto práci (teda PDB).

4.2.8 Klasifikácia výsledkov metódy pre zadanú oblasť sekvenčnej identity

V predchádzajúcej kapitole bol popísaný postup, ktorý bol použitý pre vyhodnotenie výsledkov všetkých metód a kde nebolo uvažované žiadne obmedzenie na sekvenčnú identitu.

Cieľom tejto práce bolo aj získať dáta popisujúce úspešnosť metód podľa zadanej oblasti sekvenčnej identity (teda ROC súbory pre každú oblasť identity). Základom tohto postupu je získanie ROC súboru pre každú metódu a nastavenie (viď 4.2.7) a jeho filtrácia podľa sekvenčnej identity, ktorá je uvedená pri každej dvojici domén z tohto súboru. Tá prebiehala týmto spôsobom:

- **nájdený TP pár so sekvenčnou identitou rovným alebo väčším než zadaný prah** – pár označený ako TP a uložený do ROC súboru danej sekvenčnej oblasti,
- **nájdený TP pár so sekvenčnou identitou menšou ako zadaný prah** – pár označený ako FP a uložený do ROC súboru danej sekvenčnej oblasti,
- **nájdený FP pár** – pár označený ako FP a uložený do ROC súboru danej sekvenčnej oblasti.

Takto je možné vytvoriť ROC súbory s rovnakým formátom ako v prípade ROC súborov metód bez rozlíšenia sekvenčnej identity a použiť ich pre vykresľovanie grafov alebo hľadanie vhodnej metódy a jej nastavenia pre zadanú oblasť sekvenčnej identity.

4.3 Návrh a implementácia užívateľskej aplikácie

Návrh aplikácie, ktorá bola cieľom tejto práce, bol rozdelený do základných častí: načítanie zdrojových dát, výber optimálnej metódy vyhľadávania, nastavenie parametrov spúšťaných externých nástrojov či samostatné spúšťanie metód a zobrazenie výsledkov. Navyše bolo treba riešiť aj ukladanie základných parametrov pre spustenie externých nástrojov a užívateľských nastavení.

Aplikácia bola implementovaná v jazyku Java s využitím sady Java Development Kit 1.8. Vývoj prebiehal vo vývojovom prostredí NetBeans IDE 8.0.2. Aplikácia vyžaduje, aby boli na cieľovom PC nainštalované požadované nástroje pre hľadanie homológnych sekvencií (budú viac popísané v podkapitole 5.1) a potrebné databázy a tiež nástroj Gnuplot, ktorý je používaný na tvorbu grafov.

4.3.1 Načítanie zdrojových dát

Sekvencie proteínov je možné zadať pomocou súborov obsahujúcich sekvencie vo formáte FASTA alebo PDB. V aplikácii je možné načítať sekvencie súčasne z viacerých súborov. Keďže požadovaným formátom pre všetky nástroje pre hľadanie homológov je FASTA formát, bolo nutné vyriešiť prevod formátu PDB na tento formát.

Formát PDB bol popísaný v kapitole 3.1.3. Na prevod do formátu FASTA je možné využiť riadky začínajúce značkou DBREF, kde sa vyskytuje názov, pod ktorým je proteínová doména uvedená v databáze. Sekvencia domény je potom získaná z riadkov začínajúcich značkou SEQRES s odpovedajúcim označením. Táto sekvencia aminokyselín zapísaných trojznakovým kódom bola skonvertovaná na jednoznakové kódy aminokyselín a pridaná k už získanému názvu a tým vznikol odpovedajúci FASTA záznam.

Celý tento postup bol implementovaný do samostatného dialógového okna, kde je možné načítať zdrojové súbory, vybrať požadované sekvencie a zobraziť ich. Po ukončení aplikácie sú tieto otvorené súbory a vybrané proteíny uložené a po opätovnom otvorení aplikácie je možné ich použiť znova.

4.3.2 Výber optimálnej metódy vyhľadávania

Hlavnou úlohou aplikácie je pomôcť užívateľovi vybrať metódu pre vyhľadanie homológnych proteínov k tým proteínom, ktoré užívateľ zadá. Užívateľ môže špecifikovať tieto obmedzenia na výber metódy:

- **výber metódy** – užívateľ môže zadať konkrétnu metódu, na ktorej chce pracovať, aplikácia sa potom sústredí iba na výber najlepších parametrov tejto metódy. Medzi možné metódy patria tieto: BLAST, PSIBLAST, PHMMER, JACKHMMER,

CSBLAST, CSIBLAST, HHBLITS, SAM, COMPASS. Ďalšou možnosťou je nevybrať žiadnu konkrétnu metódu a toto nechať na samotnej aplikácii.

- **oblasť sekvenčnej identity** – je požadované, aby nájdená metóda dávala najlepšie výsledky pre páry s týmto rozsahom sekvenčnej identity (po štruktúrnom zarovnaní). Rozsah sekvenčnej identity od 0 do 100 % bol rozdelený po 10 % a bolo vytvorených desať oblastí (skupín): oblasť sekvenčnej identity nad 90 %, nad 80 %, nad 70 % atď. Poslednou možnosťou, ktorá predstavuje prípad hľadanie bez obmedzenia, je oblasť od 0 % do 100 % identity (význam sekvenčnej identity bol uvedený v podkapitole 4.2.5).
- **maximálna hodnota *error rate*** – táto metóda vyjadruje maximálny pomer medzi nájdenými TP a FP pármi v percentách (podrobne je popísaná v kapitole 5), ktorý musí metóda dodržať.

Ak užívateľ zadá tieto parametre, aplikácia mu na základe už urobenej analýzy metód vyhľadá požadovanú metódu a jej nastavenia, ktoré na testovacej sade dávali najlepšie výsledky. Popis tohto kroku bude uvedený ďalej. Keď aplikácia nájde požadované informácie, zobrazí užívateľovi najhlavnejšie parametre pre jeho hľadanie:

- **typ metód** – je vybraná jedna z možných metód, ak nebola zadaná priamo užívateľom.
- **hodnota maximálnej E-value –** jedná sa o maximálnu hodnotu E-value, ktorú môže mať pár nájdený metódou. Táto hodnota je už prepočítaná tým spôsobom, aby odpovedala veľkosti používanej databázy.
- **hodnota prahovej E-value** – zobrazuje sa len v prípade iteratívnych metód. Tieto hodnoty sú vyberané z odporučených hodnôt pri každej metóde.
- **maximálny počet iterácií metódy** – nastavenie počtu iterácií, ktoré metóda nesmie prekročiť, sama ale môže skončiť skôr.
- **ďalšie špecifické parametre metód** – tie môže užívateľ zadať manuálne, ich odporučenie ale nie je súčasťou tejto aplikácie (je plánované ako rozšírenie).

Postup nájdenia optimálnej metódy je závislý na existencii ROC súborov pre každé možné nastavenie (z tých, čo boli testované). Formát tohto súboru bol uvedený v kapitol 4.2.7.

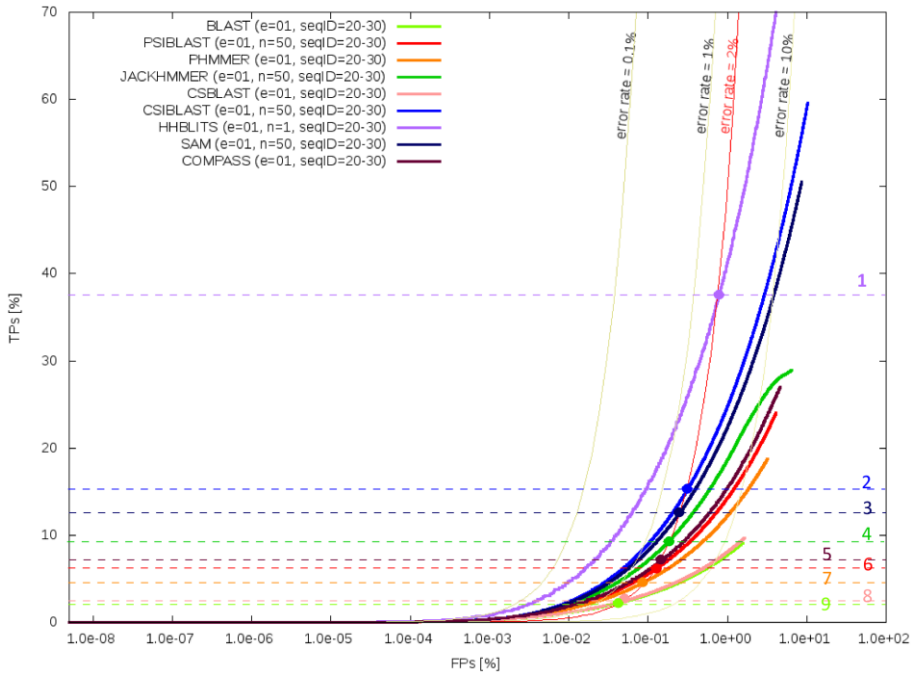
Všeobecne sa dá postup určenia optimálnej metódy ukázať na obrázku 4.7. Pre jej zistenie je nutné si v grafe spolu s krivkami pomeru TP a FP párov metód zobraziť maximálnu možnú hodnotu *error rate*, ktorú užívateľ zadal. V grafe na obrázku sú uvedené len metódy s jedným variantom nastavení parametrov, v skutočnosti sa hľadá medzi všetkými možnými nastaveniami (ak nie je toto nejakým spôsobom obmedzené). Krivka označujúca maximálnu hodnotu *error rate* by mala pretnúť ostatné krivky v jednom bode (ďalej budú označované ako priesečníky). Môžu nastať tieto situácie:

1. Ak by preťala krivku metódy vo viacerých bodoch, pre jednoduchosť uvažujeme najvyššie položený bod. Optimisticky predpokladáme, že ak metóda spĺňa pomer *error rate* s tak veľkou

hodnotou E-value, bude splnený aj pri akejkoľvek menšej hodnote (hodnota E-value bodov krivky rastie od jej počiatku).

2. Ak v žiadnom bode a krivka metódy sa vyskytuje vľavo od krivky *error rate*, zoberieme ako priesečník najvyšší bod krivky metódy (metóda splňuje požiadavku na *error rate* v každej situácii).
3. Ak v žiadnom bode a krivka metódy je napravo, tak túto metódu a jej nastavenie ďalej neuvažujeme (metóda nesplňuje za žiadnych okolností zadanú hodnotu *error rate*).

Takto sú získané priesečníky s krivkami všetkých metód a ich možných nastavení (farebné body 1 až 9 v grafe). Tieto priesečníky sú potom porovnané vzhľadom k hodnote na ose y, teda pomeru nájdených TP párov vzhľadom k celkovému počtu. Metóda, ktorej priesečník odpovedá najvyššej hodnote na ose y (v uvedenom grafe je to priesečník 1 metódy HHBLITS s prahovou E-value rovnou 0,01 a počtom iterácií 1), je potom spolu s jej parametrami riešením procesu hľadania optimálnej metódy. Požaduje sa čo najvyšší počet nájdených TP párov za splnenia podmienky maximálnej hodnoty *error rate*.



Obrázok 4.7. – Postup výberu najlepšej metódy.

Reálne postup hľadania vyzerá tak, že sú prechádzané ROC súbory všetkých metód (a ich nastavení) od konca smerom k začiatku. Postupuje sa teda od najvyšších hodnôt TP a FP (a E-value) v súbore k najnižším. Pri tomto postupe je hľadaný bod (riadok súboru), kde spočítaná hodnota *error rate* klesne pod užívateľom zadanú maximálnu hodnotu *error rate*. Takto nájdený riadok ROC súboru je následne prevzatý a použitý pri hľadaní nastavenia metódy (tej, ktorej výsledky ROC súbor popisuje). Hodnota E-value, ktorá je uvedená na tomto riadku je uvažovaná ako maximálna E-value, s ktorou môže táto metóda s týmto nastavením vyhľadávať, aby jej výsledky spĺňali zadanú

maximálnu hodnotu *error rate*. Ďalej je uložená aj hodnota počtu nájdených TP párov pre túto metódu a jej nastavenie.

Tieto dve hodnoty sú teda získané zo všetkých ROC súborov a následne porovnané podľa počtu TP párov. Uložený riadok z ROC súboru, ktorý má najvyšší počet TP párov je vybraný ako víťaz. Hodnota E-value získaná z tohto riadku je zobrazená ako maximálna E-value, ktorú má užívateľ použiť. Metóda a jej nastavenie, ktorú popisuje ROC súbor, z ktorého nájdený riadok pochádza, je prevzatá ako výsledok hľadania a zobrazená užívateľovi ako optimálna metóda, ktorú má použiť. Rovnako je z ROC súboru získané aj nastavenie maximálnej prahovej E-value a počtu iterácií.

4.3.3 Vyhľadanie homológnych párov

Aplikácia je dostupná v rámci virtuálneho stroja, ktorý je umiestnený v prílohe tejto práce. Pre správnu funkciu programu je nutné mať správne zadané cesty k externému nástroju a databáze, ktorú používa. Externé nástroje nie je treba inštalovať. Postačuje, ak sú stiahnuté a dostupné v súborovom systéme virtuálneho stroja a ak sú správne zadané parametre. Nie je treba ani nijako meniť systémové premenné v používanom operačnom systéme. Aplikácia spúšťa pre každú zadanú sekvenciu vybranú metódu v novom vlákne a čaká na vykonanie hľadania.

V aplikácii je možné v samostatnom dialógu spravovať uložené metódy a ich nastavenie. Je možné ukladať samostatne rovnaké metódy s rôznym základným nastavením. Pri každej metóde je možné zadať niekoľko sekvenčných databáz.

4.3.4 Zobrazenie výsledkov

Výsledky, ktoré sú získane zo spúšťaného nástroja po dokončení hľadania, sú spracované a zobrazené tak, že k zoznamu pôvodných (dotazovaných) proteínov je uvedený zoznam homológnych proteínov spolu so získanou E-value, ďalej skóre vyrátane touto metódou je zobrazené v samostatnej tabuľke.

Tieto záznamy sú zotriedené podľa hodnoty E-value od najmenšej po najväčšiu. Hodnoty skóre metódy sú uvedené iba pre prípad, že by sa užívateľ nechcel spoliehať len na univerzálny ukazovateľ – E-value. Ďalej je možné pre každú dvojicu zobraziť zarovnanie, ktoré vytvoril použitý nástroj.

Získané sekvencie homológov je možné jednoducho uložiť do súboru vo FASTA formáte.

5 Meranie úspešnosti metód na zvolenej dátovej sade

Meranie prebiehalo pre každú metódu zvlášť. V podkapitole 5.1 sú popísané metódy a ich nastavania, ktoré boli testované. Ďalej sú tu zobrazené výsledky pre jednotlivé oblasti sekvenčné identity formou grafu, ktorý má na ose x vždy uvedený pomer nájdených FP k celkovému počtu FP párov z referenčného zoznamu a na ose y potom pomer nájdených TP párov k celkovému počtu uvedenému v referenčnom zozname TP párov. V podkapitole 5.2 sú popísané vybrané najlepšie nastavania jednotlivých metód, porovnané medzi sebou a znázornené rovnakým typom grafu.

V prílohe 1. sú navyše umiestené grafy zobrazujúce ROC krivky v ROC priestore pre všetky metódy bez ohľadu na sekvenčnú identitu.

Výpočet prebiehal na gridoch organizácie MetaCentra, ktorá poskytuje vedcom a akademickým pracovníkom prístup k výpočtovým zdrojom [42]. Takto boli hľadané homológne páry pre každú z vybraných metód s vybranými hodnotami ich základných parametrov. Tieto hodnoty budú popísané samostatne v podkapitolách venovaných jednotlivým metódam.

