VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION DEPARTMENT OF BIOMEDICAL ENGINEERING

POKROČILÉ ZAROVNÁVÁNÍ A URČOVÁNÍ GENETICKÉ ODLIŠNOSTI SEKVENCÍ DNA

DETERMINATION GENETICS DIFFERENCIES USING ALIGNMENT SIGNAL OF BIOLOGICAL SEQUENCES DNA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE ONDŘEJ TRNĚNÝ

AUTHOR

VEDOUCÍ PRÁCE Ing. MARTIN VALLA

SUPERVISOR

BRNO 2011

Abstrakt

Biologické sekvence se neustále vyvíjejí, dochází u nich k mutacím, delecím a inzercím. Z důvodu potřeby klasifikovat sekvence a stanovit míru jejich podobnosti byly vytvořeny metody pro jejich zarovnání jako jsou bodová matice nebo algoritmy Needleman-Wunsch a Smith-Waterman pro globální a lokální zarovnání. Tyto konzervativní metody jsou však omezeny na předpoklad, že přestože došlo v sekvencích ke změnám, zachovaly si malou vzdálenost mezi podobnými úseky. Proto byly vytvořeny metody pro porovnání bez zarovnání, jako je metoda znaků v sekvenci, Euklidovská vzdálenost nebo Univerzální sekvenční mapy, které se snaží nedostatky metod využívajících zarovnání eliminovat.

Klíčová slova

Matlab, rozšířené globální zarovnání, globální zarovnání, lokální zarovnání, Needleman-Wunsch, Smith-Waterman, BLOSUM, PAM

Abstract

Biological sequences are constantly evolving so there are mutations, deletions and inserts. Because of need to classify these sequences and determine degree of their similarity have been developed aligment methods. For example Dot matrix or algorithms like Needleman-Wunsch and Smith-Waterman used for global and local alignment. These methods can be considered as conservative and are limited because it is assumed that although there have been changes during evolution they still preserve small distance between similar regions. Therefore number of methods have been proposed to eliminate these limitations by comparing sequences without alignment. These methods for example Words in sequences, Euklidean distance or Universal sequence maps are designed to eliminate limitations of alignment using methods.

Key words

Matlab, generalized global alignment, global alignment, local alignment, Needleman-Wunsch, Smith-Waterman, BLOSUM, PAM

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma Pokročilé zarovnávání a určování genetické odlišnosti sekvencí DNA jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 27. května 2011 .....

podpis autora

Poděkování

Děkuji vedoucímu semestrálního projektu Ing. Martinu Vallovi. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce. Zvláštní poděkování náleží prof. Xiaoqiu Huangovi, za nezištnou pomoc při sestavování rozšířeného globálního zarovnání.

V Brně dne 27. května 2011 .....

podpis autora

Obsah

Seznam obrázků

1 Úvod

Bioinformatika je definována jako mezioborová vědní disciplína, která spojuje biologii, informatiku, matematiku a statistiku za účelem analýzy biologických dat, obsahu genů a odhalováním funkce a struktury těchto dat. V posledních letech začíná bioinformatika získávat stále větší podíl i na poli biologických a medicinských výzkumů. Tradiční informatika vždy byla doménou matematiků a inženýrů. Jejím účelem bylo shromažďování dat a jejich správa. Bioinformatika tento původní účel rozšířila na porovnávání biologických dat a genů. Tyto poznatky jsou pak dále využívány v oborech jako je farmakologie a jiné oblasti s biologickými daty související. Typickým příkladem dat zpracovávaných v bioinformatice jsou sekvence proteinů a nukleových kyselin. Přestože má bioinformatika široké pole působnosti, stále základním tématem pro bioinformatiku zůstává sekvenční analýza a následné stanovení míry podobnosti zkoumaných sekvencí.

Oblastí příbuznou k bioinformatice je výpočetní biologie, která se také částečně zabývá využitím výpočetní techniky pro některé z výše popsaných účelů. Přesto se více zaměřuje na problematiku vývoje nových algoritmů pro řešení problémů, které mohou v biologii nastat. Bioinformatika se více soustředí na vývoj praktických nástrojů pro správu a analýzu dat.

2 Zarovnání

Zarovnání dvou sekvencí představuje shodu znaků na stejné pozici zarovnávaných sekvencích. Správné zarovnání nukleotidů nebo aminokyselin v sekvencích reprezentuje evoluční příbuznost mezi dvěma nebo více sekvencemi, které vycházejí ze stejného základu.

Obecně se dá říci, že pro určení podobnosti analyzovaných sekvencí jsou přiloženy pod sebe tak, aby obě měly stejně umístěný začátek. Tento postup je nazýván přiřazením. Následně je určena (vypočtena) celková hodnota podobnosti tzv. skóre (score). A to pro každou dvojici vzniklou přiřazením zvlášť. Tyto hodnoty se odvíjejí od předem daných pravidel. Při nejjednodušším stanovení score mohou nastat dvě varianty – shoda (match) a neshoda (mismatch).

Pro přiřazení nestejně dlouhých nebo příliš odlišných sekvencí nelze aplikovat předchozí zjednodušený postup. Je nutné tedy stanovit postupy pro případy, kdy nelze jednoduše určit shodu nebo neshodu, např. když je v jedné sekvencí více znaků než ve druhé. Pro tyto případy jsou určeny dvě metody – globální a lokální zarovnání. Globální zarovnání dvě odlišné sekvence přiřadí po celé jejich délce a to i za cenu vnášení mezer. Lokální zarovnání skončí s přiřazováním v okamžiku, kdy se sekvence začínají příliš lišit.[4]

Globální přiřazení je vhodné pro případy, kdy od sebe nejsou sekvence příliš evolučně vzdáleny. Pokud je použito pro nevhodný pár sekvencí, může docházet k nežádoucím chybám.[4]

Vhodnou metodou volby, zda použít globální nebo lokální zarovnání, se jeví bodový diagram (nebo také dotplot). Bodový diagram poskytuje možnost odhadnout, zda je smysluplné se snažit o globální zarovnání.[4]

2.1 Zarovnání bez mezer

Nejjednodušším případem zarovnání dvou sekvencí je metoda, kdy není povoleno vkládat mezery do zarovnání. Zarovnání je tedy záležitostí zvolení počátečního místa přiložení kratší z obou sekvencí.

Sekvence 1: ATCATACG Sekvence 2: GATACG

Tři případy možného zarovnání:

ATCATACG	ATCATACG	ATCATACG
GATACG	GATACG	GATACG

Obrázek 1 - Možné případy zarovnání bez mezer[11]

2.2 Zarovnání s využitím mezer

Vkládání mezer (gaps) při zarovnání reprezentuje možnost inzercí a delecí, což zvyšuje počet možných zarovnání u porovnávaných sekvencí.

Sekvence 1: AATCTATA Sekvence 2: AAGATA

Tři případy možného zarovnání:

AATCTATA	AATCTATA	AATCTATA
AA-G-ATA	AAG-AT- A	AA- -GATA

Obrázek 2 - možná zarovnání s vložením mezer[11]

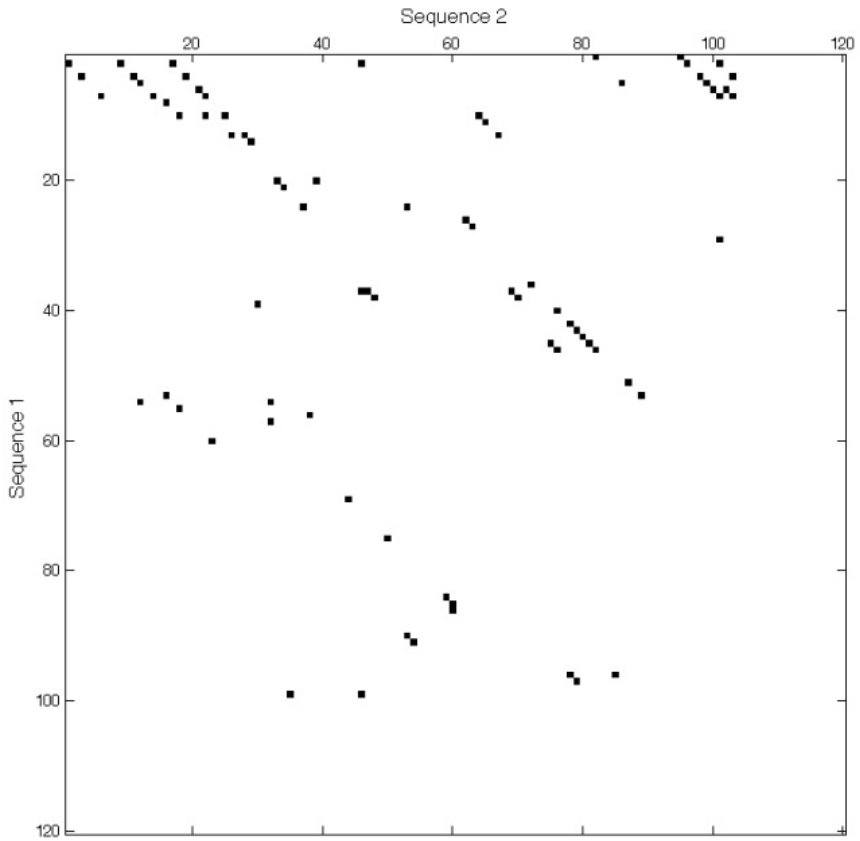
S použitím mezer je nutné u výpočtu skóre zohlednit tzv. penalizaci mezer (gap penalty). Penalizace za mezeru musí být započítána do skóre.[11]

$$\sum_{i=1}^{n} \begin{cases} Sekvence\ 1_i = ' - ' nebo\ sekvence\ 2_i = ' - ' \ldots penalizace\ mezery \\ Sekvence\ 1_i = \ sekvence\ 2_i \ldots hodnota\ shody \\ Sekvence\ 1_i \neq nebo\ sekvence\ 2_i \ldots hodnota\ neshody \end{cases}$$

2.3 Bodový diagram

Bodový diagram představuje jednu z nejjednodušších metod pro porovnávání podobností mezi dvěma sekvencemi. Tato metoda umožňuje zobrazit úseky podobností mezi oběma sekvencemi. Bodový diagram představuje vhodnou pomůcku pro zvolení typu přiřazení, které

bude použito k porovnání sekvencí. Sestavení diagramu je poměrně jednoduché. Porovnávané sekvence se vynesou na osy X a Y. Poté je zvoleno okno vhodné velikosti, které klouže po sekvenci s krokem o velikosti 1. Tento postup může velmi rychle znepřehlednit výsledek v případě, že jsou porovnávány dvě dlouhé a velmi podobné sekvence. Vhodně zvolená délka okna redukuje nežádoucí šum, který u dlouhých podobných sekvencí vzniká. Pokud se v oknu nachází shoda mezi písmenem na ose X a na ose Y, tak do grafu vyznačíme tečku (odtud dotplot). Z takto vzniklých diagonál se vezmou pouze ty, které jsou delší, než předem stanovená hodnota. Výsledné diagonály poukazují na přítomnost podobnosti sekvencí, eventuelně lze stanovit i zda proběhla například delece či inzerce.[4]



Obrázek 3. - Bodový diagram dvou sekvencí s klouzavým oknem o délce 20

2.4 Skórovací matice

Jakmile je sestaveno zarovnání všech možných párů nukleotidů, použije se k ohodnocení každé pozice, která neobsahuje mezeru, skórovací matice V případě zarovnání sekvencí nukleotidů jsou skórovací matice primitivní. Jedna z používaných skórovací maticí, je matice programu BLAST určeného k zarovnávání nukleotidů.