5.1 Vyhodnotenie na základe typov metód

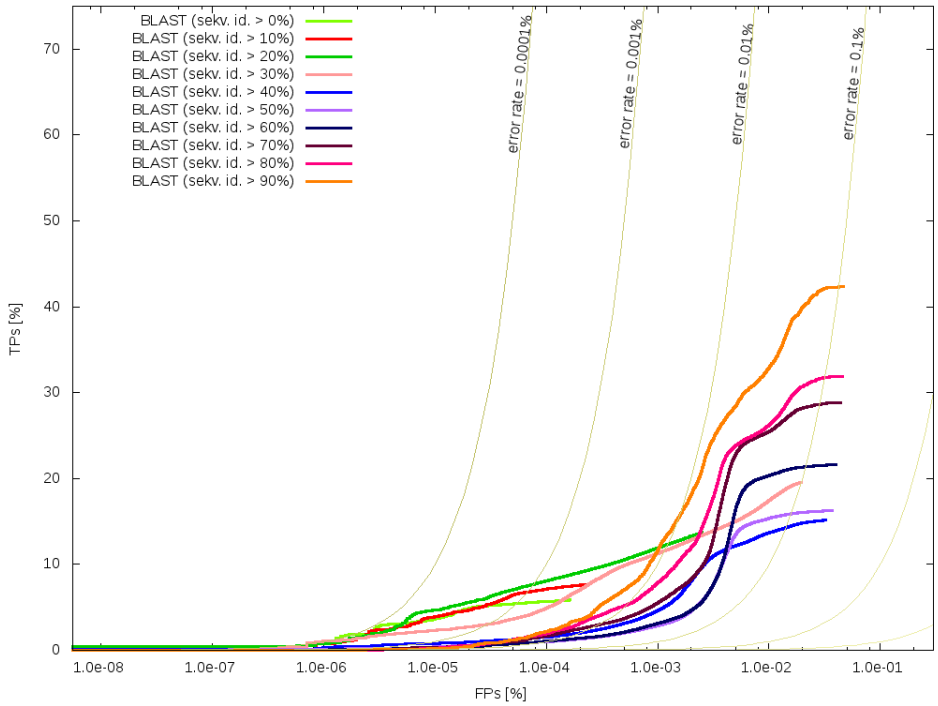
V tejto podkapitole sú zhrnuté výsledky testovania metód samostatne. Pri každej metóde sú v jednom grafe uvedené krivky s pomerom počtu TP párov k počtu FP párov podľa sekvenčnej identity. Pre každú skupinu sekvenčnej identity je zobrazená vždy len krivka metódy s najlepším nastavením parametrov, pre ktoré bolo získaných najviac TP párov na základe zvolenej hodnoty *error rate*.

5.1.1 BLASTp

Použitý nástroj BLASTp je súčasťou balíka BLAST+ verzia 2.2.29 od organizácie NCBI. Pre moje meranie bola použitá databáza *pdbaa* (získaná z webových stránok organizácie NCBI [43]), ktorá obsahuje sekvencie proteínov, ktoré sú dostupné v PDB. Nástroje založené na algoritme BLAST vyžadujú databázu v špeciálnom formáte BLAST. Na serveroch NCBI je dostupné aj množstvo iných databáz proteínov ako napríklad tieto:

- NR databáza proteínov (označenie *nr*)
- sekvencie proteínov pre metagenómy (*env_nr*)
- NCBI proteínové referenčné sekvencie (*refseq_protein*)
- Swiss-Prot databáza sekvencií (*swissprot*)

Nástroj BLASTp bol spúšťaný so základným nastavením. Na obrázku 5.1. je zobrazená ROC krivka získaná týmto nástrojom, z grafu je zrejmé, že nástroj je najlepší pri hľadaní najbližších homológov (tých najpodobnejších), ktoré majú vysokú identitu.



Obrázok 5.1. – ROC krivka zostavená z výsledkov nástroja BLASTp.

(Poznámka: u všetkých grafov je nepresne označená prvá oblasť sekvenčnej identity obsahujúci v skutočnosti aj páry so sekvenčnou identitou rovnou 0%)

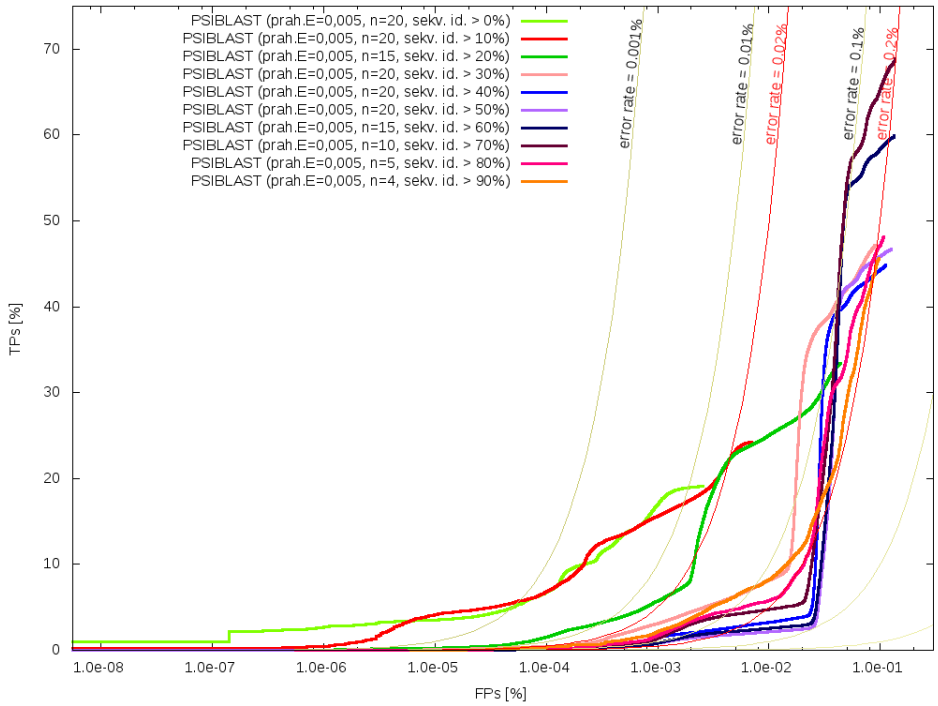
5.1.2 PSIBLAST

Nástroj PSIBLAST patrí medzi iteratívne metódy a je súčasťou rovnakého balíka ako nástroj BLASTp. Tiež využíva metódu BLAST pre hľadanie homológnych párov. Je možné využiť rovnakú sekvenčnú databázu.

Nástroj PSIBLAST bol spúšťaný s rôznymi nastaveniami maximálneho počtu iterácií a aj s rozdielnou veľkosťou prahovej E-value (tá sa vyskytuje len u iteratívnych metód). Boli testované tieto veľkosti maximálnej prahovej E-value: 0,01, 0,005 (základné nastavenie), 0,001. Krajné hodnoty boli vybrané s ohľadom na ostatné testované metódy, kde sú tieto hodnoty používané ako základné nastavenie. Pre počet iterácií boli vybrané hodnoty: 2, 3, 4, 5, 10, 15 a 20. V zozname nie je maximálny počet iterácií 1, ktorý odpovedá použitiu nástroja BLASTp.

Výsledky získané metódou PSIBLAST sú znázornené na obrázku 5.2. Na grafe sú zobrazené iba krivky zastupujúce variant metódy s maximálnou prahovou E-value 0,005, ktorá sa ukázala pre všetky sekvenčné identity ako najlepšia, pretože nájde najvyšší počet TP párov zo všetkých možných variant. Pre rôzne oblasti sekvenčnej identity bol nájdený rozdielny ideálny počet iterácií. Zatiaľ čo pre najvzdialenejšie homológne páry sa ukázalo ako najlepšie nastaviť veľký počet iterácií, pre blízke

homológy bol lepší menší počet. Je to zrejme spôsobené faktom, že pri väčšom počte iterácií sú do konštrukcie PSSM matice použité aj menej podobné domény, vďaka ktorým metóda určuje blízke homológy horšie.



Obrázok 5.2. – ROC krivka zostavená z výsledkov nástroja PSIBLAST.

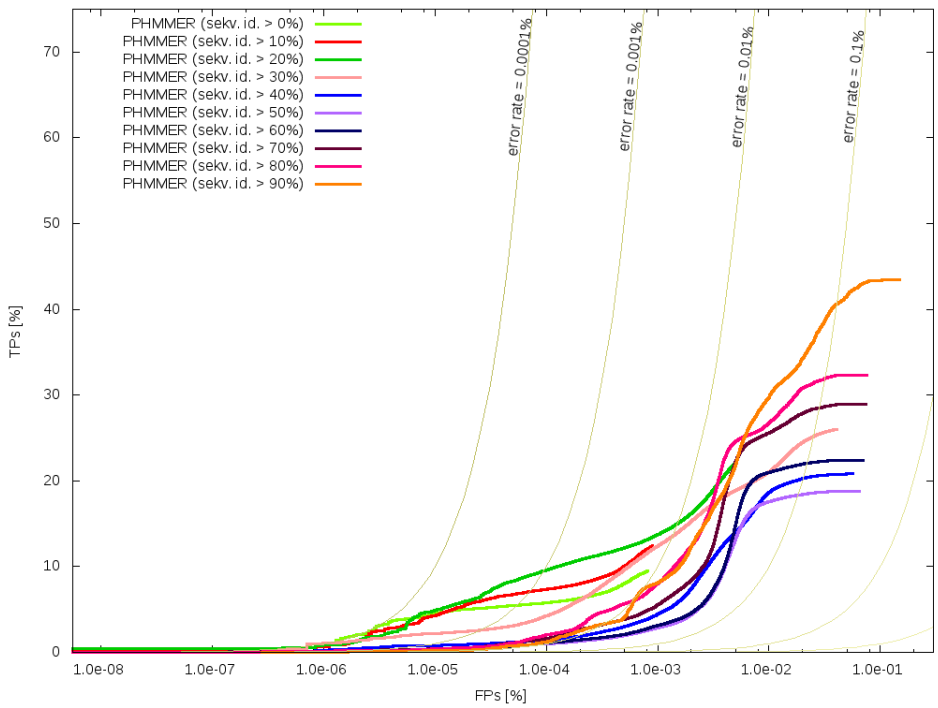
5.1.3 PHMMER

Nástroj PHMMER využíva metódu HMMER a pracuje s HMM profilmi. Tie si vytvára podľa zadanej sekvencie sám. Použitá verzia nástroja bola súčasťou balíka HMMER verzie 3.1 z webových stránok Janelia Research Campus [44]. Pre hľadanie využíva sekvenčnú databázu vo formáte FASTA a nevyžaduje žiadne špeciálne formátovanie, čím sa zjednodušuje jeho použitie.

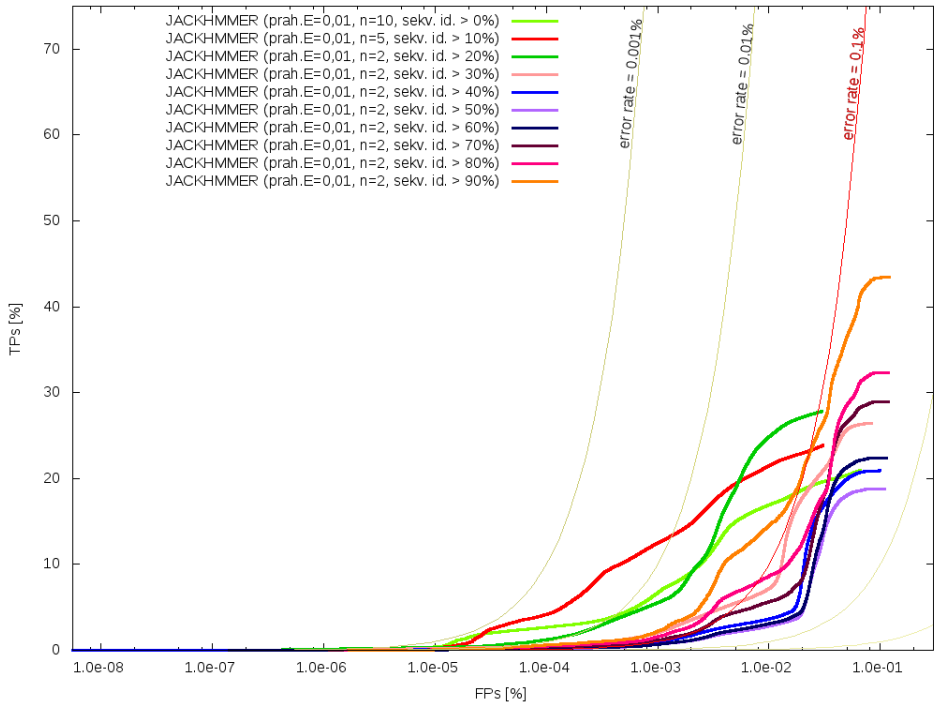
Bola použitá rovnaká databáza *pdbaa*, tento raz ale vo formáte FASTA. Tak ako aj v prípade nástroja BLASTp boli ponechané základné nastavenia. Získaná ROC krivka je zobrazená na obrázku 5.3. Je vidieť, že metóda je opäť výrazne najúspešnejší pri hľadaní blízkych homológov. Ak ju však porovnáme s metódou BLAST, tak má menšiu presnosť (podľa hodnôt *error rate*).

5.1.4 JACKHMMER

Nástroj JACKHMMER bol získaný z rovnakého balíka HMMER ako nástroj PHMMER. Pracuje s rovnakými nastaveniami, ale ide o iteratívny variant. Naviac je možné nastaviť maximálny počet iterácií a hodnotu prahu pre E-value. Tieto parametre boli nastavované tak, ako pre nástroj PSIBLAST, aby bolo možné následne nástroje porovnať. Pre nástroj JACKHMMER je základná hodnota prahovej E-value = 0,01. Pri testovaní bola vynechaná hodnota maximálneho počtu iterácií 1, keďže toto nastavenie odpovedá spusteniu nástroja PHMMER.



Obrázok 5.3. – Výsledky nástroja PHMMER zobrazené ROC krivkou.



Obrázok 5.4. – ROC krivka zostavená z výsledkov nástroja JACKHMMER.

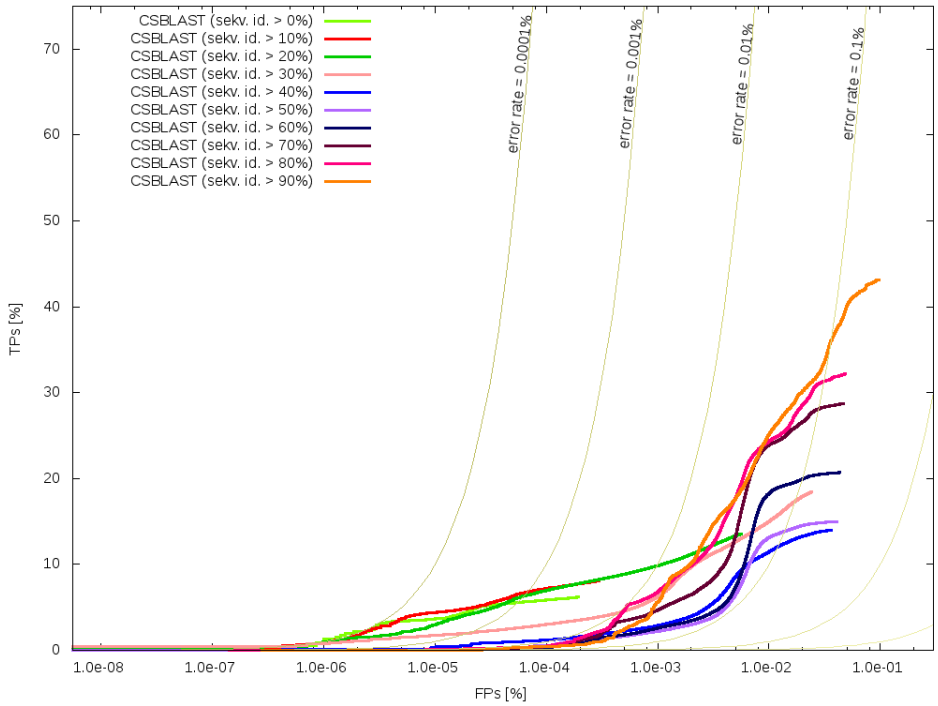
Z výsledkov získaných spúšťaním nástroja JACKHMMER na testovacej sade proteínov bol získaný graf 5.4. Hoci hodnota prahovej E-value výsledok výrazne neovplyvňuje, najlepšie výsledky boli vždy získané pre variant s hodnotou rovnou 0,01. Tak ako aj nástroju PSIBLAST, sa tu najlepšie darilo hľadať blízkych homológov pri malom počte iterácií. Väčší počet iterácií sa oplatí využiť pri hľadaní TP párov s malou sekvenčnou identitou.

5.1.5 CSBLAST

Nástroj CSBLAST je založený na modifikovanej metóde BLAST a bol získaný z webových stránok Max-Planck Institute for Developmental Biology [45]. Naviac vyžaduje nástroj BLASTpgp z balíka BLAST z webových stránok NCBI [43]. Metóda CSBLAST je založená na použití knižnice krátkych profilov, ktoré využíva pri určovaní pravdepodobnosti výskytu kontextovo špecifických mutácií. Pre testovanie som využila súbor so základným nastavením, ktorý bol súčasťou stiahnuté balíka.

Nástroje založené na metóde CSBLAST vyžadujú databázy v rovnakom formáte ako tie založené na BLAST. Pri testovaní zostalo nastavenie na základných hodnotách, okrem hodnoty max. E-value, tá bola menená tak ako pri nástrojoch BLASTp a PHMMER.

Výsledná ROC krivka je znázornená na obrázku 5.5. Jej výsledky sú relatívne porovnateľné s výsledkom z nástroja PHMMER.



Obrázok 5.5. – ROC krivka zobrazená z výsledkov nástroja CSBLAST.

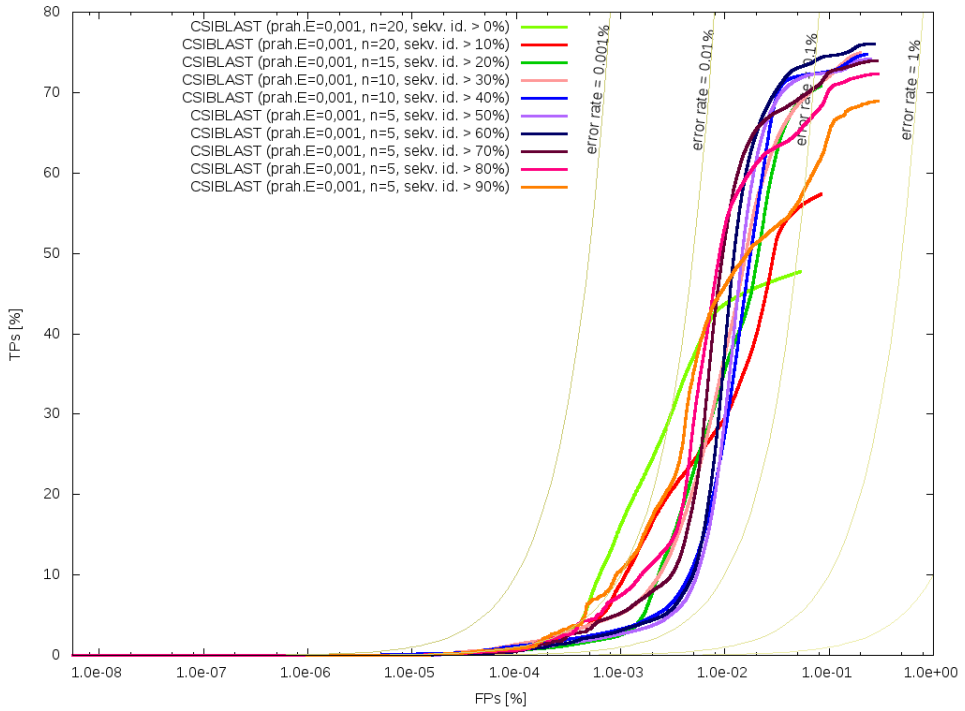
5.1.6 CSIBLAST

Ide o nástroj využívajúci iteratívny variant metódy CSBLAST. Vyžaduje rovnaké parametre a typ databázy ako vyššie spomenutý nástroj CSBLAST.

Jediným rozdielom pri testovaní bolo nastavovanie rôznych hodnôt prahovej E-value a maximálneho počtu iterácií. Tieto hodnoty boli nastavované rovnako ako v prípade nástrojov PSIBLAST a JACKHMMER. Základné nastavenie prahovej E-value bolo 0,001.

Na obrázku 5.6. je zobrazený výsledok pre testovaciu sadu proteínov znova rozdelených podľa sekvenčnej identity. Najlepšie výsledky v tomto prípade mala vždy maximálna prahová E-value

s hodnotou 0,001. Pri použití správneho počtu iterácií je možné nájsť veľkú časť homológov z celkového počtu. Súčasne je metóda veľmi presná, ako ukazujú vynesené hodnoty *error rate*. Slabinou tejto metódy je ale rýchlosť hľadania, ktorá je oproti ostatným iteratívnym metódam aj niekoľkokrát menšia.



Obrázok 5.6. – ROC krivka z výsledkov nástroja CSIBLAST.

5.1.7 HHBLITS

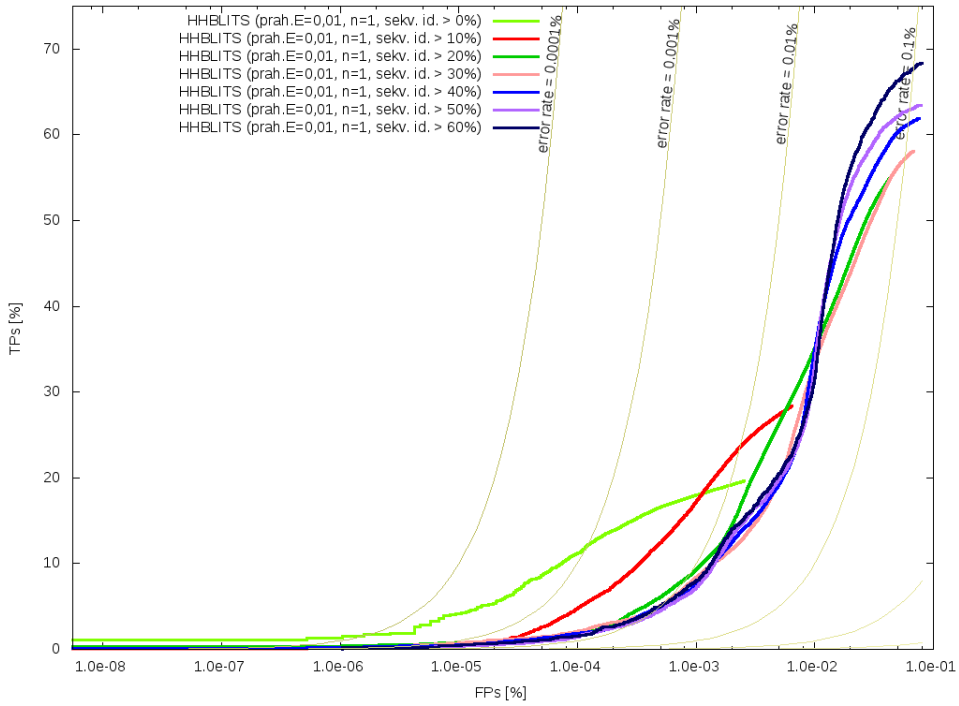
Nástroj HHBLITS je určený predovšetkým na prácu s HMM profilmi, ale je možné ho využiť aj na hľadanie zvolenej sekvencie voči sekvenčnej databáze. Je súčasťou balíka HHsuite, ktorý bol získaný z webových stránok Max-Planck Institute for Developmental Biology [45]. Problémom nástroja bolo získať rovnakú databázu *pdbaa* aká bola použitá pri ostatných metódach. Metódy z tohto balíka totiž vyžadujú špeciálny formát databáz. Na výber sú len tieto typy databáz:

- UniProt databáza so sekvenčnou identitou max. 20 % (s označením *uniprot20*)
- Non-Redundant databáza so sekvenčnou identitou max. 20 % (*nr20*)
- PDB databáza so sekvenčnou identitou max. 70 % (*pdb70*)
- SCOP databáza so sekvenčnou identitou max. 70 % (*scop70*)
- PfamA databáza (*pfamA*)

Z týchto piatich databáz som pre moje testovanie vybrala databázu *pdb70*, ktorá tvorí podmnožinu databázy *pdbaa* použitej pri ostatných metódach (databáza SCOP nebola použitá, keďže nie je dostupná aj pre iné metódy). Keďže táto databáza neobsahuje všetky domény, ktoré boli zahrnuté do testovacej sady, bolo nutné v tomto prípade obmedziť *zoznam TP, unknown a FP párov* tým spôsobom, aby neobsahovala domény chýbajúce v databáze *pdb70*.

Nástroj HHBLITS patrí medzi iteratívne metódy, maximálny počet iterácií sa nastavuje v rozsahu od 1 do 8. Viacej nie je treba, pretože metóda by mala rýchlo konvergovať. Ku konečnému výsledku dochádza zvyčajne rýchlejšie ako pri ostatných iteratívnych metódach.

Boli použité rovnaké hodnoty prahovej E-value: 0,01, 0,005, 0,001. Graf zobrazujúci ROC krivky pre jednotlivé sekvenčné identity je zobrazený na obrázku 5.7. Výsledky ukazujú, že najlepšie výsledky sú dosiahnuté pri maximálnej prahovej E-value rovnej 0,001. Zaujímavosťou je, že pri hľadaní párov s akoukoľvek sekvenčnou identitou je najúspešnejší variant s jednou iteráciou. Zrejme je to dané tým, že metóda vyhodnocuje sekundárnu štruktúru proteínov a rýchlo konverguje. Metóda pracovala iba s databázou s maximálnou sekvenčnou identitou do 70 %, preto chýbajú ROC krivky pre skupiny s vyššou identitou.



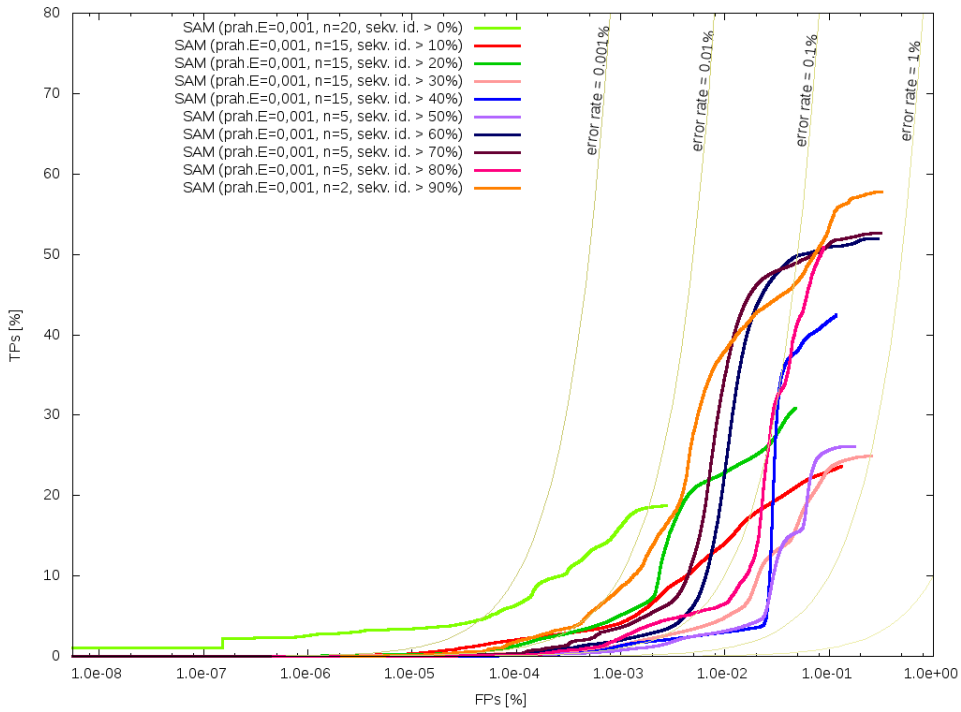
Obrázok 5.7. – ROC krivka zobrazujúca výsledky nástroja HHBLITS.

5.1.8 SAM

SAM je ďalším z nástrojov určených primárne k práci s HMM profilmi, dokáže však pracovať aj so sekvenciami a sekvenčnými databázami. Bol získaný z webových stránok Baskin Center for Computer Engineering and Science [46]. Pracuje s databázami vo formáte FASTA, a tak bola použitá rovnaká sekvenčná databáza *pdbaa* ako v prípade nástrojov PHMMER a JACKHMMER.

Nástroj SAM pracuje iteratívne a pri testovaní bol nastavovaný parameter maximálny počet iterácií na už známe hodnoty: 1, 2, 3, 4, 5, 10, 15 a 20. Štandardne boli nastavované aj hodnoty maximálnej prahovej E-value.

Výsledky je možné vidieť na obrázku 5.8. Ako pri väčšine iteratívnych metód bol najlepším variantom vždy ten so základným nastavením prahovej E-value, v tomto prípade na hodnotu 0,001.



Obrázok 5.8. – ROC krivka zostavená z výsledkov metódy SAM.

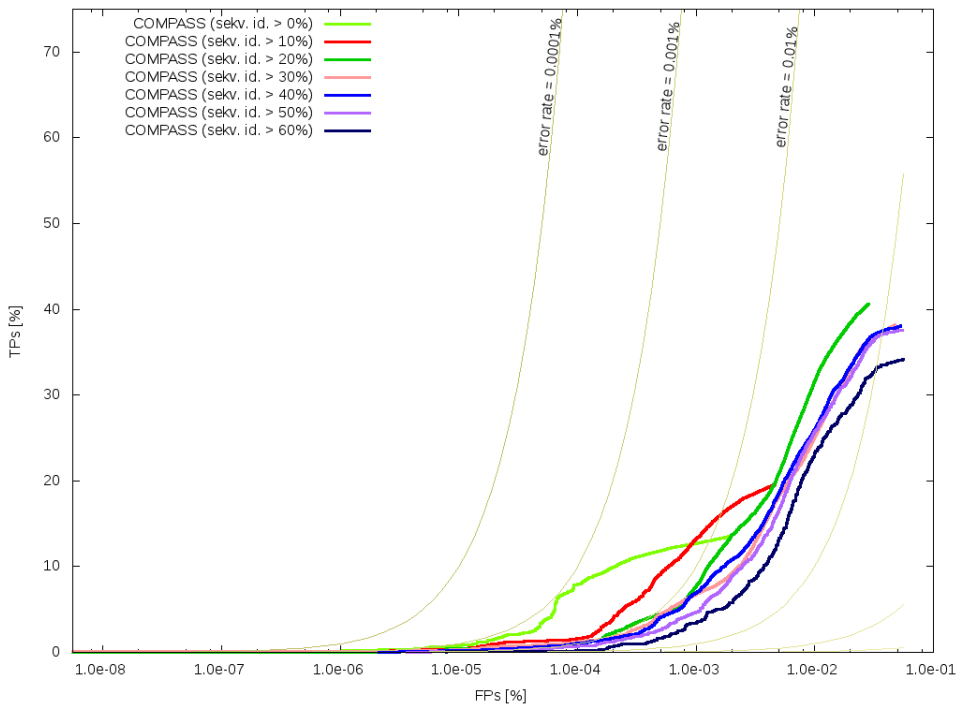
5.1.9 COMPASS

Nástroj COMPASS bol získaný z webových stránok Howard Hughes Medical Institute [47]. Tento nástroj je v skutočnosti zložený z dvoch skriptov. Je založený na práci s profilmi, ktoré sú vytvorené z viacnásobného zarovnania, prvý skript teda najprv prehľadá pomocou nástroja BLASTpgp zadanú databázu a vytvorí zarovnanie so zadanou sekvenciou. Toto zarovnanie je potom vstupom druhého skriptu, ktorý prevedie vyhľadávanie v profilovej databáze. Výsledkom je tak ako aj u iných metód zoznam nájdených párov a ich zarovnanie.

Za databázu, pre prvú časť výpočtu s nástrojom BLASTpgp, bola použitá už spomenutá databáza *pdbaa* vo formáte BLAST. Pre nástroj COMPASS potom existujú tieto naformátované profilové databázy:

- PDB databáza so sekvenčnou identitou max. 70 % (*pdb70*)
- SCOP databáza so sekvenčnou identitou max. 40 % (*scop40*)

Z týchto možností bola z rovnakých dôvodov ako pre nástroj HHBLITS vybraná databáza *pdb70*. Rovnakým spôsobom potom prebiehalo aj vyhodnotenie výsledkov nástroja. Získané výsledky sú na zobrazené obrázku 5.9.



Obrázok 5.9. – ROC krivka zostrojená z výsledkov nástroja COMPASS.