	A	T	C	G
A	5	-4	-4	-4
T	-4	5	-4	-4
C	-4	-4	5	-4
G	-4	-4	-4	5

Obrázek 4 - Skórovací matice BLAST[11]

Tato skórovací matice udává, že v případě nalezení shodného páru nukleotidů přiřazuje skóre +5, v případě neshody pak skóre -4.

Jiné typy skórovací matice jsou založeny na frekvenci výskytu jednotlivých aminokyselin v přírodě. V případě, že se dvojice aminokyselin vyskytuje častěji, je toto zohledněno a nalezení páru těchto aminokyselin vyústí v příznivější skóre. Naopak nalezení páru aminokyselin, jejichž výskyt není v přírodě příliš častý, končí penalizací. Příkladem takto sestavované skórovací matice je třeba PAM (Point accepted mutation). Tato matice byla sestavena Margaret Dayhoff v roce 1970. Existuje více typů matic PAM, které jsou odlišeny číslem. Toto číslo udává, jaká míra substituce se dá očekávat, jestliže určité procento aminokyselin podlehlo změnám. V případě PAM1 se uvažuje, že změny nastaly u 1 % aminokyselin. Nasledně odvozené matice končí u PAM250. V praxi jsou používány nejčastěji PAM30 a PAM70. [8][9]

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	4															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-4	-4	-4	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	-1	-1	-3	0	-1	-1	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	3				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-2	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	2	4

Obrázek 5 - matice PAM250[13]

Matice PAM jsou vhodné pro příbuzné druhy, na druhou stranu nejsou vhodné pro evolučně vzdálené sekvence. V průběhu mnoha let došlo k evolučním změnám na sekvencích a proto bylo potřeba nalézt řešení, které by tyto odchylky zohledňovalo. Tímto řešením je matice BLOSUM(BLOck Substitution Matrix). BLOSUM byla vytvořena na základě vícenásobných zarovnání evolučně odlišných sekvencí. BLOSUM je založena na myšlence pravděpodobnosti zachování určitých úseků na sekvencích, které odolaly vývoji a byly nalezeny prostřednictvím vícenásobného zarovnání. Stejně jako matice PAM jsou jednotlivé matice BLOSUM označeny číslem. Například matice BLOSUM62 reprezentuje případ, kdy více jak 62 % sekvencí bylo identických.[8][9]

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0	-2	-1	0	-4
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3	-1	0	-1	-4
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3	3	0	-1	-4
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3	4	1	-1	-4
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1	-3	-3	-2	-4
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2	0	3	-1	-4
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3	-1	-2	-1	-4
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3	0	0	-1	-4
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3	-3	-3	-1	-4
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1	-4	-3	-1	-4
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2	0	1	-1	-4
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1	-3	-1	-1	-4
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1	-3	-3	-1	-4
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2	-2	-1	-2	-4
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2	0	0	0	-4
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0	-1	-1	0	-4
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3	-4	-3	-2	-4
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1	-3	-2	-1	-4
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4	-3	-2	-1	-4
B	-2	-1	3	4	-3	0	1	-1	0	-3	-4	0	-3	-3	-2	0	-1	-4	-3	-3	4	1	-1	-4
Z	-1	0	0	1	-3	3	4	-2	0	-3	-3	1	-1	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
X	0	-1	-1	-1	-2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-2	0	0	-2	-1	-1	-1	-1	-1	-4
*	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	1

Obrázek 6 - matice BLOSUM62[14]

2.5 Needleman-Wunsch

Needleman-Wunsch je algoritmus pro globální zarovnání dvou sekvencí - tedy zarovnání, které bere v potaz celou sekvenci. Standardně je používán v bioinformatice k zarovnání proteinových nebo nukleotidových sekvencí. Algoritmus byl představen poprvé v roce 1970. Výsledné zarovnání a hodnota jeho skóre je vypočítávána z podobnostní matice. Algoritmus Needleman-Wunsch postupuje při zarovnání sekvencí podle tří základních kroků: inicializace skórovací matice, výpočet skóre a vyplnění matice zarovnání s vyznačením cesty zpětného průchodu tzv. zpětného trasování a zarovnání sekvencí podle matice zarovnání. [1][8][3]

Jako jednoduchý příklad je možno uvést zarovnání dvou sekvencí:

$$Sekvence\ 1: GAATT$$

$$Sekvence\ 2: GGAT$$

	A	T	C	G
A	1	0	0	0
T	0	1	0	0
C	0	0	1	0
G	0	0	0	1

Obrázek 7. – Skórovací matice[11]

$$penalizace\ za\ otevření\ mezery(gap) = 0$$

-	-	G	A	A	T	T
-	0	0	0	0	0	0
G	0	1	1	1	1	1
G	0	1	1	1	1	1
A	0	1	2	2	2	2
T	0	1	2	2	3	3

Obrázek 8. –Matice Needleman – Wunsch

Cesta zpět je vyznačena modře(viz. Obrázek 4).

Výsledné zarovnání:

$$\begin{array}{ccccc} G & A & A & T & T \\ | & & | & & | \\ G & G & A & - & T \end{array}$$

$$Skóre\ je\ 3.$$

2.6 Smith-Waterman

Jedná se o algoritmus používaný pro lokální zarovnání. Lokální zarovnání přeruší přiřazování v okamžiku, kdy se od sebe dané úseky začnou příliš lišit, resp. oblasti na sekvenci, které jsou příliš vzdálené od úseků podobnosti, nejsou brány v úvahu. Příslušnému úseku porovnávaných sekvencí se vypočítá dosažené skóre. Algoritmus postupuje při porovnávání po jednotlivých dvojicích. Každé dvojici je přiřazeno skóre. Následně po zarovnání je ve výsledné matici nalezen pár s nejvyšším skóre. Dalším krokem je postup zpět, čímž se získá

cesta, jakou bylo tohoto nejvyššího skóre dosaženo, resp. z jakých hodnot bylo vypočteno. Po vykonání předchozího kroku je tedy zjištěno, na jakých úsecích bylo dosaženo nejlepšího přiřazení. Na rozdíl od algoritmu Needleman-Wunsch vyhledává lokální zarovnání pouze ty nejpodobnější úseky sekvencí a oblasti s nimi nesouvisející jsou zanedbány.[1][4][6]

Lokální zarovnání dvou sekvencí:

$$Sekvence\ 1: ATCAG$$

$$Sekvence\ 2: GTCAG$$

	A	C	G	T
A	1	-1	-1	-1
C	-1	1	-1	-1
G	-1	-1	1	-1
T	-1	-1	-1	1

Obrázek 9. – Skórovací matice

$$penalizace\ za\ otevření\ mezery(gap) = -2$$

-	-	A	T	C	A	G
-	0	0	0	0	0	0
G	0	0	0	0	0	1
T	0	0	1	0	0	0
C	0	0	0	2	0	0
A	0	1	0	0	3	1
G	0	0	0	0	1	4

Obrázek 10. – Matice lokálního zarovnání algoritmem Smith-Waterman

Výsledné zarovnání:

$$\begin{array}{cccc} T & C & A & G \\ | & | & | & | \\ T & C & A & G \end{array}$$

$$Skóre\ je\ 4.$$

3 Vícenásobné zarovnání

Zarovnání více sekvencí představuje cestu k porovnání více než dvou sekvencí v jednom okamžiku. Problematika vícenásobného zarovnání zahrnuje několik problémů, z nichž největší představuje stanovení standardu pro správné zarovnání resp. lepší zarovnání od toho nesprávného. Jelikož se sekvence vyvíjejí a jejich předchůdci mohou být rekonstruováni jen teoreticky, je téměř nemožné experimentálně sestavené zarovnání ověřit. Prosté přiřazení skóre potom může vést k nesprávným závěrům ohledně optimálního zarovnání porovnávaných sekvencí.

Mimo problematiky správné definice skóre zarovnání existuje potřeba algoritmu, který by provedl vícenásobné zarovnání s nejlepším skóre. Založení takového algoritmu na metodě dynamického programování, je teoreticky popsáno, ale jeho realizace resp. jeho činnost by byla velmi časově náročná při zarovnání pouhých deseti sekvencí.

Pro vícenásobné zarovnání sekvencí nukleotidů a proteinů existuje v současné době program ClustalW. Zarovnání mohou být globální (zahrnují celou sekvenci) nebo lokální (zaměřené na lokální místa podobnosti). ClustalW vypočítává nejlepší shodu pro zvolené sekvence a řadí je tak, že mohou být identifikovány podobnosti a rozdíly mezi nimi.[7][8][10]

```
SeqA Name                            Len(aa)  SeqB Name                            Len(aa)  Score
==========================================================================================
1    gi|31317218|ref|NP_851850.1|    388      2    gi|66346654|ref|NP_001017528.1  379      100
1    gi|31317218|ref|NP_851850.1|    388      3    gi|66346656|ref|NP_001017529.1  293      100
2    gi|66346654|ref|NP_001017528.1  379      3    gi|66346656|ref|NP_001017529.1  293      100
==========================================================================================

     gi|31317218|ref|NP_851850.1|      MRTLRRLKFMSSPSLSDLGKREPAAAADERGTQQRRACANATWNSIHNGV 50
     gi|66346656|ref|NP_001017529.1    --------------------------------------------------
     gi|66346654|ref|NP_001017528.1    ---------MSSPSLSDLGKREPAAAADERGTQQRRACANATWNSIHNGV 41

     gi|31317218|ref|NP_851850.1|      IAVFQRKGLPDQELFSLNEGVRQLLKTELGSFFTEYLQNQLLTKGMVILR 100
     gi|66346656|ref|NP_001017529.1    ---------------------------------------------MVILR 5
     gi|66346654|ref|NP_001017528.1    IAVFQRKGLPDQELFSLNEGVRQLLKTELGSFFTEYLQNQLLTKGMVILR 91
                                                                                 *****

     gi|31317218|ref|NP_851850.1|      DKIRFYEGQKLLDSLAETWDFFFSDVLPMLQAIFYPVQGKEPSVRQLALL 150
     gi|66346656|ref|NP_001017529.1    DKIRFYEGQKLLDSLAETWDFFFSDVLPMLQAIFYPVQGKEPSVRQLALL 55
     gi|66346654|ref|NP_001017528.1    DKIRFYEGQKLLDSLAETWDFFFSDVLPMLQAIFYPVQGKEPSVRQLALL 141
                                       **************************************************
```