5.2 Vyhodnotenie na základe sekvenčnej identity

V tejto práci som sa zamerala na vyhodnotenie výsledkov metód s ohľadom na zvolenú oblasť sekvenčnej identity. Pre tieto oblasti som hľadala najlepšie varianty od každej metódy a z tých vyberala tu najlepšiu, pri ktorej získame najviac skutočne homológnych párov. Percentuálne zastúpenie TP párov z celkového počtu v jednotlivých oblastiach je znázornené v tabuľke 5.1.

sekvenčná identita	>= 0 %	> 10 %	> 20 %	> 30 %	> 40 %
zastúpenie TP párov v %	100	76,91	41,27	17,77	12,47

sekvenčná identita	> 50 %	> 60 %	> 70 %	> 80 %	> 90 %
zastúpenie TP párov v %	7,87	4,07	1,37	0,60	0,19

Tabuľka 5.1. – Percentuálne rozdelenie všetkých TP párov do jednotlivých oblastí podľa sekvenčnej identity.

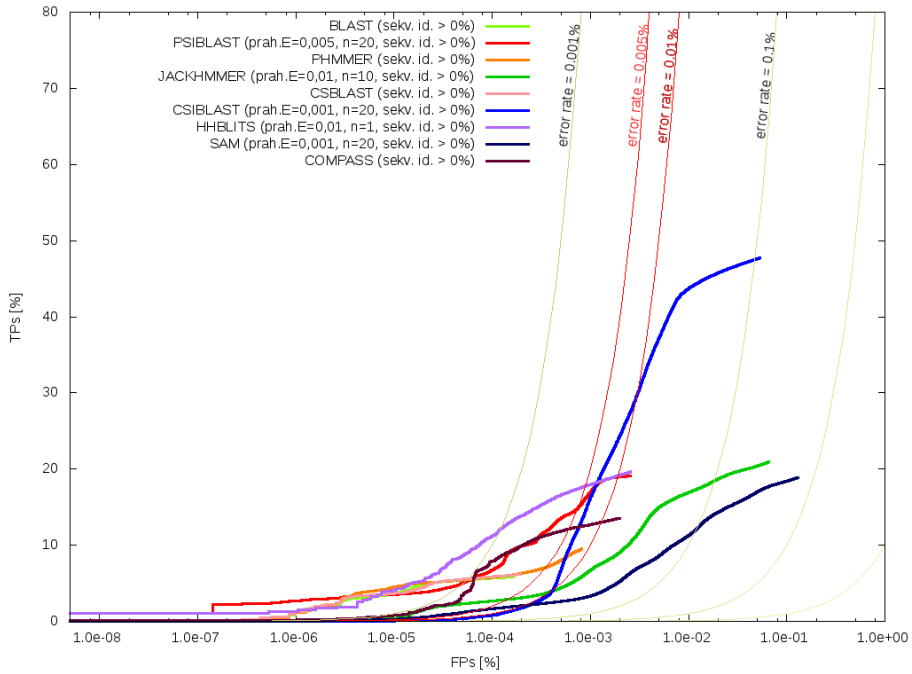
Pri hodnotení metód je vždy dôležité uvažovať nejakú mierku, podľa ktorej zvolíme najlepšiu metódu. V tejto práci je takou mierkou hodnota *error rate*, podľa ktorej je možné jasne určiť, ktorá metóda je najlepšia podľa počtu nájdených TP párov.

Hľadanie homológnych párov proteínov s ľubovoľnou sekvenčnou identitou

Táto oblasť je významná tým, že zahŕňa všetky existujúce TP páry a jedná sa o prípady, kedy užívateľ hľadá aj tie najvzdialenejšie príbuzné proteíny. Pri tomto hľadaní sú najúspešnejšie iteratívne metódy, ako je možné vidieť na obrázku 5.10.

Ktorú metódu vyberieme ako tú najlepšiu, závisí od voľby hodnoty *error rate*. Ak by sme požadovali hodnotu najviac 0,005%, bude vybraná metóda HHBLITS s 1 iteráciou. Avšak táto je schopná pre túto oblasť nájsť relatívne malý počet TP párov. Ak by sme zadali maximálnu mieru chybovosti (error rate) na 0,01%, podarí sa nám nájsť väčší počet skutočne homológnych párov, tentoraz ale s metódou CSIBLAST s 20 možnými iteráciami.

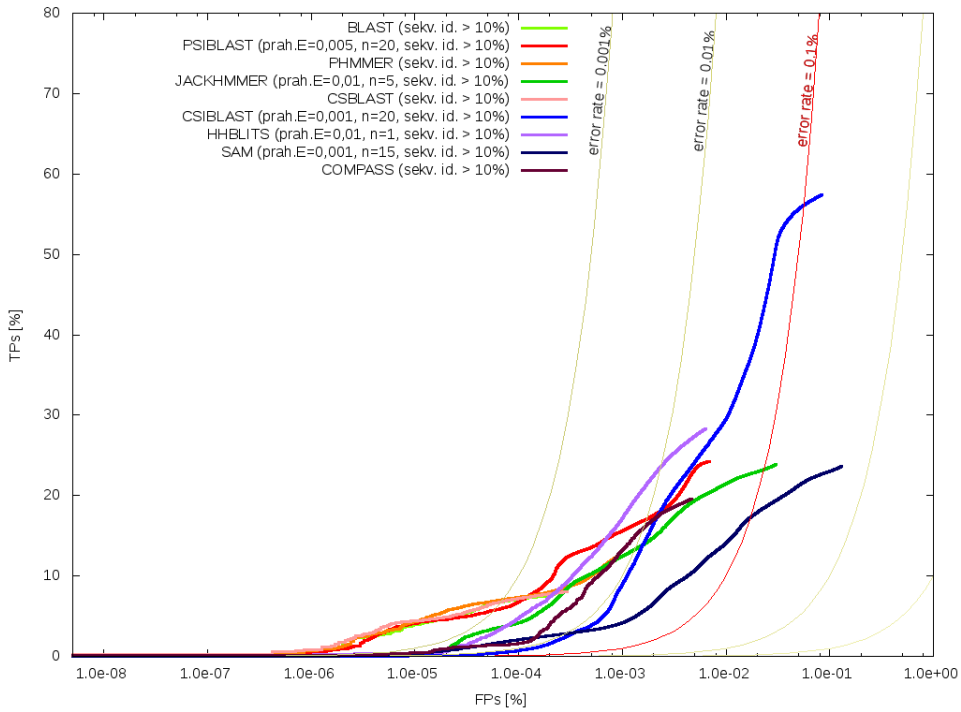
Z obrázku je zrejmé, že základné nástroje ako PHMMER a BLASTp majú veľké problémy nájsť väčšie množstvo TP párov, ktorých veľkú časť (v tejto oblasti) tvoria vzdialene príbuzné páry.



Obrázok 5.10. – Výsledky metód pre hľadanie homológnych párov s akoukoľvek sekvenčnou identitou.

Hľadanie homológnych párov so sekvenčnou identitou väčšou ako 10 %

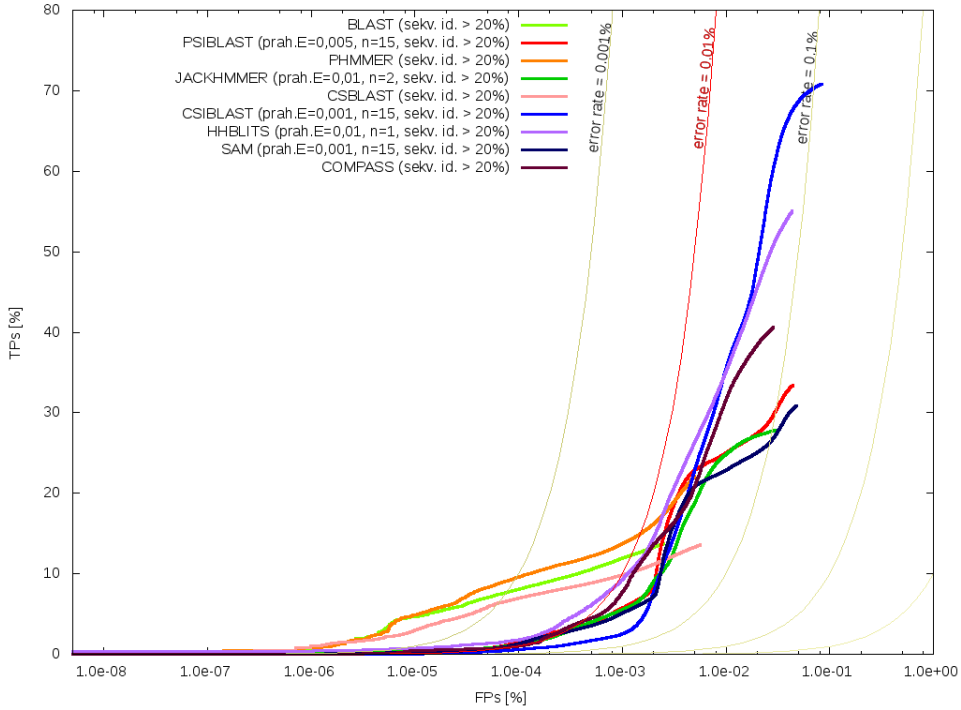
V grafe na obrázku 5.11 sú znázornené krivky popisujúce výsledky metód pre oblasť so sekvenčnou identitou väčšou ako 10%. Rovnako ako v predchádzajúcej oblasti (ktorej je táto podmnožinou) sa prejavuje dominancia iteratívnych metód a to predovšetkým metódy CSIBLAST s maximálnym počtom iterácií. Opäť môžeme povedať, že vybraný variant metódy závisí na zadanej hodnote *error rate*. V grafe je vidno, že niektoré krivky nepreťali krivku znázorňujúcu maximálnu možnú hodnotu *error rate*. V takom prípade, kedy krivka výsledku metódy leží naľavo od krivky *error rate*, sa pri porovnaní uvažuje najväčší nájdený počet TP párov. Dôvodom, prečo sa krivky nepreťali je fakt, že hľadanie homológnych párov bolo zastavené pri dosiahnutí maximálnej hodnoty E-value, ktorú metóda umožňuje nájsť.



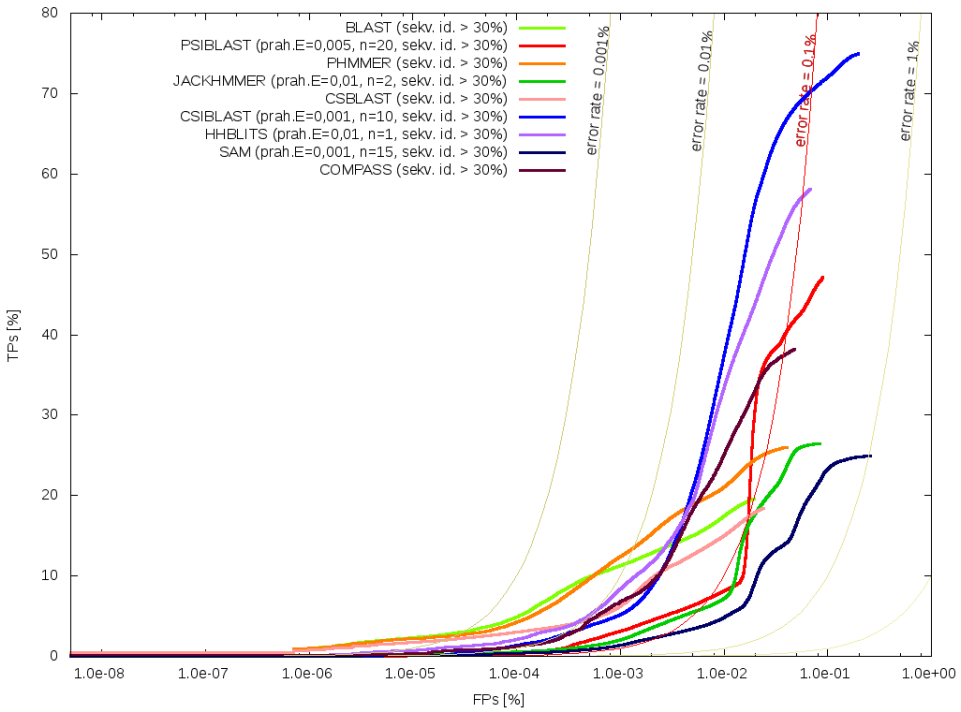
Obrázok 5.11. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 10%.

Hľadanie homológnych párov so sekvenčnou identitou nad 20%, nad 30% a nad 40%

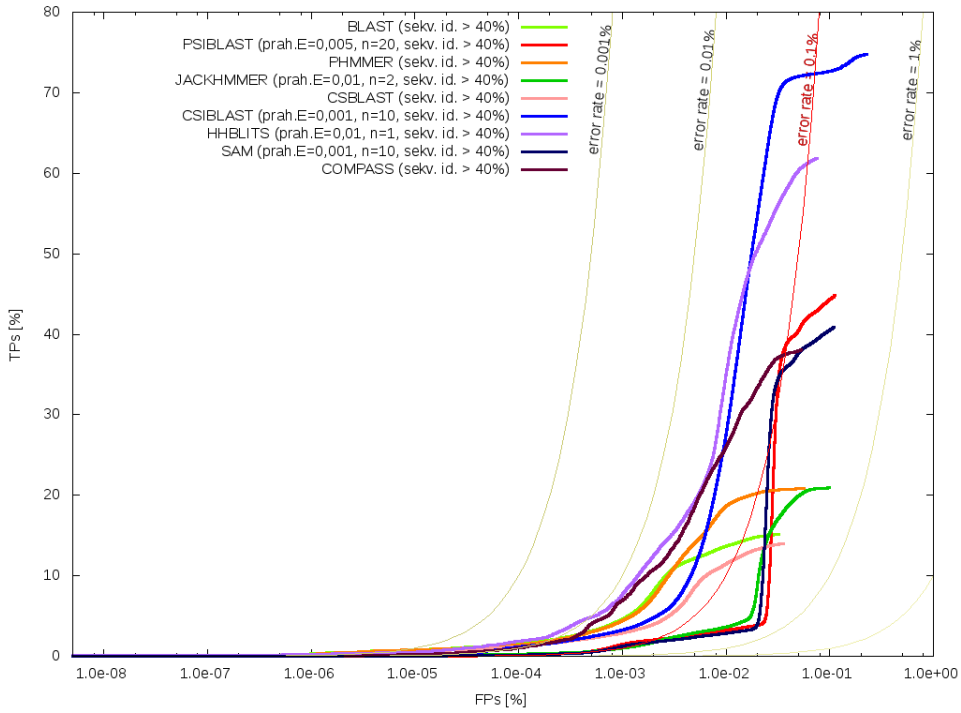
Podobný trend ako v predchádzajúcej skupine je vidieť aj pri tomto intervale. Jasne dominuje metóda CSIBLAST, ktorej stačí už len 15 iterácií, a HHBLITS s jednou iteráciou. So zvyšujúcim sa podielom viac identických párov klesá potrebný počet iterácií. To je možné vidieť na obrázkoch 5.12 až 5.14.



Obrázok 5.12. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 20%.



Obrázok 5.13. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 30%.

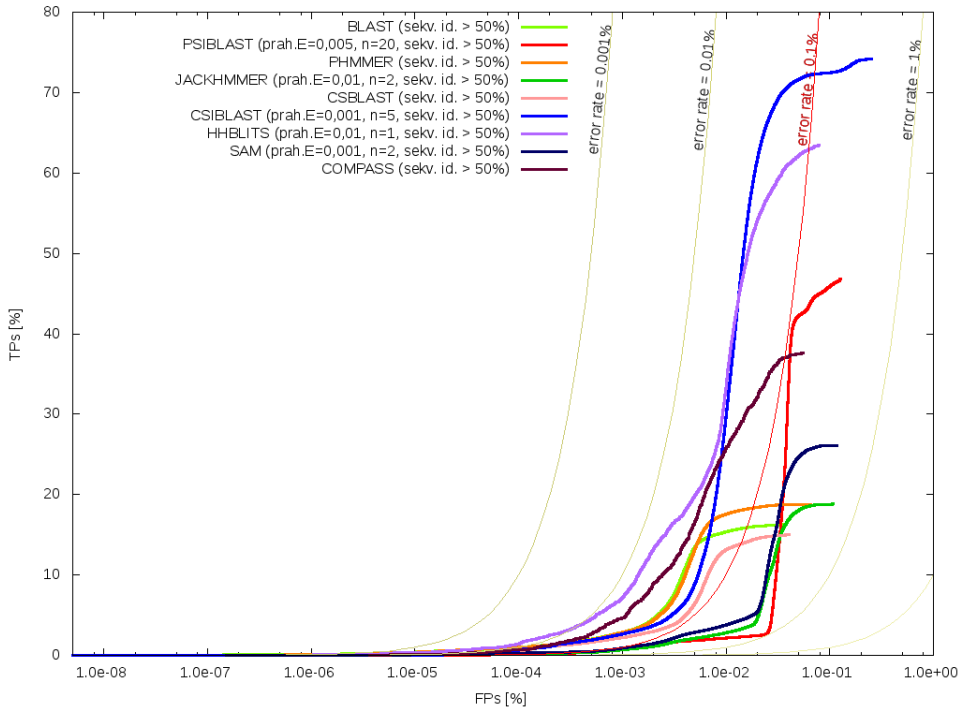


Obrázok 5.14. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 40%.