Obrázek 111. Zarovnání programem ClustalW[12]

4 Metody porovnání bez algoritmů zarovnání

Naprostá většina práce na metodách porovnání bez algoritmů zarovnání (také alignment free) probíhala v minulých dvou dekádách. Pro aligment free metody byly definovány dva hlavní směry - postupy založené na frekvenci znaků a postupy, jejichž řešení nevyžaduje pevně stanovenou délku znaků. První kategorie je založena na statistikách četnosti znaků, na vzdálenostech, které jsou definovany souřadnicovým systémem vymezenou četností výskytu a na informačním obsahu četnosti rozložení. Druhá kategorie zahrnuje využití teorie chaosu a Kolmogorovy komplexnosti. Navzdory jejich obtížnému rozpoznání jsou aligment free měřítka často používána pro filtry před samotným zarovnáním - což představuje široké pole upotřebení. Nedávné publikace rozšiřují jejich využití jako nezávislou metodiku pro rozpoznání shody v sekvenci, kde je jejich vzdálenost mimo možnosti standardního zarovnání. [1]

4.1 Znaky v sekvencích

Sekvence X o délce n je definována jako lineární posloupnost znaků z abecedy A o délce r. A se skládá z L znaků, kde $L \leq n$ je nazýváno $L - tic$í. Množina W_L se skládá, ze všech $L - tic$ v sekvenci X a má K prvků.

$$W_L = \{w_{L,1}, w_{L,2}, \dots, w_{L.K}\}$$

$$K = r^L \tag{4.1}$$

Výpočetně se hledání $L - tic$ realizuje jako klouzavé okno o délce L, které se pohybuje po sekvenci od pozice 1 k n-L+1

$$C_L^X = (C_{L,1}^X, \dots, C_{L,K}^X) \tag{4.2}$$

Následně je možné vypočítat i frekvenci slov, f_L^X, která umožňuje stanovit pravděpodobnost $p_{L,i}^X$, nalezení specifického slova $w_{L,i}^X$, který souhrně definuje vektor pravděpodobnosti $L - tice$.

$$P_L^X = (P_{L1}^X, P_{L,2}^X, \dots, P_{L,K}^X) \tag{4.3}$$

Vektor frekvence f_L^X je vyjádřen jako relativní četnost každého slova.

$$f_L^X = \frac{C_L^X}{\sum_{J=1}^{K} C_{L,j}^X} \leftrightarrow f_{L,j}^X = \frac{C_{L,i}^X}{n - L + 1} \tag{4.4}$$

Například pro DNA sekvence, $A = \{A, T, C, G\}$, $r = 4$. Třípísmenné slovo, $L = 3$, což může být $w_3 = ATC$. Pro sekvenci $X = ATATAC$, kde $n = 6$, je vektor p_3^X určen relativní četností všech trinukleotidů. Četnosti určené klouzajícím oknem o délce tří znaků $n - L + 1 = 4$ by byly tyto:

$$W_3 = \{AT\ A, T\ AT, T\ AC, AAA, \dots\}$$

$$C_3^X = (2, 1, 1, 0, \dots)$$

$$f_3^X = (0.5, 0.25, 0.25, 0, \dots)$$

(4.5)

Vektory c_3^X a f_3^X mají délku $K = 4^3 = 64$, nulové souřadnice odpovídají chybějícím slovům v X, v tomto případě se jedná o chybějící trinukleotidy.[1][7][10]

4.2 Vzdálenost mezi sekvencemi

Vzdálenost mezi sekvencemi je určena funkcí d. Tato je definována jako $d(X, Y)$, funkce která přiřadí reálné číslo, každému páru X a Y patřícímu do příslušné množiny. Díky požadavku na vzdálenost, která musí být metrická. Proto je nutné splnit tyto podmínky. [1]

$$Kladnost : d(X, Y) \geq 0\ a\ d(X, Y) = 0 \leftrightarrow X = Y$$

$$Symetrie : d(X, Y) = d(Y, X)$$

$$Trojúhelníkový\ nepoměr : d(X, Y) + d(Y, Z) \geq d(X, Z)$$

4.3 Teorie informace

Poprvé byla teorie informace formulována Claudem Shannonem roku 1948. Teorie informace matematicky popisuje vlastnosti a parametry ovlivňující přenos informací komunikačními kanály. Hlavní myšlenkou je představa entropie nebo neurčitosti. Je definována jako entropie libovolné proměnné založené na pravděpodobnosti všech možných výsledků. Tato definice je následovně aplikována na sekvence. Proměnné představují $L - tice$. Entropie těchto $L - tic, H$ je označena W_L a její hodnota je získána výpočtem z pravděpodobnosti jednotlivých slov v sekvenci X. [1][7]

$$H(W_L^X) = -\sum_{i=1}^{K} p_{L,i}^X \, log_2\left(p_{L,i}^X\right) \tag{4.6}$$

Tato definice je platná pro slovo jakékoliv délky L. $H(W)$ nabývá maximálních hodnot, když všechna možná K všech slov jsou stejně pravděpodobná.

$$H(W) = log_2(K) \tag{4.7}$$

A je minimální, když $p_{L,i}^X = 1$ pro některá i-slova. Bylo zjištěno, že výstup učiní pravděpodobnost rovnou nule. [1][7]

4.4 Euklidovská vzdálenost

První systematická zpráva o použití součtu L-tic k sekvenční analýze pochází z roku 1986 (Blaisdell, 1986). Odlišnost dvou sekvencí, která byla modelována jako Markovův řetězec, byla v Blasidellově práci vypočítána jako mocnina Euklidovské vzdálenosti mezi jejich přechodovými maticemi. Přestože je tato metoda jednoduchá, představuje poměrně účinnou efektivní variantu k zarovnávacím metodám. [1][10]

Skutečnost, že přechodová matice Markovova řetězce může být nalezena s využitím frekvence všech L-tic, vyústila ve stanovení dalších kritérií pro podobnost sekvencí. Pro každou délku znaků L je mocnina Euklidovské vzdálenosti mezi sekvencemi X a Y definována rovnicí, kde $C_L^X = (\ C_{L,1}^X, \dots,\ C_{L,K}^X)$ a $C_L^Y = (\ C_{L,1}^Y, \dots,\ C_{L,K}^Y)$ jsou vektory představující počet znaků v těchto sekvencích a K je počet lišících se L-tic uvažovaných pro délku L.[1][10]

$$d_L^E(X,Y) = (\ C_L^X - \ C_L^Y)^T \cdot (\ C_L^X - \ C_L^Y) \ = \sum_{i=1}^{K}(\ C_{L,i}^X - C_{L,i}^Y)^2 \tag{3.8}$$

O tři roky později Blaisdell (1989) stanovil nová měřítka pro porovnání bez zarovnání a následně prokázal jejich funkčnost při porovnání rozsáhlých genomických sekvencí, které již byly po stránce fylogenetické příbuznosti dobře prozkoumány. Následná zpráva představila několik nových měřítek zjištěných na základě statistických výpočtů. Byly formulovány filtrační metody založené na předběžném porovnání s těmito novými měřítky. Zjistilo se, že je možné vyfiltrovat sekvence s nízkou podobností, tedy ty které nesdílejí stejnou stavbu znaků. Tato předfiltrace urychlila prohledávání databází, které se stále více rozšiřují o nové informace. [1][10]

4.5 Váhovaná Euklidovská vzdálenost

Výše popsaná Euklidovská vzdálenost má i svá omezení. Rozdílná frekvence výskytu znaků u specifických částí sekvencí rozdílně ovlivňuje Euklidovskou vzdálenost, což vedlo k vytvoření váhovaných měřítek. Prvotní formulace byla založena na počtu váhovaných L-tic. Odtud tedy váhované vzdálenosti jsou kombinovány sečtením váhovaného součtu odlišnosti a rozpoznání odlišnosti , od *l* k *u-ticím*. [1][10]

$$d^2(X,Y) = \sum_{L=l}^{u}\sum_{i=1}^{K}\rho_i(C_{L,i}^X - \ C_{L,i}^Y)^2 \tag{4.9}$$

Označení pro tuto formulaci je $d^2\ vzdálenost$, následně od ní byla odvozena neváhovaná varianta. Tato byla úspěšně použita jako velmi výkonný postup pro rychlé porovnání sekvencí z obsáhlých databází (Hide et al., 1994). Byl proveden pokus s vyhledáváním lipáz v genomické databázi a pro délku L=8 bylo dosaženo totožných výsledků jako při vyhledávání algoritmem FASTA. [1][10]

Při praktickém použití $d^2\ vzdálenosti$ se tato metoda jeví jako vysoce výkonná, sensitivní a selektivní. Tyto vlastnosti ve spojení s ostatními výhodami aligment free metod, kterými jsou nezávislost na obsahu analyzovaných sekvencí a generování d^2 hodnoty přes inzerce a delece v sekvenci, vyústily v zapracování této metody do softwarových balíčků. Jedním z nich je i softvarový balík STACK (Sequence Tag Alignment and Consensus Knowledgebase). [1]

4.6 Korelační struktura

Poté, co byl formulován převod sekvencí na četnost $L-tic$, byla pro ně velmi rychle stanovena měřítka. Pro metrickou vzdálenost mezi sekvencemi byla založena na korelačních koeficientech. Výpočet lineárního korelačního koeficientu (LKK) mezi dvěma sekvencemi Xa Y z četnosti $L-tic$, f_L^X a f_L^Yje založen na Pearsonovském přístupu. [1]

$$d_L^{LKK}(X,Y) = \left[K \sum_{i=1}^{K} f_{L,i}^X \cdot f_{L,i}^Y - \sum_{i=1}^{K} f_{L,i}^X \cdot \sum_{i=1}^{K} f_{L,i}^Y\right] / \left[\left[K \sum_{i=1}^{K} \left(f_{L,i}^X\right)^2 - \left(\sum_{i=1}^{K} f_{L,i}^X\right)^2\right]^{1/2} \times \left[K \sum_{i=1}^{K} \left(f_{L,i}^Y\right)^2 - \left(\sum_{i=1}^{K} f_{L,i}^Y\right)^2\right]^{1/2}\right] \quad (4.10)$$

Což může být jednoduše vyjádřeno jako použití vektorů f_L^Xa f_L^Y jako pár v R^2, vynesením K bodů $\left(f_{2,i}^X, f_{2,i}^Y\right)$ do grafu a výpočtem korelačního koeficientu R.

Stejně jako pro Euklidovskou vzdálenost platí, že metody porovnání založené na korelaci jsou velmi vhodné pro prohledávání rozsáhlých databází dat a byly použity pro vyhledávání v proteinových databázích (Petrilli a Tonukari, 1997). Toto použití vyústilo v řadu zjednodušujících poznatků, které značně obohatily jejich praktický význam. Například pouze 25 ze 400 možných dipeptidových četností stačilo ke správnému zařazení proteinových rodin. [1]

4.7 Kovarianční metody

Kovarianční metody představují použití Euklidovské vzdálenosti a korelací mezi zastoupením L-tic v sekvencích. Hlavní slovo zde má Mahalanobisova vzdálenost a standardizovaná Euklidovská vzdálenost. [1]

$$d_L^M(X,Y) = (c_L^X - C_L^Y)^T \cdot S^{-1} \cdot (C_L^X - C_L^Y)$$

$$= \sum_{i=1}^{K} \sum_{j=1}^{K} \left(C_{L,i}^X - C_{L,i}^Y\right) \cdot s_{ij}^{inv} \cdot \left(C_{L,j}^X - C_{L,j}^Y\right)$$

(4.11)

Rovnice výše se skládá z těchto prvků: $S = \left[s_{ij}\right]$ představuje kovarianční matici počtu L-tic, která se po převrácení skládá z $K \times K$ elementů s_{ij}^{inv}. Standardní Euklidovská vzdálenost představuje $\mathrm{cov}\left(c_i, c_j\right) = 0 \; pro \; i \neq j$. Proto jsou v tomto výpočtu vzdálenosti ignorovány korelace mezi rozdílnými slovy a pouze variace stejných slov jsou spočítány. [1]

$$d_L^{SE}(X,Y) = (c_L^X - c_L^Y)^T \cdot \; [diag(s_{11}, \dots, s_{KK})]^{-1}$$

$$\cdot (c_L^X - c_L^Y) = \sum_{i=1}^{K} \frac{\left(c_{L,i}^X - c_{L,i}^Y\right)}{s_{ii}}$$

(4.12)

Důležitost tohoto zjednodušení je zřejmá v kontextu se omezením standardní Euklidovské vzdálenosti (4.12) na mocninou Euklidovskou vzdálenost (4.8) v případě, že vynecháme variaci struktury, např. $s_{ii} = 1, \; i = 1, \dots, K$. Jak Mahalanobisova, tak standardní Euklidovská vzdálenost byly pro sekvenční analýzu použity teprve nedávno. [1]

4.8 Měřítka založená na teorii informace

Jedná se o určení rozdílnosti mezi sekvencemi založené používá vektory L-tic, ale v tomto případě vychází jejich metrická soustava z teorie informace. K dosažení výše uvedeného bylo navrženo použití Kullbyck-Leiblerovy odchylky. KL odchylka mezi sekvencemi X a Y, je vypočítána z četnosti L-tic. [1]

$$d_L^{KL}(X,Y) = \sum_{i=1}^{K} f_{L,i}^X \cdot \log_2\left(\frac{f_{L,i}^X}{f_{L,i}^Y}\right)$$

(4.13)

Aby $d_L^K(X,Y)$ nebylo nekonečné, v případě že $f_{L,i}^Y = 0$, je doporučeno upravit rovnici (4.13) přidáním jednotky k oběma proměnným četnosti. [1]

4.9 Úhlový metrický systém

Nedávno byl představen nový systém (Stuart, et al., 2002a,b), kde vzdálenost mezi sekvencemi je založena na úhlu mezi dvěma vektory součtu L-tic, rovnice (4.13). Protože tyto vektory mají většinou velký rozměr ($K = r^L$) je použit analytický rozklad podle singulární hodnoty (SVD) před samotným výpočtem cosinového úhlu. Jsou použity pouze rozměry s vyšší vlastní hodnotou, tímto podstatným zmenšením rozměru je navíc odstraněn šum z této informace. [1]

$$d_L^{cos}(X,Y) = \theta_{XY}, \text{kde} \cos(\theta_{XY}) = \frac{(c_L^X)^T \cdot c_L^Y}{\|c_L^X\| \cdot \|c_L^Y\|}$$

$$= \frac{\sum_{i=1}^{n} c_{L,i}^X \cdot c_{L,i}^Y}{\sqrt{\sum_{i=1}^{n}\left(c_{L,i}^X\right)^2} \cdot \sqrt{\sum_{j=1}^{n}\left(c_{L,j}^Y\right)^2}}$$

(4.14)

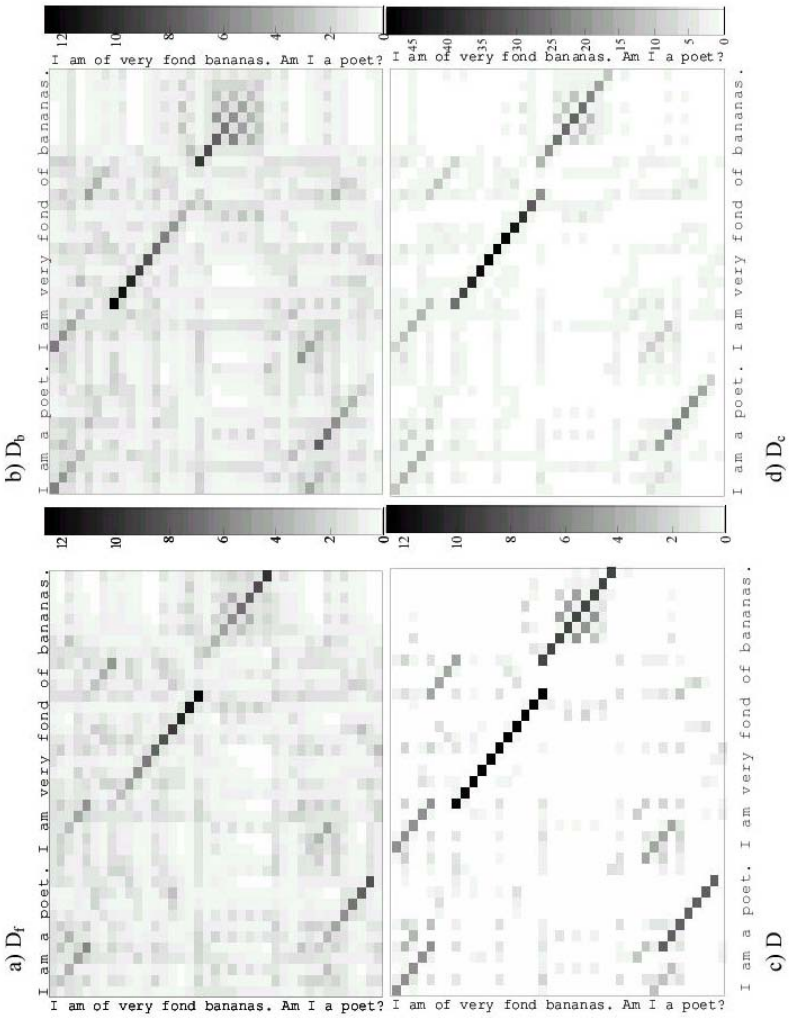
Tato metrická soustava není citlivá na opakování. Například, pokud je sekvence X porovnávána se svým dvojitým opakováním XX, vektor c součtu bude mít odlišné měřítko, ale bude mít stejný směr v prostoru. Protože $c^X = 2c^{XX}$ způsobí, že úhlová vzdálenost mezi nimi bude nulová. [1]

4.10 Univerzální sekvenční mapy

Teorie chaosu je pro své použití iteračních funkcí, základem pro vytvoření metod pro výpočet vzdálenosti nezávislý na rozpoznání L-tic. Na počátku této myšlenky stojí práce nazvaná Chaos Game Representation (CGR), ve které byly použity interční funkce k zastoupení DNA. Využití CGR k vytvoření přechodové matice bez rozpoznání L-tic a následné odvození metrických vzdáleností bylo poprvé formulováno roku 2001. Výhodou univerzálních sekvenčních map (USM) je možnost přesně zastoupit a shrnout jakoukoliv sekvenci ve vícerozměrném prostoru. Porovnání pozic jakýchkoliv dvou prvků vyústí v získání stupně podobnosti příslušné části sekvence. [1]

Například: zastoupení dvou prvků $a = (a_1, \dots, a_n)$ a $b = (b_1, \dots, b_n)$ v souřadnicích USM může být využito k určení odlišnosti mezi těmito dvěma prvky v původních sekvencích.[1]

$$d^{USM}(a, b) = -\log_2 \left(\max_i |a_i - b_i| \right) \tag{4.15}$$



Obrázek 12 – Univerzální sekvenční mapy[5]

Metoda USM může být použita u DNA, proteinů i textů biologického jazyka, bohužel je stále ve stádiu experimentálního vývoje a nebyla otestována na rozsáhlých souborech náročných dat. Zároveň by bylo vhodné použití této metody otestovat pro vícenásobné zarovnání a dotazy pro veřejné databáze.[1]

5 Rozšířené globální zarovnání

Rozšířené globální zarovnání je možné použít k porovnání sekvencí se střídavými úseky podobnosti. Algoritmus rozšířeného zarovnání lze možné použít u stejných sekvencí, které jsou identické v některých úsecích a v jiných jsou rozdílné. Například u DNA organizmů jako je člověk a myš jsou typicky stejné v úsecích exonů, ale rozdílné v úsecích intronů. Tyto sekvence mají mnohem menší globální podobnost v případě, že rozdílné úseky jsou mnohem delší než ty podobné. [2]

Model zarovnání

Algoritmus pro model rozšířeného globálního zarovnání byl vytvořen k práci se sekvencemi, které obsahují střídající se úseky podobnosti. Tedy $A = a_1 a_2 \dots a_m$ a $B = b_1 b_2 \dots b_n$ jsou dvě sekvence o délce m a n. Rozšířené globální zarovnání sekvencí A a B se skládá ze substitucí, mezer a bloků rozdílnosti. Dosazení spojuje rezidua sekvence A s rezidui sekvence B. Mezera obsahuje pouze rezidua jedné sekvence, každému reziduu je přiřazen symbol „-“. Existují dva druhy mezer. Mezera delece, která obsahuje pouze rezidua sekvence A a mezera inzerce obsahující rezidua sekvence B. Blok rozdílnosti se skládá z reziduí jedné nebo obou sekvencí a každé reziduum je spojeno se symbolem „+“. Celkem se vyskytují tři druhy bloků rozdílnosti. První typ se skládá jen z reziduí sekvence A, druhý typ se skládá z reziduí sekvence B a blok rozdílnosti třetího typu se skládá z reziduí obou sekvencí.[2]