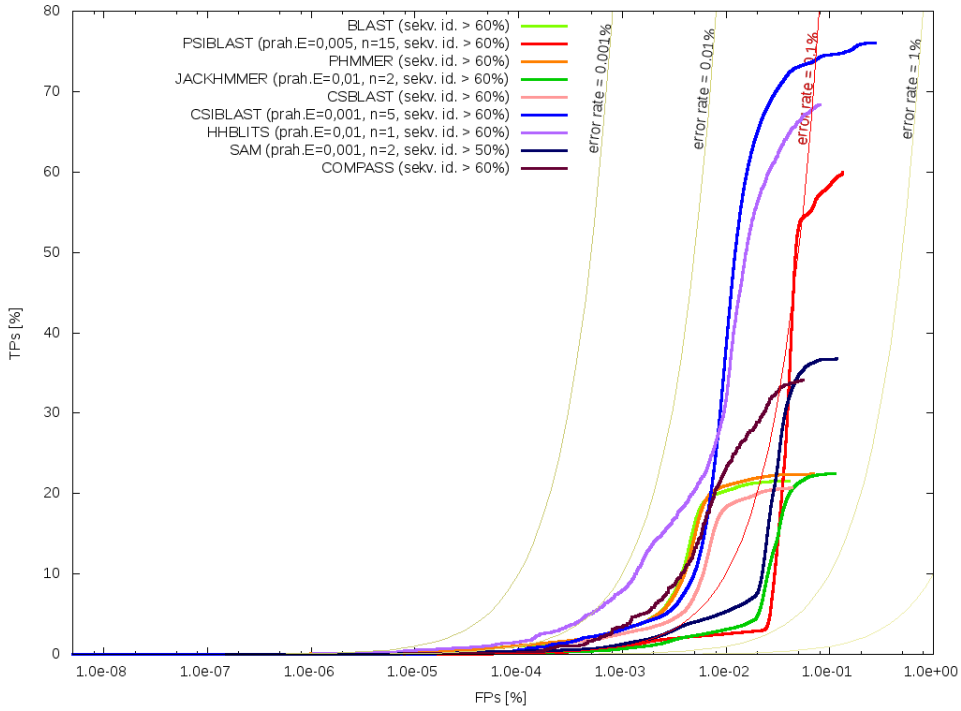
Hľadanie homológnych párov so sekvenčnou identitou od 60% a od 70 %

Keď mierime do týchto oblastí sekvenčnej identity, stačia pri iteratívnych metódach aj najmenšie počty iterácií. Metóda CSIBLAST dokáže pri zadanej miere *error rate* rovnej hodnote 0,1% nájsť maximálny počet TP párov (z testovaných možností) pri maximálne 5 iteráciách. S rastúcim počtom

viac sekvenčne identických párov sa zlepšujú aj výsledky metódy PSIBLAST (viď obrázky 5.15 a 5.16)



Obrázok 5.15. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 50%.

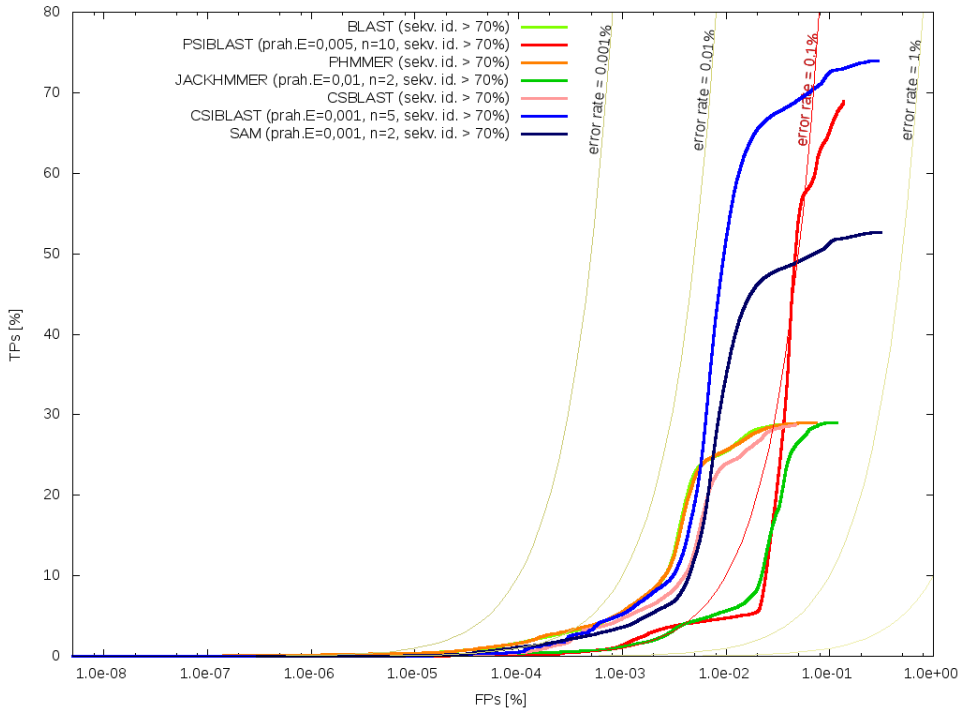


Obrázok 5.16. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 60%.

Hľadanie homológnych párov so sekvenčnou identita od 70% a od 80 %

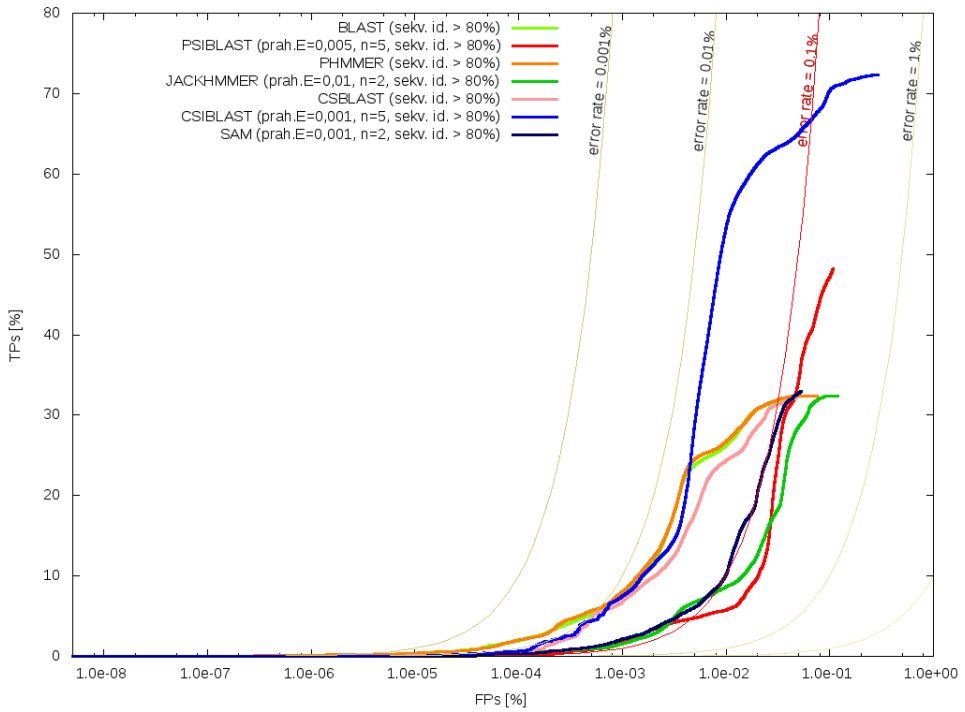
V tejto oblasti už neboli používané metódy HHBLITS a COMPASS, pre ktoré sa nepodarila získať žiadnu naformátovanú databázu s doménami so sekvenčnou identitou vyššou ako 70 %.

Najlepšou voľbou pre vybranú *error rate* rovnajúcu sa hodnote 0,1% pre páry so sekvenčnou identitou od 70% (viď obrázok 5.17) sú metódy CSIBLAST s maximálne 5 iteráciami a PSIBLAST s maximálne 10 iteráciami. Pri tomto výbere je možné zvážiť aj rýchlosť hľadania, ktorá by znamenala výhodu pre PSIBLAST, aj keď môže používať väčší počet iterácií.



Obrázok 5.17. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 70%.

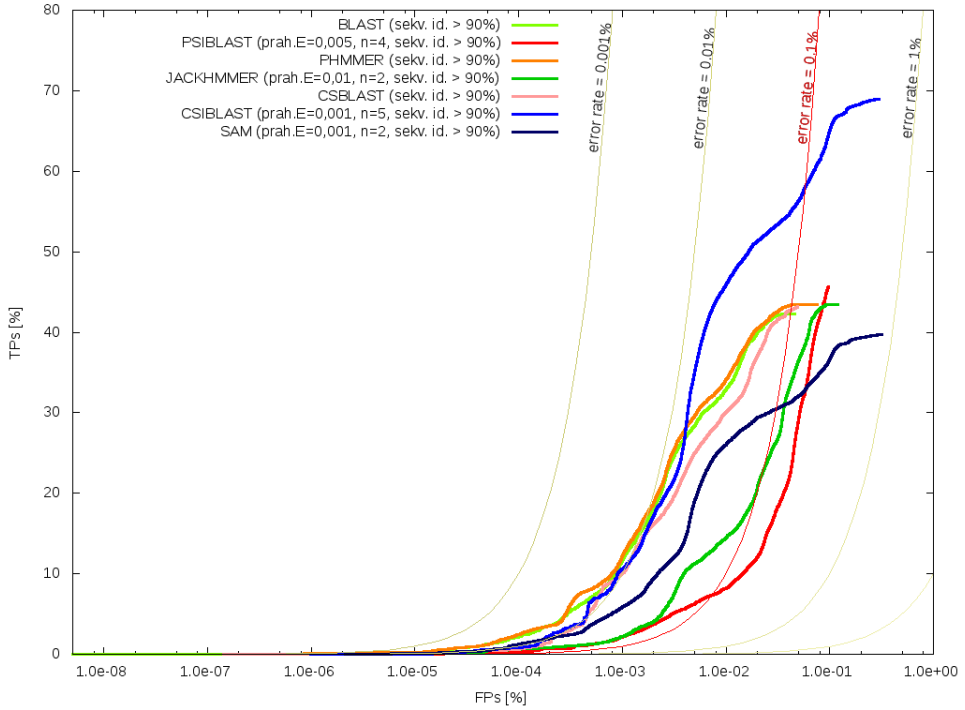
V nasledujúcej oblasti opäť pre väčšinu hľadaní ostáva ako najlepšia metóda CSIBLAST s 5 iteráciami (obr. 5.18.).



Obrázok 5.18. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 80%.

Hľadanie najbližších homológnych párov so sekvenčnou identitou nad 90%

Pri párach s najvyššou identitou je nutné brať do úvahy aj neiteratívne metódy, ktoré sú schopné nájsť okolo 50% všetkých TP párov. Najlepšou metódou, pre maximálnu hodnotu *error rate* rovnej hodnote od 0,001% a viac, zostáva metóda CSIBLAST s maximálnymi 5 iteráciami (viď obr. 5.19.).



Obrázok 5.19. – Výsledky metód pre hľadanie homológnych párov so sekvenčnou identitou nad 90%.

5.2.1 Namerané najlepšie výsledky podľa sekvenčnej identity

Okrem samotného ohodnotenia metód podľa schopnosti nájsť čo najväčší počet TP párov pri zadanej hodnote *error rate* bola skúmaná aj rýchlosť vyhľadávania. Výsledky sú zhrnuté v nasledujúcej tabuľke 5.2.

Z tabuľky je zrejmé, že vo väčšine prípadov, predovšetkým pri hľadaní párov so strednou a vyššou sekvenčnou identitou, sa podarilo nájsť najviac TP párov pomocou metódy CSIBLAST. Tieto výsledky sú samozrejme dosť závislé od zvolenej hodnoty *error rate*.

Pri porovnaní hrá dôležitú úlohu aj rýchlosť vyhľadávania homológnych proteínov. Preto bolo uskutočnené aj meranie priemerného času, za aký je každá z metód schopná nájsť pre doménu s priemerne dlhou sekvenciou aminokyselín jej homológne domény. Pokus bol uskutočnený s piatimi priemerne veľkými doménami (dĺžka okolo 170 aminokyselín).

Aby bolo toto meranie vypovedajúce, bol pre všetky merania využitý rovnaký stroj s nasledujúcou konfiguráciou: 4-jadrový procesor Intel Core i3 (3,4 GHz) s 4MB L2 cache, 8GB RAM, pevný disk 1TB 7200 otáčok/minutú. Výsledky sú zhrnuté v nasledujúcej tabuľke 5.2.

Sekvenčná identita [%]	***error rate* [%]**	**Názov a parametre najlepších metód**	**Nájdený počet TP párov [%]**	**Čas hľadania pre priemernú doménu [s]**
> 0	0,005	HHBLITS (n = 1, e = 0,01)	18	3,5
		PSIBLAST (n = 20, e = 0,005)	15	4,3
	0,01	CSIBLAST (n = 20, e = 0,001)	30	47
		HHBLITS (n = 1, e = 0,01)	19	3,5
> 10	0,1	CSIBLAST (n = 20, e = 0,001)	58	47
		HHBLITS (n = 1, e = 0,01)	30	3,5
> 20	0,01	PHMMER	15	1,1
		BLASTp	12	0,3
> 30	0,1	CSIBLAST (n = 10, e = 0,001)	70	23
		HHBLITS (n = 1, e = 0,01)	58	3,5
> 40	0,1	CSIBLAST (n = 10, e = 0,001)	72	23
		HHBLITS (n = 1, e = 0,01)	60	3,5
> 50	0,1	CSIBLAST (n = 5, e = 0,001)	72	12,2
		HHBLITS (n = 1, e = 0,01)	60	3,5

> 60	0,1	CSIBLAST (n = 5, e = 0,001) HHBLITS (n = 1, e = 0,01)	73 69	12,2 3,5
> 70	0,1	CSIBLAST (n = 5, e = 0,001) PSIBLAST (n = 10, e = 0,005)	70 58	12,2 4,3
> 80	0,1	CSIBLAST (n = 5, e = 0,001) PHMMER	68 33	12,2 1,1
> 90	0,1	CSIBLAST (n = 5, e = 0,001) PHMMER	60 46	12,2 1,1

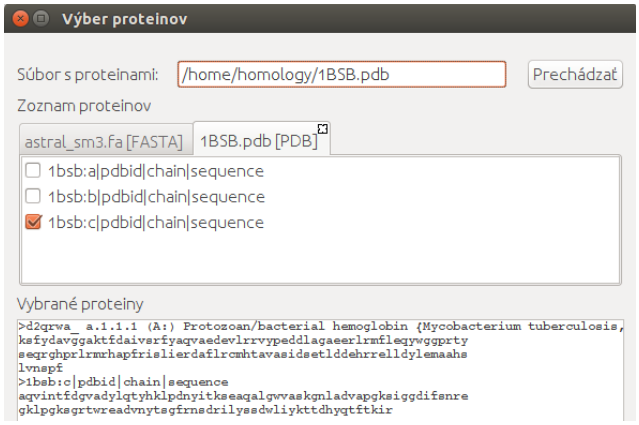
Tabuľka 5.2. – Najlepšie výsledky pre danú sekvenčnú identitu so zvolenou *error rate*.

(Poznámka: pri popise metódy označuje *n* maximálny počet iterácií metódy a *e* v prípade iteratívnych metód je rovné maximálnej prahovej E-value)

6 Testovanie aplikácie a jej ovládanie

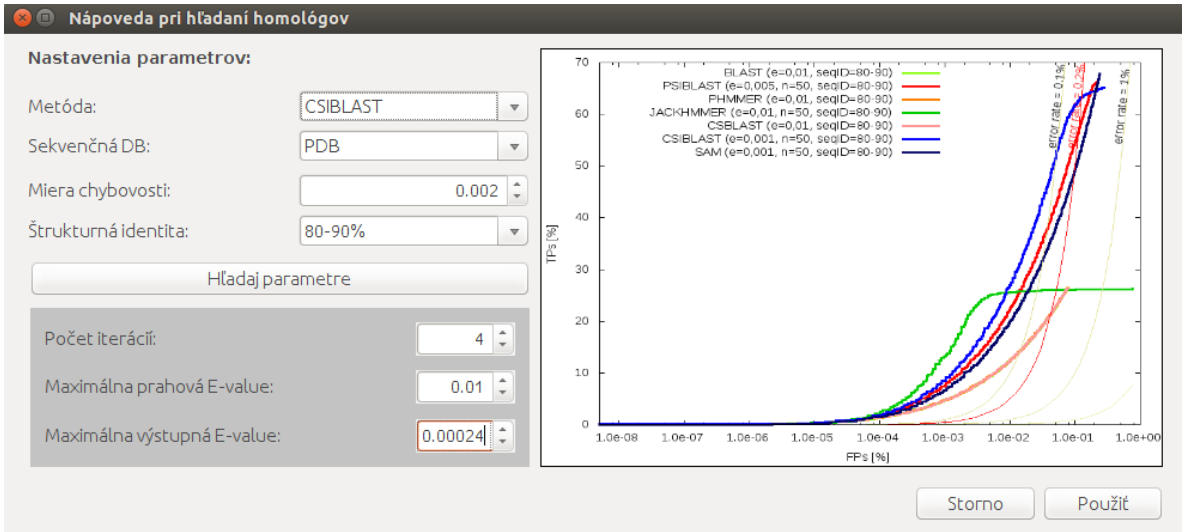
Aplikácia bola testovaná s pomocou vybranej vzorky domén z hierarchie SCOP, ktoré boli umiestnené do niekoľkých súborov.

Na obrázku 6.1. je vidieť načítanie jednotlivých súborov s doménami, pre ktoré chceme vyhľadávať homológne domény. Je možné načítať súbory vo formáte FASTA alebo PDB. Vybrané domény, ktoré budú použité pre hľadanie, sú zobrazené už len vo formáte FASTA. Takto naformátovaný zoznam bude použitý ako vstup vybraného nástroja pre hľadanie homológov.



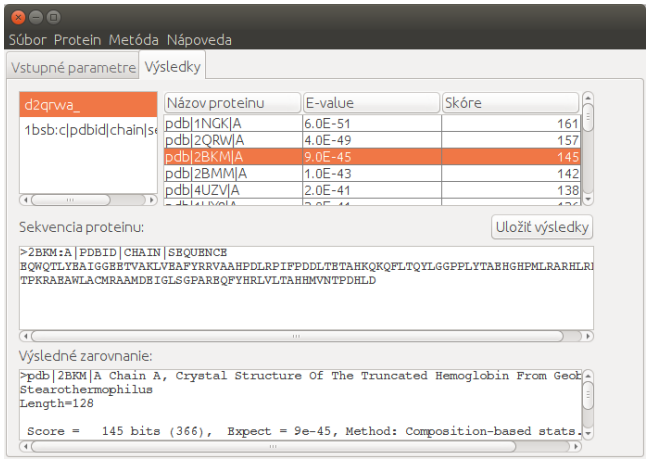
Obrázok 6.1. – Načítanie zdrojových súborov a výber proteínov.

Ďalej môže užívateľ využiť hlavnú prednosť aplikácie, ktorou je pomoc pri výbere metódy hľadania a nájdenie jej ideálnych parametrov. Takéto dialógové okno je zobrazené na obrázku 6.2., kde po zadaní akcie *Hľadaj parametre* aplikácia podľa uvedených parametrov typu metódy, miery chybovosti a štruktúrnej identity nájde najlepšie nastavenie pre parametre počtu iterácií, maximálnej prahovej E-value a maximálnej výstupnej E-value. Ak užívateľ nezadá konkrétny typ metódy, je nájdený aj ten. Tie parametre sú len odporučením a užívateľ ich môže ľahko zmeniť.



Obrázok 6.2. – Dialógové okno pre hľadanie optimálnej metódy.

Keď užívateľ vyberie metódu a jej nastavenia a spustí proces hľadania homológov, aplikácia bude postupne spúšťať odpovedajúci nástroj pre jednotlivé vybrané sekvencie. Dĺžka hľadania závisí od použitého nástroja a počtu iterácií. Pre niektoré metódy môže proces trvať dlho, až niekoľko minút v závislosti na počte sekvencií (predovšetkým pri nástrojoch CSIBLAST, SAM, COMPASS). Výsledky metód sú spracované a zobrazené v záložke *Výsledky* (viď. obrázok 6.3.). V zozname v ľavej hornej časti sú uvedené dopytované proteíny, v pravej časti sú potom k nim nájdené odpovedajúce homológne proteíny spolu s hodnotou ich E-value a získaného skóre danou metódou (význam tohto skóre závisí na type metódy). V dolnej časti je zobrazená sekvencia nájdeného homológneho proteínu a jeho zarovnanie k vstupnému proteínu.



Obrázok 6.3. – Dialógové okno zobrazujúce výsledky hľadania.

7 Záver

Motiváciou pre vznik tejto práce bolo predovšetkým to, že hoci v dnešnej dobe existuje množstvo metód pre vyhľadávanie homológnych sekvencií, každá z nich pracuje vhodnejšie pre iné vstupné požiadavky. Cieľom bolo vytvoriť nástroj, ktorý by pomáhal užívateľom pri rozhodovaní akú metódu a s akými nastaveniami vybrať, aby fungovala čo najlepšie pre nimi zadané kritéria. Za týmto účelom bolo otestovaných viacero metód a na základe zhodnotenia výsledkov vznikla podporná aplikácia.

Pre jednotlivé metódy a vybrané hodnoty ich parametrov boli uvedené ich výsledky pre zvolenú testovaciu vzorku dát. Bolo uskutočnené porovnanie jednotlivých nastavení a metód medzi sebou. Oproti iným prácam na túto tému boli takto získané dáta vyhodnotené aj z pohľadu sekvenčnej identity (získanej po štruktúrnom zarovnaní) a rozdelené do intervalov po desiatich percentách. Takto získané hodnoty môžu byť užitočné pre užívateľa, ktorého zaujímajú iba homológne proteíny s určitou sekvenčnou identitou. Testovacia vzorka dát bola vybraná z hierarchickej klasifikácie SCOPe. Vďaka použitiu tejto manuálne spravovanej hierarchie bolo možné získať relevantné informácie a jednoznačne určiť, aké percento získaných párov údajne homológnych proteínov bolo skutočne homológnych (podľa tvorcov tej klasifikácie) a koľko percent párov bolo chybne označených za homológne.

Po tomto prieskume existujúcich metód bol navrhnutý postup, ako je možné pomôcť užívateľovi pri výbere metódy a jej nastavenia, keď potrebuje nájsť k zadaným proteínom ich homológne páry, na základe uskutočnenej analýzy metód. Preto bola navrhnutá aplikácia, ktorá umožňuje užívateľovi načítať sekvencie proteínov, pre ktoré bude hľadať homológov, a potom spustiť nástroj, ktorý mu aplikácia na základe jeho požiadaviek poradí. Aplikácia je ponúkaná formou jednoduchého GUI s možnosťou jednoducho spúšťať rôzne nástroje a zobrazovať ich výstupy jednotným spôsobom a to jednoducho priamo v nej. Ponúka tiež možnosť uložiť aktuálne nastavenie tak, aby bolo možné daný experiment spustiť opakovane.

Z porovnania metód vyplynulo, že niektoré z nich podávajú veľmi odlišné výsledky od ostatných. Vo väčšine oblastí by bola jasnou voľbou metóda CSIBLAST, ale s rôzne nastaveným maximálnym počtom iterácií. Celkovo prekvapila predovšetkým metóda HHBLITS, ktorá veľmi úspešne hľadá homológne páry so strednou a vyššou sekvenčnou identitou a to už pri jednokrokovom variante. Pre metódy HHBLITS a COMPASS sa, bohužiaľ, nepodarilo získať odpovedajúce databázy, aby bolo možné ich plnohodnotne porovnať s ostatnými metódami pri všetkých hodnotách sekvenčnej identity. Niektoré ďalšie metódy ako napríklad PROCAIN sa nepodarilo z dôvodu neznámych chýb a nedostupných zdrojových kódov vôbec vyskúšať.

V ďalšom pokračovaní práce by bolo možné sa zamerať na zhodnotenie výsledkov podľa dosiahnutej hodnoty pokrytia (*coverage*) podobným spôsobom, ako teraz pri sekvenčnej identite.

Literatúra

[1] ROSYPAL, Stanislav. *Úvod do molekulárnej biológie*. 1998. ISBN 80-902-5625-2.

[2] CRICK, Francis. Central Dogma of Molecular Biology. *Nature* [online]. 1970, roč. 227, č. 5258, s. 561–563. Dostupné z: doi:10.1038/227561a0

[3] ALBERTS, B., A. JOHNSON a J. LEWIS. *The Molecular Biology of the Cell*. 4. vyd. B.m.: Garland Science, 2002. ISBN 0-8153-3218-1.

[4] PEVSNER, Jonathan. *Bioinformatics and Functional Genomics*. 2. vyd. B.m.: Wiley, 2009. ISBN 9780470451489.

[5] SHARMA, Sudha. Non-B DNA Secondary Structures and Their Resolution by RecQ Helicases. *Journal of Nucleic Acids* [online]. 2011, roč. 2011, s. e724215 [vid. 1. leden 2015]. Dostupné z: doi:10.4061/2011/724215

[6] BASU, Malay Kumar, Eugenia POLIAKOV a Igor B. ROGOZIN. Domain mobility in proteins: functional and evolutionary implications. *Briefings in Bioinformatics* [online]. 2009, roč. 10, č. 3, s. 205–216 [vid. 5. leden 2015]. ISSN 1467-5463, 1477-4054. Dostupné z: doi:10.1093/bib/bbn057

[7] EMBL-EBI. *Pfam* [online]. Dostupné z: http://pfam.xfam.org/

[8] NCBI. *Conserved Domains Database* [online]. Dostupné z: http://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml

[9] MARCHLER-BAUER, Aron a Stephen H. BRYANT. CD-Search: protein domain annotations on the fly. *Nucleic Acids Research* [online]. 2004, roč. 32, č. Web Server issue, s. W327–331. ISSN 1362-4962. Dostupné z: doi:10.1093/nar/gkh454

[10] VENCLOVAS, Ceslovas. Methods for sequence-structure alignment. *Methods in Molecular Biology (Clifton, N.J.)* [online]. 2012, roč. 857, s. 55–82. ISSN 1940-6029. Dostupné z: doi:10.1007/978-1-61779-588-6_3

[11] BERMAN, Helen M., John WESTBROOK, Zukang FENG, Gary GILLILAND, T. N. BHAT, Helge WEISSIG, Ilya N. SHINDYALOV a Philip E. BOURNE. The Protein Data Bank. *Nucleic Acids Research* [online]. 2000, roč. 28, č. 1, s. 235–242 [vid. 17. květen 2015]. ISSN 0305-1048, 1362-4962. Dostupné z: doi:10.1093/nar/28.1.235

[12] UNIPROT CONSORTIUM. *UniProt* [online]. Dostupné z: http://www.uniprot.org/

[13] MURZIN, A. G., S. E. BRENNER, T. HUBBARD a C. CHOTHIA. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology* [online]. 1995, roč. 247, č. 4, s. 536–540. ISSN 0022-2836. Dostupné z: doi:10.1006/jmbi.1995.0159

[14] SIMAP. The Similarity Matrix of Proteins - Glossary. *SIMAP* [online]. [vid. 1. listopad 2014]. Dostupné z: http://liferay.csb.univie.ac.at/portal/web/simap/glossary#p_MainNavigationPortlet_WAR_SIMAP

[15] ALTSCHUL, S.F., W. GISH, W MILLER, E.W. MYERS a D.J. LIPMAN. *Basic Local Alignment Search Tool* [online]. [vid. 1. listopad 2014]. Dostupné z: http://www.blastalgorithm.com/

[16] NEEDLEMAN, S. B. a C. D. WUNSCH. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*. 1970, roč. 48, č. 3, s. 443–453. ISSN 0022-2836.

[17] SMITH, T. F. a M. S. WATERMAN. Identification of common molecular subsequences. *Journal of Molecular Biology*. 1981, roč. 147, č. 1, s. 195–197. ISSN 0022-2836.

[18] MARTÍNEK, Tomáš. *Štúdijné materiály k predmetu BIF* [online]. B.m.: FIT VUT Brno. 2013. Dostupné z: https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/BIF-IT/lectures?cid=9252

[19] HENIKOFF, S. a J. G. HENIKOFF. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*. 1992, roč. 89, č. 22, s. 10915–10919. ISSN 0027-8424.

[20] NATIONAL LIBRARY OF MEDICINE. *BLAST Program Selection Guide* [online]. [vid. 30. říjen 2014]. Dostupné z: http://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE_TYPE=BlastDocs&DOC_TYPE=ProgSelectionGuide#psi

[21] ACCELRYS INC. A SUBSIDIARY OF PHARMACOPEIA. *PSIBLAST* [online]. [vid. 1. listopad 2014]. Dostupné z: http://www.biology.wustl.edu/gcg/psiblast.html#function

[22] ALTSCHUL, S F, T L MADDEN, A A SCHAFFER, J ZHANG, Z ZHANG, W MILLER a D J LIPMAN. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*. 1997, roč. 25, č. 17, s. 3389–3402. ISSN 0305-1048.

[23] BIEGERT, A. a J. SÖDING. Sequence context-specific profiles for homology searching. *Proceedings of the National Academy of Sciences* [online]. 2009, roč. 106, č. 10, s. 3770–3775 [vid. 1. listopad 2014]. ISSN 0027-8424, 1091-6490. Dostupné z: doi:10.1073/pnas.0810767106

[24] ANGERMÜLLER, Christof, Andreas BIEGERT a Johannes SÖDING. Discriminative modelling of context-specific amino acid substitution probabilities. *Bioinformatics* [online]. 2012, roč. 28, č. 24, s. 3240–3247 [vid. 1. listopad 2014]. ISSN 1367-4803, 1460-2059. Dostupné z: doi:10.1093/bioinformatics/bts622

[25] YOON, Byung-Jun. Hidden Markov Models and their Applications in Biological Sequence Analysis. *Current Genomics* [online]. 2009, roč. 10, č. 6, s. 402–415 [vid. 1. listopad 2014]. ISSN 1389-2029. Dostupné z: doi:10.2174/138920209789177575

[26] FORNEY, Jr., G.D. The viterbi algorithm. *Proceedings of the IEEE* [online]. 1973, roč. 61, č. 3, s. 268–278. ISSN 0018-9219. Dostupné z: doi:10.1109/PROC.1973.9030

[27] EDDY, S. R. Profile hidden Markov models. *Bioinformatics (Oxford, England)*. 1998, roč. 14, č. 9, s. 755–763. ISSN 1367-4803.

[28] EDDY, S. R. *HMMER User´s Guide* [online]. B.m.: HMMER Development team. 2010. Dostupné z: ftp://selab.janelia.org/pub/software/hmmer/CURRENT/Userguide.pdf

[29] ALTSCHUL, Stephen F., John C. WOOTTON, Elena ZASLAVSKY a Yi-Kuo YU. The Construction and Use of Log-Odds Substitution Scores for Multiple Sequence Alignment. *PLoS Computational Biology* [online]. 2010, roč. 6, č. 7 [vid. 1. leden 2015]. ISSN 1553-734X. Dostupné z: doi:10.1371/journal.pcbi.1000852

[30] MENG, Xiandong a Yanqing JI. Modern Computational Techniques for the HMMER Sequence Analysis. *International Scholarly Research Notices* [online]. 2013, roč. 2013, s. e252183 [vid. 7. listopad 2014]. Dostupné z: doi:10.1155/2013/252183

[31] HUGHEY, Richard, Kevin KARPLUS a Anders KROGH. *SAM (Sequence Alignment and Modeling) documentation* [online]. 2003 [vid. 5. listopad 2014]. Dostupné z: https://compbio.soe.ucsc.edu/papers/sam_doc/sam_doc.html

[32] SADREYEV, Ruslan I., Ming TANG, Bong-Hyun KIM a Nick V. GRISHIN. COMPASS server for remote homology inference. *Nucleic Acids Research* [online]. 2007, roč. 35, č. Web Server issue, s. W653–658. ISSN 1362-4962. Dostupné z: doi:10.1093/nar/gkm293

[33] SUNYAEV, S. R., F. EISENHABER, I. V. RODCHENKOV, B. EISENHABER, V. G. TUMANYAN a E. N. KUZNETSOV. PSIC: profile extraction from sequence alignments with position-specific counts of independent observations. *Protein Engineering*. 1999, roč. 12, č. 5, s. 387–394. ISSN 0269-2139.