Nechť $\sigma(a, b)$ je skórem substituce zahrnujícím rezidua a a b. Skóre mezery o délce k je $-(q + k \times r)$, kde nezáporná čísla q a r jsou penalizace za otevření mezery a vložení mezery. Skóre bloku rozdílnosti je $-d$, kde d je konstantní penalizace za každý blok rozdílnosti v zarovnání. Skóre rozšířeného zarovnání je sumou skóre všech substitucí, všech mezer, všech bloků rozdílnosti. Optimální zarovnání je to s nejvyšším skóre. [2]

Algoritmus pro výpočet optimálního rozšířeného zarovnání sekvencí A a B je zprostředkováván prostřednictvím dynamického programování. Kde $A_i = a_1 a_2 \dots a_i$ a $B_j = b_1 b_2 \dots b_j$ jsou počáteční části o délkách i a j. Matice $S(i, j)$ je maximální skóre rozšířeného zarovnání A_i B_j. Po splnění této podmínky představuje $S(m, n)$ skóre optimálního

rozšířeného zarovnání A a B. Pro výpočet matice S jsou potřeba další tři matice. $H(i,j)$ je maximálním skóre obecného zarovnání A_i a B_j, které končí blokem rozdílnosti. Podobně je definována $D(i,j)$, kde obecné zarovnání končí delecí. $I(i,j)$ je zakončeno inzercí. Následné kroky pro výpočet matic jsou odvozeny z jejich definic. [2]

$$S(0,0) = 0$$

$$S(i,0) = max\{D(i,0), H(i,0)\}\ kde\ i > 0,$$

$$S(0,j) = max\{I(0,j), H(0,j)\}\ kde\ j > 0,$$

$$S(i,j) = max\{S(i-1,j-1) + \sigma(a_i, b_j), D(i,j), I(i,j), H(i,j)\}\ kde\ i > 0\ a\ j > 0.$$

$$D(0,j) = S(0,j) - q\ kde\ j \geq 0,$$

$$D(i,0) = D(i-1,j) - r\ kde\ i > 0,$$

$$D(i,j) = max\{D(i-1,j) - r, S(i-1,j) - q - r\}\ kde\ i > 0\ a\ j > 0.$$

$$I(i,0) = S(i,0) - q\ kde\ i \geq 0,$$

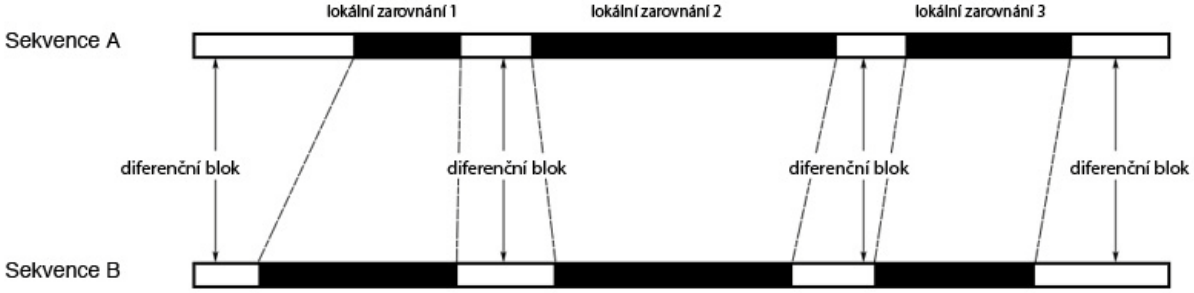
$$I(0,j) = I(0,j-1) - r\ kde\ j > 0,$$

$$I(i,j) = max\{I(i,j-1) - r, S(i,j-1) - q - r\}\ kde\ i > 0\ a\ j > 0.$$

$$H(i,j) = -d\ kde\ i = 0\ a\ j = 0,$$

$$H(i,j) = max\{H(i,j-1), H(i-1,j), S(i,j-1) - d, S(i-1,j) - d\}\ kde\ i > 0\ a\ j > 0. \tag{5.1}$$

Oproti obecnému zarovnání a rozšířenému globálnímu zarovnání je největší rozdíl v matici H. Matice jsou vypočítávány v přesném pořadí[2]



Obrázek 13 - Rozšířené globální zarovnání představuje soubor seřazených lokálních zarovnání oddělených diferenčními bloky[2]

Optimální rozšířené zarovnání je takové, které se skládá ze dvou nebo více diferenčních bloků.[2]

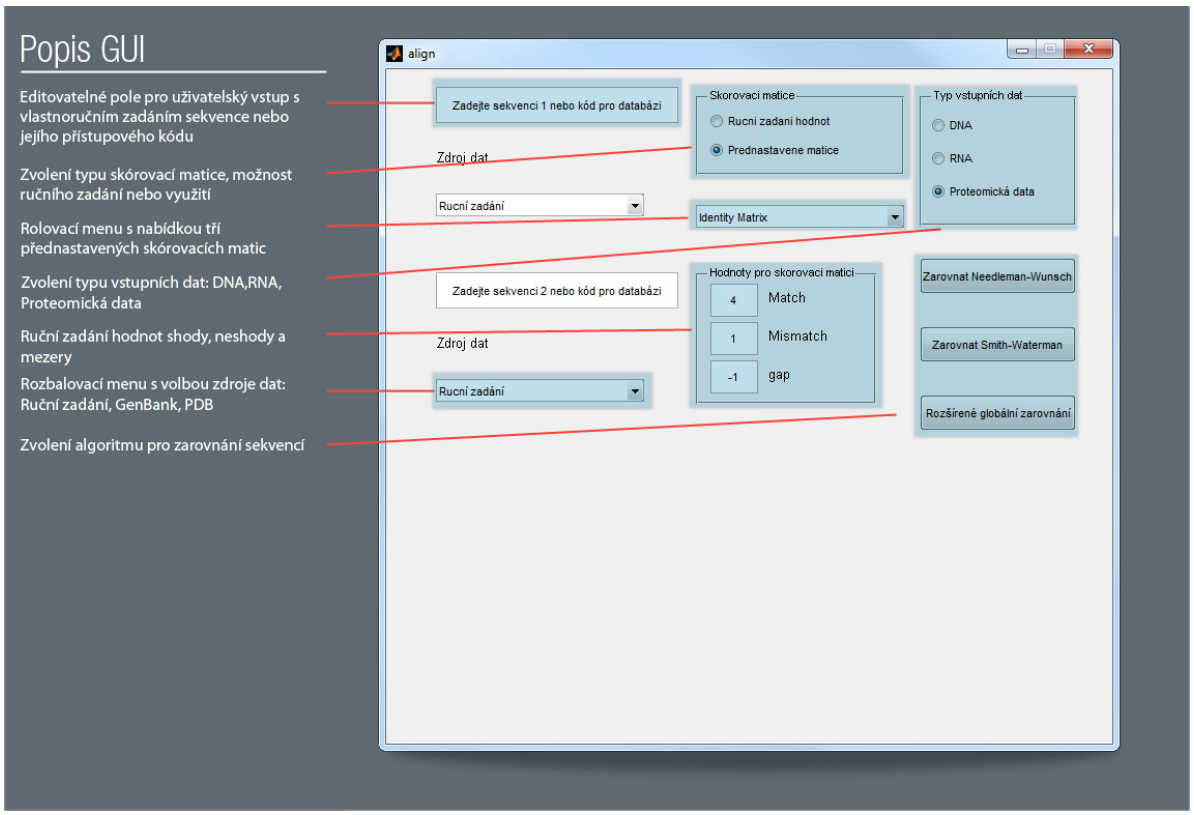
6 Realizace programu

Výstupem této práce je aplikace, která umožňuje uživateli porovnávat biologické sekvence: DNA,RNA a proteomická data. K porovnávání sekvencí obsahuje tři funkce a to: zarovnání algoritmem Needleman-Wunsch, Smith-Waterman a rozšířené globální zarovnání. Rozšířené globální zarovnání je schopné v současné chvíli zpracovávat pouze sekvence DNA. Algoritmy Needleman-Wunsch a Smith-Waterman jsou schopny zarovnávat všechny tři typy vstupních dat. Uživatel má dvě možnosti, jak data požadovaná k zarovnání vložit:

1. Ruční vložení sekvence
2. Vložení kódu sekvence pro jednu ze dvou databází, pro které bylo implementováno rozhranní.

V případě databází se jedná o veřejně přístupné databáze GenBank a Protein Data Bank (PDB).

6.1 GUI – grafické uživatelské rozhraní



Obrázek 14 - GUI aplikace

Aplikace pro zarovnávání sekvencí umožňuje uživateli její použití prostřednictvím grafického uživatelského rozhraní, aniž by musel přicházet do kontaktu se zdrojovým kódem programu. Grafické rozhraní zahrnuje dvě textová pole, do nichž je možné vložit sekvenci nebo přístupový kód požadované sekvence. Dále tři rozbalovací menu (pop-up menu), z nichž dvě přísluší k jednotlivým textovým polím a obsahují volby, zda vložené textové řetězce jsou sekvencemi nebo přístupovými kódy pro dvě implementované databáze. Třetí rozbalovací menu náleží k volbě skórovací matice pro algoritmus Needleman-Wunsch a rozšířené globální zarovnání. Toto menu ovlivňuje chod programu jen v případě, že v nabídce skórovacích matic byly vybrány přednastavené matice. V opačném případě se k vytvoření skórovací matice použijí hodnoty z polí pro ruční zadání hodnot shody (match) a neshody (mismatch). Poslední skupinou tlačítek jsou volby typu sekvencí, zda se jedná o DNA, RNA nebo proteomická data. K samotnému spuštění běhu programu slouží tlačítka tří možných metod zarovnání: Needleman-Wunsch, Smith-Waterman a rozšířené globální zarovnání.

U použité verze Matlabu R2010 byly zaznamenány problémy s diakritikou u GUI, které se nepodařilo zatím uspokojivě vyřešit

6.2 Metody zarovnání použité v aplikaci

Globální zarovnání je realizováno algoritmem dynamického programování Needleman-Wunsch. Program začíná vytvořením matice ohodnocení, jejíž rozměry jsou odvozeny od délky jednotlivých sekvencí, ke kterým je přičteno číslo jedna, protože Matlab není schopen indexovat pozici 0,0 a je tedy nutné začínat od pozice 1,1. První řádek a sloupec nereprezentuje hodnoty pro zarovnání a je naplněn hodnotami, které vycházejí z penalizace mezery. Pozice 1,1 má hodnotu 0, následně se v prvním řádku přičítá penalizace mezery, vždy k předchozí hodnotě. Stejný postup se použije pro první sloupec. Následně algoritmus prochází zadané sekvence po jednotlivých párech, v případě DNA, nukleotidů a na základě jejich shody nebo neshody plní matici ohodnocení příslušnými hodnotami. Mohou nastat tři případy:

1. Nejvyšší bude hodnota zleva po přičtení penalizace za mezeru. Tato možnost reprezentuje vložení mezery.

2. Nejvyšší je hodnota shora opět s přičtením penalizace za mezeru. Také tato možnost reprezentuje vložení mezery.
3. Nebo je nejvyšší hodnota na diagonále po přičtení hodnoty shody nebo penalizace za neshodu. Tato možnost reprezentuje zarovnání nukleotidů.

Do matice ohodnocení se uloží hodnota, a sice ta z výše popsaných tří možností, která je nejvyšší. Když je matice ohodnocení naplněna, hodnota v pravém dolním rohu představuje optimální zarovnání. Následně program hledá cestu zpět, při které postupuje po jednotlivých hodnotách až na počáteční pozici 1,1. Pro tyto kroky slouží zaznamenání údaje o tom, z jaké pozice byla ta aktuální vypočítána. Vertikální pohyb znamená mezeru na horizontální sekvenci, zatímco pohyb vlevo znamená mezeru na sekvenci vertikální.

Aplikace umožňuje zvolit si vlastní skórovací matici zadáním hodnot pro shodu, neshodu. Dále je možné zadat hodnotu penalizace mezery, standardně je nastavena na -1. Jako další možnost jsou připraveny tři přednastavené skórovací matice: matice identity, matice blast a přechodová matice.

	A	T	C	G
A	1	0	0	0
T	0	1	0	0
C	0	0	1	0
G	0	0	0	1

matice identity

	A	T	C	G
A	5	-4	-4	-4
T	-4	5	-4	-4
C	-4	-4	5	-4
G	-4	-4	-4	5

BLAST matice

	A	T	C	G
A	1	-5	-5	-1
T	-5	1	-1	-5
C	-5	-1	1	-5
G	-1	-5	-5	1

přechodová matice

Obrázek 15 – Skórovací matice[11]

Lokální zarovnání je prováděno prostřednictvím algoritmu Smith-Waterman. Principiálně se lokální zarovnání algoritmem Smith-Waterman podobá zarovnání algoritmem Needleman-Wunsch s několika změnami. Konkrétně se jedná o nahrazení všech záporných hodnot skóre v matici ohodnocení nulou. Následuje vyhledání nejvyšší hodnoty částečného zarovnání v celé matici. Od nalezení nejvyšší hodnoty je započato zarovnávání a to až do chvíle, kdy je dosaženo pozice o hodnotě nula. Výsledné lokální zarovnání představuje nejlepší zarovnání obou zarovnávaných sekvencí.

Rozšířené globální zarovnání je třetí součástí aplikace. Algoritmus je vhodný k porovnávání sekvencí se střídajícími se úseky podobností. Algoritmus rozšířeného globálního zarovnání implementovaný v programu vychází z práce *A generalized global alignment algorithm*

profesora Iowské státní univerzity Xiaoqiu Huanga. Pro připravení podkladů k sestavení rozšířeného globálního zarovnání je využito globálního zarovnání algoritmem Needleman-Wunsch resp. matice ohodnocení vytvořené globálním zarovnáním. Hlavní rozdíl mezi globálním zarovnáním a rozšířeným globálním zarovnáním spočívá ve čtyřech maticích použitých pro výpočet matice ohodnocení. Matice S, matice D, matice I a matice H.

Postup pro sestavování zarovnání byl konzultován přímo s autorem práce o rozšířeném globálním zarovnání prof. Huangem, který také poskytl níže popsaný postup pro výpočet matic, které jsou nezbytné k sestavení rozšířeného zarovnání.

Matice H představuje hlavní rozdíl mezi rozšířeným globálním zarovnáním a globálním zarovnáním. Je nutné tyto matice vypočítat v pořadí:

$$S(0,0) = 0$$

$$I(0,0) = S(0,0) - q = -q$$

$$I(0,1) = I(0,0) - r = -q - r$$

$$H(0,1) = -d$$

$$S(0,1) = max\{I(0,1), H(0,1)\} = max\{-q - r, -d\}$$

$$D(0,1) = S(0,1) - q$$

$$i = 1$$

$$D(1,0) = D(0,0) - r$$

$$H(1,0) = -d$$

$$S(1,0) = max\{D(1,0), H(1,0)\}$$

$$I(1,0) = S(1,0) - q$$

$$i = 1 \; a \; j = 1$$

$$D(1,1) = max\{D(0,1) - r, S(0,1) - q - r\}$$

$$I(1,1) = max\{I(1,0) - r, S(1,0) - q - r\}$$

$$H(1,1) = max\{H(0,1), H(1,0), S(0,1) - d, S(1,0) - d\}$$

$$S(1,1) = max\{S(0,0) + \sigma(a_1, b_1), D(1,1), I(1,1), H(1,1)\} \tag{6.1}$$

V případě programovacího prostředí MATLAB je nutné všechny indexy uváděné výše zvýšit o jedna, z již zmíněného důvodu indexování od 1 v MATLABU.

```
for i=2:m
    for j=2:n

        I(1,j)=I(1,j-1)-r;
        H(1,j)=-d;
        S(1,j)=max(I(1,j),H(1,j));
        D(1,j)=S(1,j)-q;

        D(i,1)=D(i-1,1)-r;
        H(i,1)=-d;
        S(i,1)=max(D(i,1),H(i,1));
        I(i,1)=S(i,1)-q;

    end
end

%vypocet hodnot matic S,I,H,D

for i=2:m
    for j=2:n
        D(i,j)=max(D(i-1,j)-r,S(i-1,j)-q-r);
        I(i,j)=max(I(i,j-1)-r,S(i,j-1)-q-r);
        H(i,j)=max(max(H(i-1,j),H(i,j-1)),max(S(i-1,j)-d,S(i,j-1)-
d));
        S(i,j)=max(max(S(i-1,j-1)+maticeOhodnoceni(i-1,j-
1),D(i,j)),max(I(i,j),H(i,j)));
    end
end
```

Obrázek 16 – Příklad zdrojového kódu v MATLABU – výpočet matic S, D, I, H

Matice S obsahuje optimální rozšířené zarovnání. Zarovnání vytvořené touto cestou obsahuje místa s podobnostmi mezi dvěma zarovnávanými sekvencemi a také diferenční bloky. Diferenční bloky jsou úseky sekvencí obsahující mezery, inzerce a delece. Výsledné rozšířené zarovnání je vytvořeno lokálním zarovnáním úseků podobnosti a mezi nimi vloženými diferenčními bloky. Rozšířené globální zarovnání je seřazený soubor lokálního zarovnání a diferenčních bloků.

Tato část spočívá v rozpoznání diferenčních bloků v zarovnání a míst podobnosti. Příslušný úsek podobnosti je zarovnání pomocí algoritmu lokálního zarovnání Smith-Waterman a následně jej za něj přiřazen příslušný diferenční blok, který má následovat.

Skóre rozšířeného zarovnání je součtem skóre všech shod, mezer a diferenčních bloků v zarovnání.

Možnosti porovnávání sekvencí jsou omezeny velikostí paměti, která je Matlabu přidělena. Pro zarovnání byly testovány sekvence o délce až 4453 párů bazí. Při pokusu o zarovnání sekvencí o délkách 5000 párů bazí a více bylo dosaženo limitu pro velikost proměných a jeho změna způsobila pád programu, proto se nedoporučuje.

6.3 Kompatibilita

Přiložený program je v podobě proprietárních souborů Matlab a je spustitelný na počítačích s nainstalovanou aplikací Matlab. Program byl vytvořen vprostředí Matlab R2010a, zpětná kompatibilita byla testována na verzích R2007 a R2009b, ale pro správnou funkci je doporučeno použít nejnovější verzi Matlabu.

Pro zjednodušení uživatelské dostupnosti a použitelnosti byl program prostřednictvím Matlab Compiler převeden do podoby samostatně spustitelné aplikace, kterou je možné provozovat na počítačích se systémem Windows XP a novější. Podmínkou pro spuštění takto zkompilovaného programu je nainstalováné Runtime prostředí Matlab, díky čemuž je možné spouštět programy vytvořené v Matlabu bez nutnosti vlastnit licenci.

6.4 Použitý hardware

Aplikace byla otestována pro zarovnání vybraných sekvencí z GenBank a Protein Data Bank na dvou stolních počítačích a přenosném počítači.

Konfigurace testovacích počítačů			
Typ počítače	stolní počítač	stolní počítač 2	přenosný počítač
Typ procesoru	AMD Athlon II X2	AMD Phenom II X4	Intel Core 2 Duo T7200
Počet jader	2	4	2
Takt procesoru	2.9 GHz	3.2 GHz	2.0 GHz
RAM	2 GB	4 GB	4GB
Operační systém	Windows XP 32-bit	Windows 7 32-bit	Windows 7 64-bit
Verze Matlabu	R2007	R2009b	R2010a

.

Obrázek 17 – Otestované konfigurace PC

Jelikož je výkonnost programu závislá na použitém hardwaru, mohou se výsledné časy zarovnání na jiných PC lišit. Druhým faktorem ovlivňujícím rychlost zarovnání je v případě získávání dat z databazí rychlost internetového připojení daného počítače a rychlost odezvy rozhraní dotazované databáze. Zde uvedené příklady představují výsledky získané při použití PC o výše popsané konfiguraci.

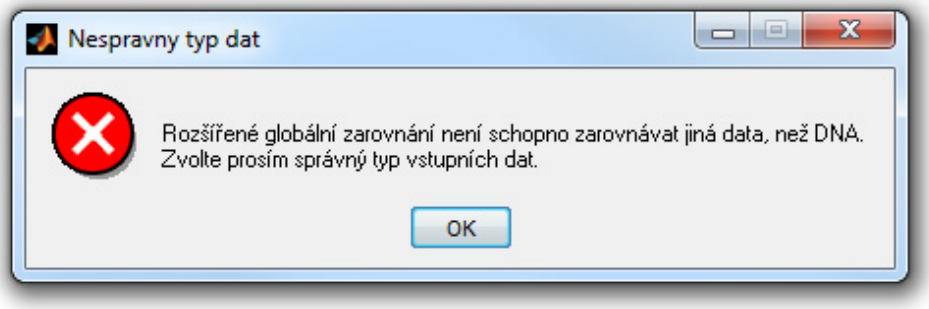
6.5 Práce s programem

Pro správnou činnost programu je nutné vložit do obou textových polí přístupové kódy sekvencí pro načtení sekvencí z databází nebo ručně zadat celou sekvenci. Dále je nutné z rozbalovacích menu vybrat zdroj dat, ačkoliv je standardně nastaven na ruční zadání, je nutné tuto volbu označit, protože v některých případech nebyl Matlab schopen rozpoznat defaultní volbu a vykázal chybu. Toto platí i pro všechny ostatní volby v GUI. Dále je nutné zvolit, jaká skórovací matice se použije, zda budou zadány ručně hodnoty nebo se použije některá z přednastavených. Poté je třeba vybrat, jaký typ dat je určen k zarovnání, zda se jedná o sekvence DNA, RNA nebo proteomická data. Posledním krokem je zvolení algoritmu, kterým budou sekvence zarovnány.