[34] SADREYEV, Ruslan a Nick GRISHIN. COMPASS: A Tool for Comparison of Multiple Protein Alignments with Assessment of Statistical Significance. *Journal of Molecular Biology* [online]. 2003, roč. 326, č. 1, s. 317–336 [vid. 18. listopad 2014]. ISSN 0022-2836. Dostupné z: doi:10.1016/S0022-2836(02)01371-2

[35] WANG, Yong, Ruslan I. SADREYEV a Nick V. GRISHIN. PROCAIN: protein profile comparison with assisting information. *Nucleic Acids Research* [online]. 2009, roč. 37, č. 11, s. 3522–3530. ISSN 1362-4962. Dostupné z: doi:10.1093/nar/gkp212

[36] SÖDING, Johannes. Protein homology detection by HMM–HMM comparison. *Bioinformatics* [online]. 2005, roč. 21, č. 7, s. 951–960 [vid. 9. prosinec 2014]. ISSN 1367-4803, 1460-2059. Dostupné z: doi:10.1093/bioinformatics/bti125

[37] SIEW, N., A. ELOFSSON, L. RYCHLEWSKI a D. FISCHER. MaxSub: an automated measure for the assessment of protein structure prediction quality. *Bioinformatics (Oxford, England)*. 2000, roč. 16, č. 9, s. 776–785. ISSN 1367-4803.

[38] REMMERT, Michael, Andreas BIEGERT, Andreas HAUSER a Johannes SÖDING. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nature Methods* [online]. 2012, roč. 9, č. 2, s. 173–175. ISSN 1548-7105. Dostupné z: doi:10.1038/nmeth.1818

[39] ZHANG, Yang a Jeffrey SKOLNICK. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research* [online]. 2005, roč. 33, č. 7, s. 2302–2309 [vid. 14. květen 2015]. ISSN 0305-1048. Dostupné z: doi:10.1093/nar/gki524

[40] QI, Yuan, Ruslan I. SADREYEV, Yong WANG, Bong-Hyun KIM a Nick V. GRISHIN. A comprehensive system for evaluation of remote sequence similarity detection. *BMC Bioinformatics* [online]. 2007, roč. 8, č. 1, s. 314 [vid. 13. leden 2015]. ISSN 1471-2105. Dostupné z: doi:10.1186/1471-2105-8-314

[41] FAWCETT, Tom. An introduction to ROC analysis. *Pattern Recognition Letters* [online]. 2006, roč. 27, č. 8, ROC Analysis in Pattern Recognition, s. 861–874 [vid. 14. květen 2015]. ISSN 0167-8655. Dostupné z: doi:10.1016/j.patrec.2005.10.010

[42] CESNET. *MetaCentrum NGI* [online]. 30. leden 2015 [vid. 17. květen 2015]. Dostupné z: https://www.metacentrum.cz/cs/

[43] NCBI National Centre for Biotechnology Information. *Index adresára/blast/* [online]. [vid. 18. květen 2015]. Dostupné z: ftp://ftp.ncbi.nlm.nih.gov/blast/

[44] Janelia Research Campus. *Janelia Research Campus* [online]. [vid. 18. květen 2015]. Dostupné z: https://www.janelia.org/

[45] Max Planck Institute for developmental biology. *Bioinformatics Toolkit* [online]. [vid. 18. květen 2015]. Dostupné z: http://toolkit.tuebingen.mpg.de/

[46] UCSC COMPUTATIONAL BIOLOGY GROUP. Sequence Alignment and Modeling System. *SAM:UCSC´s Sequence Alignment and Modeling System* [online]. [vid. 18. květen 2015]. Dostupné z: https://compbio.soe.ucsc.edu/sam.html

[47] HHMI.org. *The Howard Hughes Medical Institute - Science Forward* [online]. [vid. 18. květen 2015]. Dostupné z: https://www.hhmi.org/

Zoznam použitých skratiek

DNA **D**eoxyribo**N**ucleic **A**cid – deoxyribonukleová kyselina

FN **F**alse **N**egative

FP **F**alse **P**ositive

FPR **F**alse **P**ositive **R**ate

HMM **H**idden **M**arkov **M**odel – skryté Markovove modely

HSP **H**igh **S**coring **S**egment **P**airs – páry sekvencií s vysokým ohodnotením

PDB **P**rotein **D**ata **B**ank

PSSM **P**osition **S**pecific **S**coring **M**atrices – pozične špecifická skórovacia matica

RNA **R**ibo**N**ucleic **A**cid – ribonukleová kyselina

ROC **R**eceiver **O**perating **C**haracteristic

SCOP **S**tructural **C**lassification **O**f **P**roteins – databáza

TN **T**rue **N**egative

TP **T**rue **P**ositive

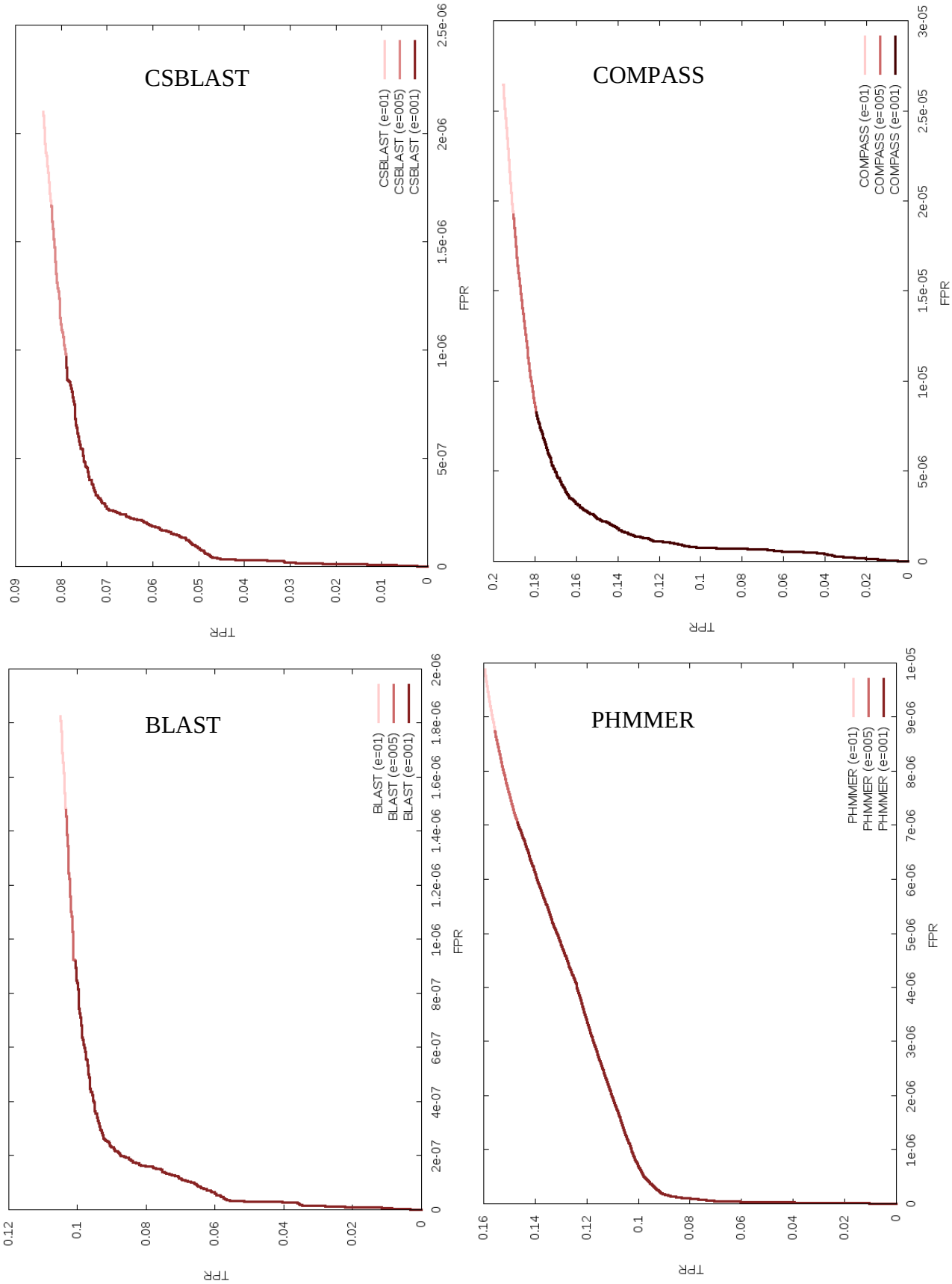
TPR **T**rue **P**ositive **R**ate

Zoznam príloh

Príloha 1. Grafy ROC priestoru
Príloha 2. USB flash disk (s virtuálnym strojom)
Príloha 3. CD (s manuálom, všetkými ROC krivkami použitými v tejto práci, zdrojovými kódmi aplikácie)