Na několika počítačích se vyskytla chyba, kdy program hlásil problém s cyklem switch, který se provádí u všech voleb. V současné chvíli nebyla odhalena příčina této chyby, ale zřejmě nesouvisí se samotným zdrojovým kódem programu. V případě této chyby bylo nutné Matlab restartovat a následovně již program pracoval v pořádku.

6.6 Nesprávné použití programu

Program je ošetřen chybovými hlášeními pro případě, že uživatel zvolí kombinaci nastavení, kterou není možné provést. Tato informační okna uživatele upozorní na nedovolenou operaci a zobrazí příčinu, pro kterou není možné příkaz vykonat.



Obrázek 18 - Příklad chybového hlášení

7 Výsledky

7.1 Genomická data

Jako první vzorek byly porovnávány sekvence z GenBank s přístupovým kódem HM161754.1 o délce 1870 párů bází z organismu patřícím mezi červené řasy - *Cyanidioschyzon* a sekvence HM161753.1 o délce 1791 bp organizmu *Candida* patřící mezi kvasinky. Sekvence byly nejprve globálně zarovnány algoritmem Needleman-Wunsch, výsledné zarovnání bylo dosaženo při použití matice identity jako skórovací matice za 8s. S výsledkem 29 % podobnosti.

```
Identities = 551/1870 (29%)
0001  CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTTAAAGATTAAGCCATGCATGTCCAAGTATA
      ||||||||||||||||||||||||||||||||| ||||||||||||||||||||| |||||||
0001  CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCTAAGTATA

0065  AGCCCCCATGGGGTGAAACTGCGGATGGCTCATTAAATCGGTTGGAATCTACATGAGAGTTATT
      |||  | ||   |||||||||||| ||||||||||||||| |||    | ||  ||| |||   |
0065  AGCAACTATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTATTTGATAGTACCT

0129  CTTTAAAAGGATAACCCCGGAAAATCTGCGGCTAATACTTGCAAAAAAGGGGGGCAACTCCCAG
       |   |   |   | ||  |  |||    |    |          |||     |    |
0129  TTACTACTTGGATAACCGTGGTAATTCTAGAGCTAATACATGCTTAAAACCCCGACTGTTTGGA

0193  CATTTATCAGCTCGAGAGGATGGCGTAGCAATACGTCTCGGTGAATCATGATAAACAAAGCAGA
              |  |       |     |       | ||   |    |           ||  |
0193  AGGGGTGTATTTATTAGATAAAAAATCAATGCTCTTCGGAGCTCTTTGATGATTCATAATAACT
```

Obrázek 19 - Zarovnání sekvencí HM161754.1 a HM161753.1 algoritmem Needleman – Wunsch

Stejné sekvence byly zarovnány algoritmem lokálního zarovnání Smith-Waterman. V tomto případě byly sekvence zarovnány za 6s. Výsledné zarovnání je s podobností 72 %.

```
Identities = 1394/1946 (72%)
0001  CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTTAAAGATTAAGCCATGCATGTCCAAGTATA
      ||||||||||||||||||||||||||||||||| |||||||||||||||||||||| |||||||
0001  CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCTAAGTATA

0065  AGCCCCCATGGGGTGAAACTGCGGATGGCTCATTAAATCGGTTGGAATCTACATGAGAGTTATT
      |||  | ||   |||||||||||| ||||||||||||||| |||    | ||  ||| ||| |
0065  AGCAACTATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTATTTGATAGT-AC-

0129  CTTTAAAA---GGATAACCCCGGAAAATCT-GCGGCTAATACTTGCAAAAAAGGGGGGCAACTC
      |||||  |   ||||||||  || || ||| | | ||||||| |||  ||||    | |   |
0127  CTTTACTACTTGGATAACCGTGGTAATTCTAGAG-CTAATACATGCTTAAAACCCCGACTGTTT

0189  CCA-GC-----ATTTATCAGCTCGAGAGG-A-TGG-CGTAGCAATACGTCTCGG-TGAATCATG
        | |      |||||| || |  | |   | ||  | | | |   | |||  | ||| ||||
0190  GGAAGGGGTGTATTTATTAGATAAAAAATCAATGCTCTTCGGAG--C-TCTTTGATGATTCATA
```

Obrázek 20- Zarovnání sekvencí HM161754.1 a HM161753.1 algoritmem Smith – Waterman

Nakonec byly sekvence zarovnány algoritmem rozšířeného globálního zarovnání. V tomto případě bylo zarovnání dosaženo za 15s. S výslednou podobností mezi sekvencenmi 73 % a počtem diferenčních bloků: 148.

```
Identities = 1266/1737 (73%)
0001  GTGAAACTGCGGATGGCTCATTAAATCGGTTGGAATCTACATGAGAGTTATCTTTAAAA---GG
      |||||||||||| ||||||||||||||| |||    | ||  ||| ||| | |||||  |   ||
0001  GTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTATTTGATAGT-A-CTTTACTACTTGG

0062  ATAACCCCGGAAAATCT-GCGGCTAATACTTGCAAAAAAGGGGGGCAACT-G-----ATTTATC
      ||||||  || || ||| | | ||||||| |||  ||||    | |   | |     ||||||
0063  ATAACCGTGGTAATTCTAGAG-CTAATACATGCTTAAAACCCCGACTGTTAGGGTGTATTTATT

0119  AGCTCGAGA-A-TG-CGTAGTACGTCTCGG-TGAATCATGATAAACAAAGCAGA-TCGCACGAT
      || |  | | | || || ||  | |||  | ||| |||| |||| |    | || ||||| |
0126  AGATAAAAACAATGTCGGAG--C-TCTTTGATGATTCATAATAA-CTTTTC-GAATCGCATGGC

0178  CTTGTTC-GGCGACT-TTCATTCGACTTTCCGCCCTATCAACTTTGGAAGGTAGTG-TAGTGGA
      ||||| | ||||| | ||||||| | |||| |||||||||||||| || ||||| | ||||||
0185  CTTGT-CTGGCGA-TGTTCATTCAAATTTCTGCCCTATCAACTTTCGATGGTAG-GATAGTGGC
```

Obrázek 21 - Zarovnání sekvencí HM161754.1 a HM161753.1 algoritmem Rozšířeného globálního zarovnání

Dalšími sekvencemi pro testování byly S74539 o délce 198 bp - *lidský hepatický protein* a S69912 o délce 129 bp - *tyrozinkynáza, člověk*. Prvně bylo opět provedeno zarovnání algoritmem Needleman-Wunsch, zarovnání bylo sestaveno za 5s s podobností 16 %.

```
Identities = 31/198 (16%)
001  TCATGCCCAGTTAACTTATTTACAATATTTAAGTTCTCTGCTTCTGCATTTGGTTGGTTTCCTG
      | | |   |    |      | |         |          |  | |   || ||       |
001  CCCTACAAGGCAGGCGCGGGCAGAGGCAGGTGGGCGGTGTGGTGGGGAGGGGGATGAGGAGGAG

065  AAGCGCNCCCTGTGAATAACAGGTGGCTTTTCATGGATGTCTCTAGTCAGAGAAAAATGATAAA
      |       |  | |        |         | ||       |   |  |      | | |
065  GACACTGGTCACTCACAGGTGTCTCTGCCCAGCTTGAGCAGAAGATGAAAGGGCCGGAGGTCAT

129  GGCTTAAATTGAGGATTAACAGAAGCAGATTAACCTCAGAAATCCTGTCTGGCTGGCAGATTTC
     |
129  G-----------------------------------------------------------------

193  AAGTAA

130  ------
```

Obrázek 22 - Zarovnání sekvencí S41539 a S69912 algoritmem Needleman - Wunsch

Druhým v pořadí bylo lokální zarovnání Smith-Waterman, které bylo sestaveno za 4s s podobností 47 %.

```
Identities = 82/175 (47%)
001  CAATATTTAAGTTCTCTGCTTCTG-CATTTGGTTGGTTTCCTGAAGCGCNCCCTGTGAATAAC-
     |  ||   |||  |   ||  | | ||   ||  |||     |  |||     ||||  |
001  CCCTAC--AAGG-CAG-GCG-CGGGCAGA-GGCAGGTG----G--GCGG----TGTGG-TGGGG

063  AGGTGGCTTTTCATGGATGTCTCTAGTCAGAGAAAAATGATAAAGGCTTAAATTGAGGATTAAC
     ||| || |    | ||| | | || ||||   | |  || |    ||  |  |||||      |
048  AGGGGGATGAGGA-GGAGGACACTGGTCACTCACAGGTG-TCTCTGCCCAGCTTGAG------C

127  AGAAGCAGATTAACCTCAGAAATCCTGTCTGGCTGGCAGATTTCAAG
     ||||| |  |        ||||    |   ||| || || |  || |
104  AGAAG-A--T--------GAAA----G---GGCCGG-AGGT--CATG
```

Obrázek 23 - Zarovnání sekvencí S41539 a S69912 algoritmem Smith – Waterman

Třetím zarovnáním bylo rozšířené globální zarovnání. Zarovnání proběhlo za 3s podobností 38 % a počtem diferenčních bloků: 19.

Poslední sadou testovaných sekvencí byly sekvence myši domácí – AF104261.1 - *Mus musculus Pax transcription* o délce 3882 bp a sekvence norské krysy – NM_001107844.1 – *Rattus norvegicus PAX interacting* o délce 3545 bp.

Zarovnání Needleman-Wunsch bylo sestaveno za 11s s podobností 22 % a rozšířené globální zarovnání bylo sestaveno za 60s s podobností 84 %.

Na tomto případu je dobře znatelné, že rozšířené globální zarovnání dosahuje výrazně lepších výsledků zarovnání u homologních sekvencí s nižší podobností a rozdílnými délkami, než globální zarovnání algoritmem Needleman-Wunsch. Rychlost zarovnání je závislá na hardware používaného počítače. V případě krátkých sekvencí jsou rozdíly rychlostí zanedbatelné, ale v případě dlouhých sekvencí je již znatelné, že rozšířené globální zarovnání je ve většině případů pomalejší než Needleman-Wunsch nebo Smith-Waterman. Zdlouhavost rozšířeného globálního zarovnání je možné přičítat rozsáhlosti výpočtů, které jsou s ním spojeny. Vliv může mít i to, že samotné zarovnání sestává z několika funkcí, které jsou postupně volány v průběhu činnosti programu.

7.2 Proteomická data

Pro příklad zarovnání sekvencí proteomických dat byly použity sekvence z databáze Protein data Bank. První z nich je sekvence s přístupovým kódem 5CYT - *Refinement of Myoglobin and Cytochrome C* o délce 104. Druhou s ní porovnávanou sekvencí byla 2KXJ - *UBX domain-containing protein 4* o délce 90. Zarovnání algoritmem Needleman-Wunsch bylo dosaženo za 10s s podobností 24 %.

```
Identities = 53/296 (18%)
001  TEEQIAEFKEAFSL-FDKDGDGTIT-TKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFP
            |    |  |   | ||   |  |    |     |         | |
001  ----VG-FN--YSIDFT--G-GT-AYT--L----RA--E-P---------N-VE------V---

063  EFLTMMARKMKDTDSEEEIREAFRVFD-KDGNGYISAAELRHVMTNLGEKLTDEEVDEMIREAD
     |  |   |           |  |     | |                | |   | |   |
026  E--TL--R-----------R--F-L-EEK-G--F-P-----------G-K---EAV---I--T-

126  IDGDGQVNYEEFVQMMTAKADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTE
         ||      |   |     |    |   | | |           | |          |
046  -----QV------Q---AP----TA---A-YRE-F-L--------V--K-L-P--------P--

190  AELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVM
       |       |          |      |       |    |  | |         | ||
064  --L----S--D----------E------RRL-----E--L-E--RLF---------ASEL----

254  TNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMMTAK-
         | |   |      |          |      |
081  -----KAT---V--L---A--------S-E------TV-G
```

Obrázek 24 - Zarovnání sekvencí 5CYT a 2KXJ algoritmem Needleman - Wunsch

Lokální zarovnání Smith-Waterman bylo vytvořeno za 9s s podobností 23 %.

```
Identities = 29/104 (28%)
001  REAFRVF-DKDGNG--YISAAELRH-VMTNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMM
        |    |  | |  |   ||    | | |   |  ||      || |    ||
001  MVGFNYSIDFTG-GTAYTLRAEPNVEVET-LRRFL--EEKGFPGKEAVIT---QVQAPTAAYRE

061  TA-KADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVM
        |   |  |   |  |   ||      |    |  ||
058  FLVKLPPLSDERRLEL-ER--LFASELKATVLASE--TVG
```

Obrázek 25 - Zarovnání sekvencí 5CYT a 2KXJ algoritmem Smith - Waterman

Druhým párem sekvencí byly 2L7L - *Fast methionine-based solution structure determination of calcium-calmodulin complexes* o délce 148 a 2RRN - *Structure and function of a membrane component SecDF that enhances protein export* o délce 92. Globální zarovnání bylo sestaveno za 9s s podobností 18 %. Lokální zarovnání bylo vytvořeno za 9s s podobností 28 %.

```
Identities = 53/296 (18%)
001  TEEQIAEFKEAFSL-FDKDGDGTIT-TKELGTVMRSLGQNPTEAELQDMINEVDADGNGTIDFP
            |    |  |   | ||   |  |    |     |         | |
001  ----VG-FN--YSIDFT--G-GT-AYT--L----RA--E-P---------N-VE------V---

063  EFLTMMARKMKDTDSEEEIREAFRVFD-KDGNGYISAAELRHVMTNLGEKLTDEEVDEMIREAD
     |  |   |           |  |     | |                | |   | |   |
026  E--TL--R-----------R--F-L-EEK-G--F-P-----------G-K---EAV---I--T-

126  IDGDGQVNYEEFVQMMTAKADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVMRSLGQNPTE
         ||      |   |     |    |   | | |           | |          |
046  -----QV------Q---AP----TA---A-YRE-F-L--------V--K-L-P--------P--

190  AELQDMINEVDADGNGTIDFPEFLTMMARKMKDTDSEEEIREAFRVFDKDGNGYISAAELRHVM
       |       |          |      |       |    |  | |         | ||
064  --L----S--D----------E------RRL-----E--L-E--RLF---------ASEL----

254  TNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMMTAK-
         | |   |      |          |      |
081  -----KAT---V--L---A--------S-E------TV-G
```

Obrázek 26 - Zarovnání sekvencí 2L7L a 2RRN algoritmem Needleman - Wunsch

```
Identities = 29/104 (28%)
001  REAFRVF-DKDGNG--YISAAELRH-VMTNLGEKLTDEEVDEMIREADIDGDGQVNYEEFVQMM
        |    |  | |  |   ||    | | |   |  ||      || |    ||
001  MVGFNYSIDFTG-GTAYTLRAEPNVEVET-LRRFL--EEKGFPGKEAVIT---QVQAPTAAYRE

061  TA-KADQLTEEQIAEFKEAFSLFDKDGDGTITTKELGTVM
        |   |  |   |  |   ||      |    |  ||
058  FLVKLPPLSDERRLEL-ER--LFASELKATVLASE--TVG
```

Obrázek 27 - Zarovnání sekvencí 2L7L a 2RRN algoritmem Smith – Waterman

Příklad zarovnání proteomických dat algortimy pro globální a lokální zarovnání ilustruje použitelnost jednotlivých algoritmů, kdy globální zarovnání je vhodné používat k zarovnávání sekvencí, které jsou si více podobné a jejich délky se příliš neliší. Naopak lokální zarovnání je vhodné k zarovnávání různě dlouhých a méně podobných sekvencí, protože se zaměřuje jen na jednotlivé úseky lokální podobnosti mezi zarovnávanými sekvencemi.

8 Závěr

Tato práce shrnuje existující metody určování podobností sekvencí DNA. Popisuje dostupné algoritmy a metody jejich aplikace. Těžistě práce potom představují algoritmy globálního a lokálního zarovnání Needleman-Wunsch a Smith-Waterman. Samostatnou část představuje rozšířené globální zarovnání, které principiálně vychází z algoritmu Needleman-Wunsch. Tyto tři algoritmy představují integrální část vytvořené aplikace v prostředí Matlab. Aplikace dále umožňuje získávat sekvence pro zarovnání z veřejně přístupných databází GenBank a Protein Data Bank.

Jádro programu tvoří rozšířené globální zarovnání, které se skládá ze tří funkcí tvořících celek algoritmu pro rozšířené zarovnání. Rozšířené globální zarovnání je vhodné pro zarovnávání jednotvárných sekvencí s méně podobnými úseky. Algoritmy Needleman-Wunsch a Smith-Waterman sekvence procházejí po menších úsecích a proto jsou úspěšnější při použití u sekvencí s vysoce podobnými úseky, Rozšířené globální zarovnání projde celou sekvenci a je schopno přesněji zarovnávat i sekvence s méně podobné úseky. Ovšem za cenu vyšší časové náročnosti.

Jako pokračování této práce se nabízejí modifikace stávající aplikace, především v oblasti automatizovaného rozpoznávaní typu vstupních dat. Další možnou změnu představuje také automatizováné zvolení zdroje dat a přidání funkcionality rozpoznání ručně zadaných sekvencí nebo přístupových kódů pro databáze dat a jejich další dělení na jednotlivé databáze GenBank a PDB. A vzhledem k členění programu na jednotlivé funkce se nabízí možnost přidat další méně časté metody zarovnání biologických sekvencí.

9 Seznam literatury

[1] VINGA, Susana and ALMEIDA, Jonas. Alignment-free sequence comparision – a review. Oxford Journals [online]. 2002, říjen [cit. 15.dubna 2011]. Dostupné na WWW: http://bioinformatics.oxfordjournals.org/content/19/4/513.full.pdf

[2] HUANG, Xiaoqiu and CHAO, Kun-Mao. A generalized global aligmnet algorithm. Oxford Journals [online]. 2002, září s[cit. 5.května 2011]. Dostupné na WWW: http://bioinformatics.oxfordjournals.org/content/19/2/228.full.pdf

[3] LIKIĆ, Vladimir. The Needleman-Wunsch algorithm for sequence alignment. 2005[cit. 6.prosince 2010].
Dostupné na WWW: http://www.ludwig.edu.au/course/lectures2005/likic.pdf

[4] CVRČKOVÁ, Fatima. Úvod do praktické bioinformatiky. 1. vyd. Praha: Academia, 2006. 148 stran. ISBN 80-200-1360-1.

[5] NCBI. Univerzální sekvenční mapa. [online]. 2002, únor [cit. 6. ledna 2010]. Dostupné na WWW:
http://www.ncbi.nlm.nih.gov/pmc/articles/PMC90187/figure/F2/

[6] AYGUADE, Eduard; NAVARRO, J., Juan a GONZALES, Jimenez, Dani. Smith-Waterman Algorithm. 2007, říjen [cit. 15. listopadu 2010]. Dostupné na WWW: http://docencia.ac.upc.edu/master/AMPP/slides/ampp_sw_presentation.pdf

[7] MOUNT, David W. Bioinformatics: Sequence and genome analysis. 2. vyd. New York: Cold Spring Harbor Laboratory Press, 2004. 692 stran. ISBN 0-87969-712-1

[8] MARKEL, Scott; LEÓN, Darryl. Sequence analysis in nutshell. 1. vyd. Sebastopol: O'Reilly & Associates, 2004. 286 stran. ISBN 0-596-00494-X

[9] BORODOVSKY, Mark; EKISHEVA Svetlana. Problems and Solutions in Biological Sequence Analysis. 1. vyd. New York: Cambridge University Press, 2006. 346 stran. ISBN-13 978-0-521-84754-4

[10] BRUNAK, Søren; BALDI Pierre. Bioinformatics: The Machine Approach. 1. vyd. Cambridge: The MIT Press, 2001. 452 stran. ISBN 0-262-02506-X

[11] KRANE. Dan E.; RAYNER Michael L. Fundamental concepts of bioinformatics. 1. vyd. San Francisco: Pearson Education, 2003. 314 stran. ISBN 0-8053-4633-3

[12] BIOINFORMÀTICA, un curs d'anàlisi de seqüències, [online]. 2007, [cit. 1. května 2011]. Dostupné na WWW:

http://bioinformatica.upf.edu/2007/projectes07/B.10/clustalwPRR5.bmp

[13] L'UFR des Sciences Médicales, [online]. 2011, [cit. 25. dubna 2011]. Dostupné na WWW: http://www.med.univ-angers.fr/discipline/bio_cel/Maitrise/Bioinfo/images/pam250.JPG

[14] Davidson College, [online]. 2008 [cit. 14. dubna 2011]. Dostupné na WWW: http://www.bio.davidson.edu/courses/genomics/2008/Simpson/BLOSUM62.png