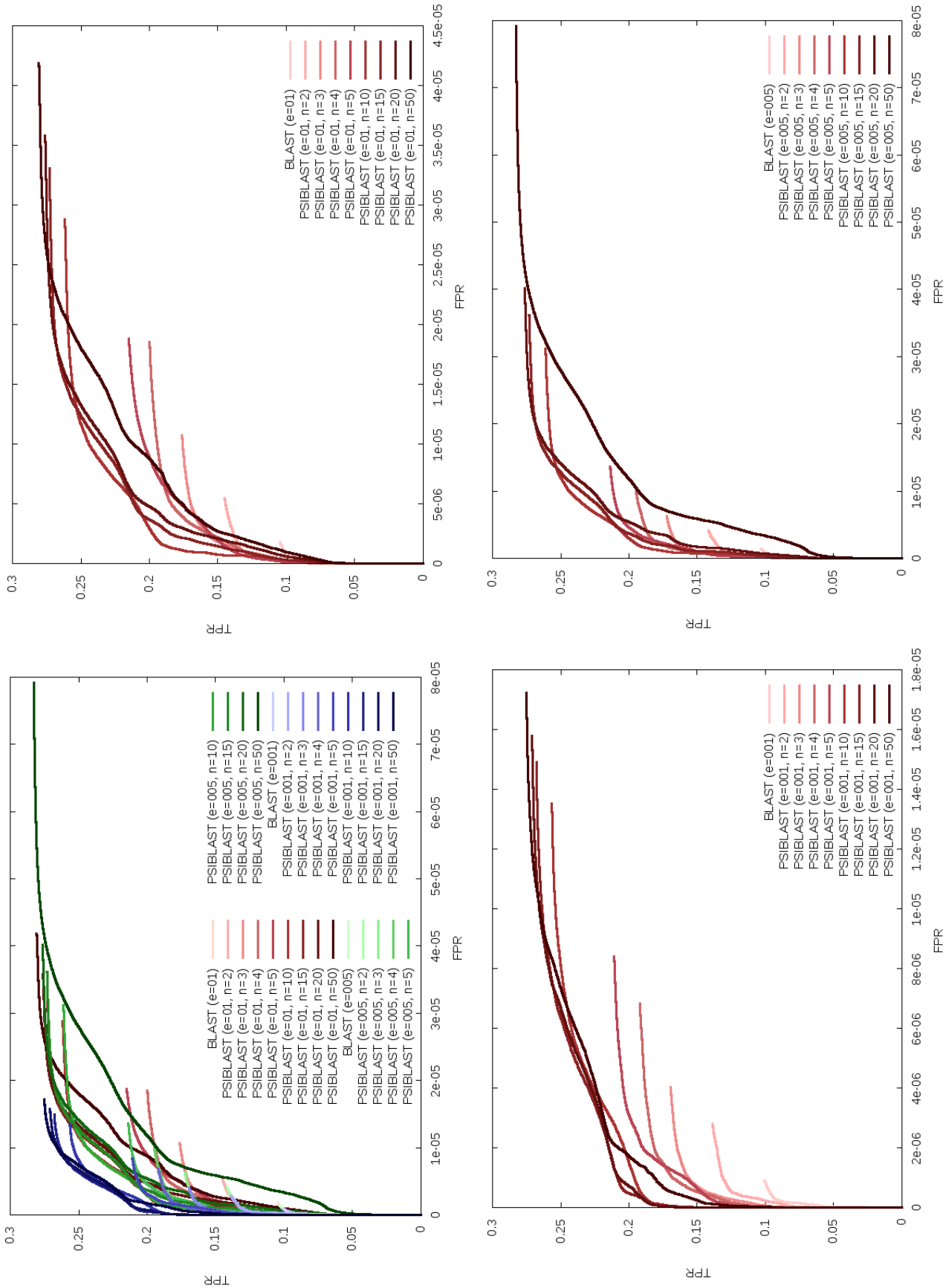
Príloha 1. Grafy ROC priestoru

Neiteratívne metódy

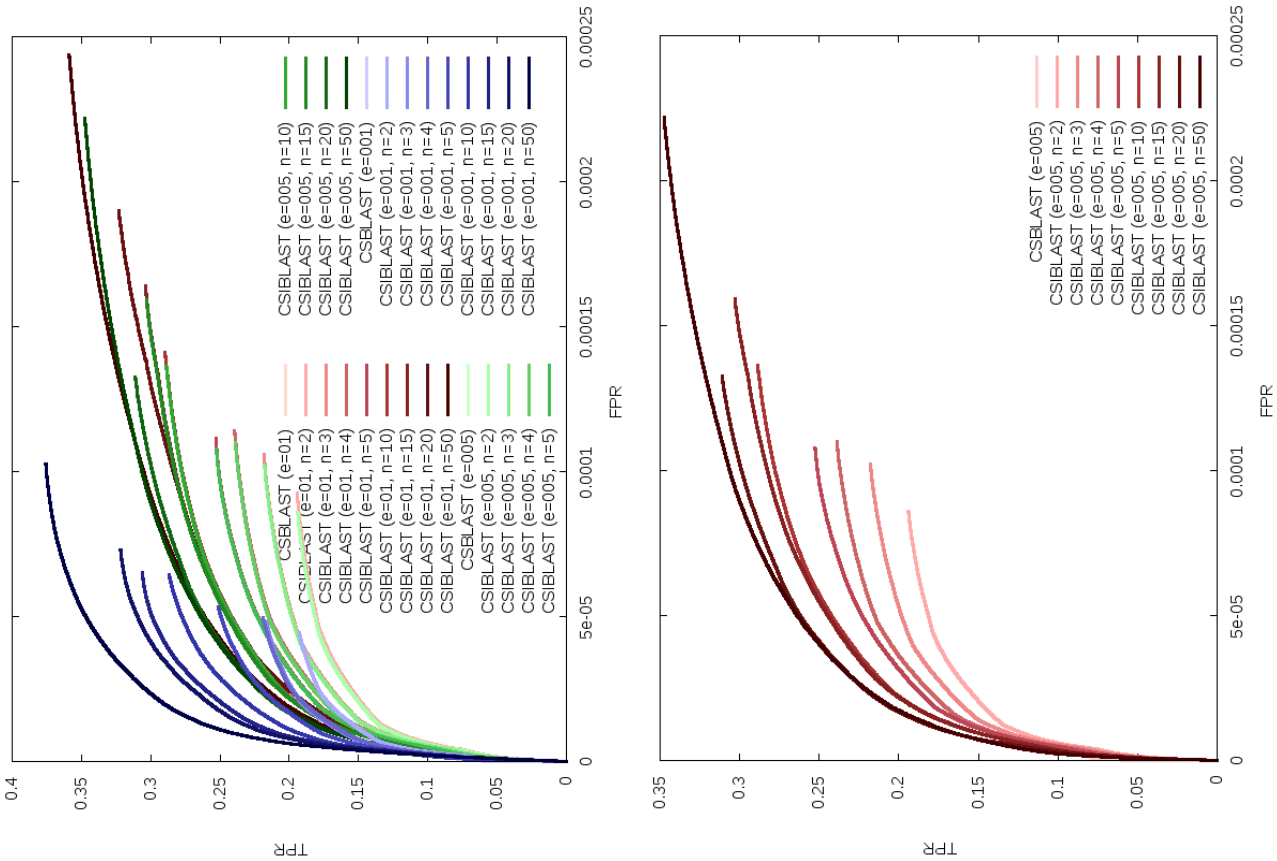


Iteratívne metódy

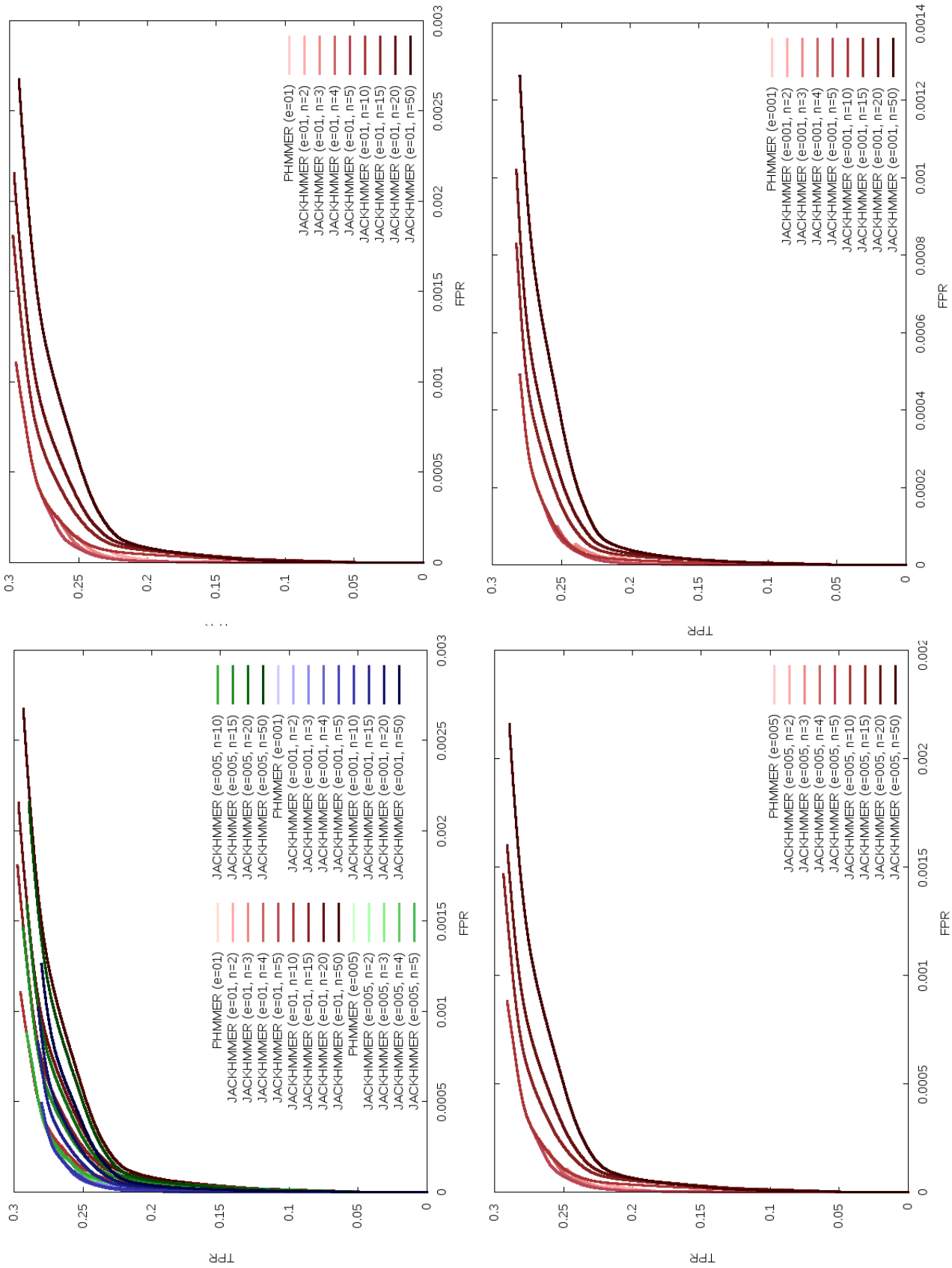
PSIBLAST



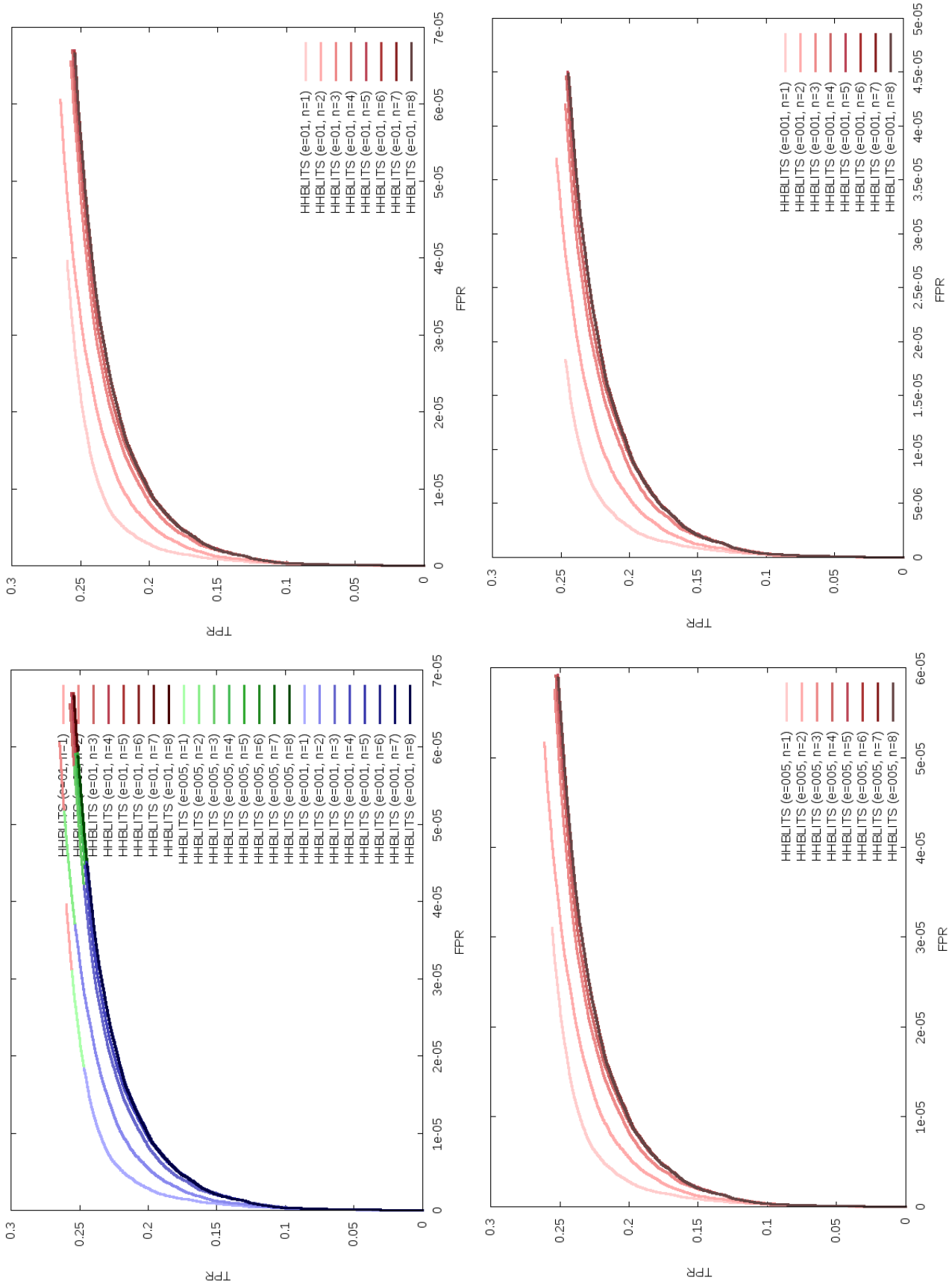
CSIBLAST


TPR
FPR
CSIBLAST (e=01)
CSIBLAST (e=01, n=2)
CSIBLAST (e=01, n=3)
CSIBLAST (e=01, n=4)
CSIBLAST (e=01, n=5)
CSIBLAST (e=01, n=10)
CSIBLAST (e=01, n=15)
CSIBLAST (e=01, n=20)
CSIBLAST (e=01, n=50)
CSIBLAST (e=001)
CSIBLAST (e=001, n=2)
CSIBLAST (e=001, n=3)
CSIBLAST (e=001, n=4)
CSIBLAST (e=001, n=5)
CSIBLAST (e=001, n=10)
CSIBLAST (e=001, n=15)
CSIBLAST (e=001, n=20)
CSIBLAST (e=001, n=50)
CSIBLAST (e=005)
CSIBLAST (e=005, n=2)
CSIBLAST (e=005, n=3)
CSIBLAST (e=005, n=4)
CSIBLAST (e=005, n=5)
CSIBLAST (e=005, n=10)
CSIBLAST (e=005, n=15)
CSIBLAST (e=005, n=20)
CSIBLAST (e=005, n=50)

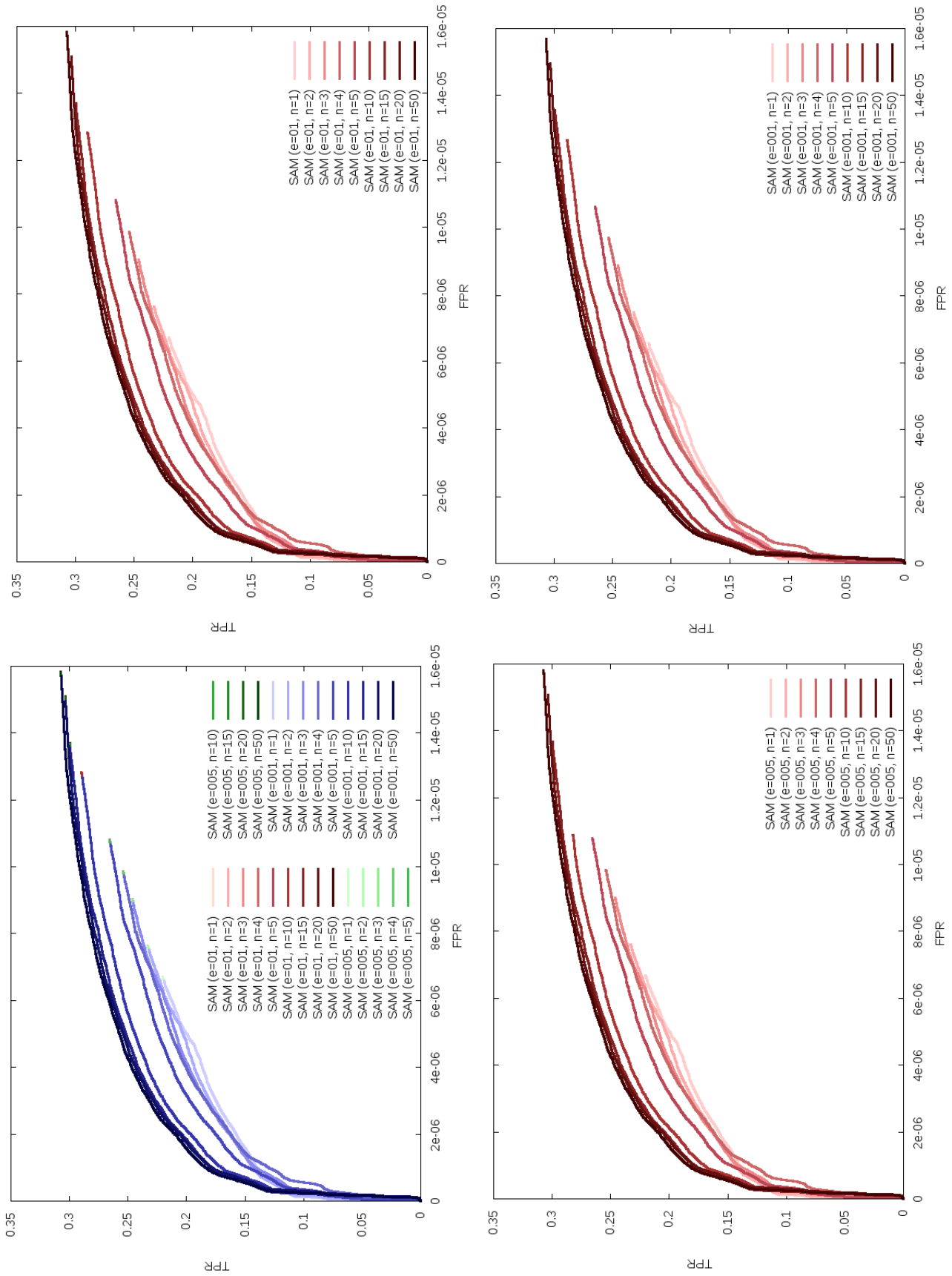
JACKHMMER


TPR
FPR
PHMMER (e=01)
JACKHMMER (e=01, n=2)
JACKHMMER (e=01, n=3)
JACKHMMER (e=01, n=4)
JACKHMMER (e=01, n=5)
JACKHMMER (e=01, n=10)
JACKHMMER (e=01, n=15)
JACKHMMER (e=01, n=20)
JACKHMMER (e=01, n=50)
PHMMER (e=001)
JACKHMMER (e=001, n=2)
JACKHMMER (e=001, n=3)
JACKHMMER (e=001, n=4)
JACKHMMER (e=001, n=5)
JACKHMMER (e=001, n=10)
JACKHMMER (e=001, n=15)
JACKHMMER (e=001, n=20)
JACKHMMER (e=001, n=50)
PHMMER (e=005)
JACKHMMER (e=005, n=2)
JACKHMMER (e=005, n=3)
JACKHMMER (e=005, n=4)
JACKHMMER (e=005, n=5)
JACKHMMER (e=005, n=10)
JACKHMMER (e=005, n=15)
JACKHMMER (e=005, n=20)
JACKHMMER (e=005, n=50)

HHBLITS


TPR
FPR
HHBLITS (e=01, n=1)
HHBLITS (e=01, n=2)
HHBLITS (e=01, n=3)
HHBLITS (e=01, n=4)
HHBLITS (e=01, n=5)
HHBLITS (e=01, n=6)
HHBLITS (e=01, n=7)
HHBLITS (e=01, n=8)
HHBLITS (e=001, n=1)
HHBLITS (e=001, n=2)
HHBLITS (e=001, n=3)
HHBLITS (e=001, n=4)
HHBLITS (e=001, n=5)
HHBLITS (e=001, n=6)
HHBLITS (e=001, n=7)
HHBLITS (e=001, n=8)
HHBLITS (e=005, n=1)
HHBLITS (e=005, n=2)
HHBLITS (e=005, n=3)
HHBLITS (e=005, n=4)
HHBLITS (e=005, n=5)
HHBLITS (e=005, n=6)
HHBLITS (e=005, n=7)
HHBLITS (e=005, n=8)

SAM


TPR
FPR
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
2e-06
4e-06
6e-06
8e-06
1e-05
1.2e-05
1.4e-05
1.6e-05
SAM (e=01, n=1)
SAM (e=01, n=2)
SAM (e=01, n=3)
SAM (e=01, n=4)
SAM (e=01, n=5)
SAM (e=01, n=10)
SAM (e=01, n=15)
SAM (e=01, n=20)
SAM (e=01, n=50)
SAM (e=001, n=1)
SAM (e=001, n=2)
SAM (e=001, n=3)
SAM (e=001, n=4)
SAM (e=001, n=5)
SAM (e=001, n=10)
SAM (e=001, n=15)
SAM (e=001, n=20)
SAM (e=001, n=50)
SAM (e=005, n=1)
SAM (e=005, n=2)
SAM (e=005, n=3)
SAM (e=005, n=4)
SAM (e=005, n=5)
SAM (e=005, n=10)
SAM (e=005, n=15)
SAM (e=005, n=20)
SAM (e=005, n=50)