Chemoinformatika a bioinformatika

Sequence alignment

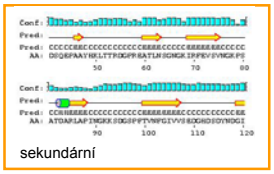
Osnova

1. Struktura biomakromolekul – sekvence
2. Alignment a jeho typy
3. Užívané algoritmy
4. Multiple sequence alignment
5. Programové balíky
6. Benchmark – porovnávání alignmentů

Struktura proteinů (NK)

ADSQTSSNRAGEFSIPPNTDFRAIF
FANAAEQQHIKLFIGDSQEPAAYHK
LTTRDGPREATLNSGNGKIRFEVSV
NGKPSATDARLAPINGKKSDGSPF
TVNFGIVVSEDGHDSDYNDGIVVL
QWPIG

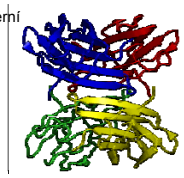
primární (sekvence)



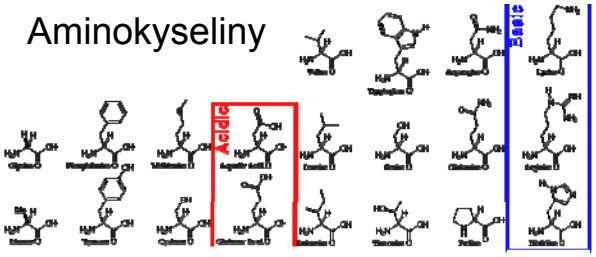
sekundární

terciární

kvartérní

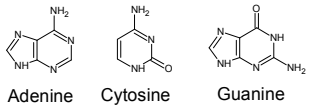


Aminokyseliny

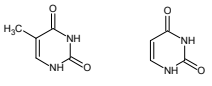


glycin	alanin	valin	leucin	isoleucin	asparagová kys.	asparagin	glutamová kys.	glutamin	arginin	lysin	histidin	fenylalanin	serin	threonin	tryptofan	tyrozin	methionin	cystein	prolin	selenocystein
Gly	Ala	Val	Leu	Ile	Asp	Asn	Glu	Gln	Arg	Lys	His	Phe	Ser	Thr	Tyr	Trp	Met	Cys	Pro	Sec
G	A	V	L	I	D	N	E	Q	R	K	H	F	S	T	Y	W	M	C	P	U

Nukleové báze



Adenine Cytosine Guanine

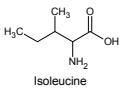


Thymine Uracil

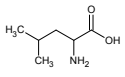
adenin	cytosin	guanin	thymin	uracil
A	C	G	T	U

Třídění aminokyselin

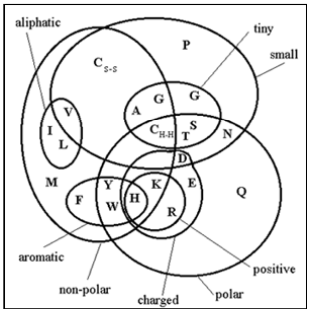
Aminokyseliny s podobnými vlastnostmi mohou plnit v proteinu stejné funkce – bývají vzájemně zastupitelné



Isoleucine



Leucine



Alignment

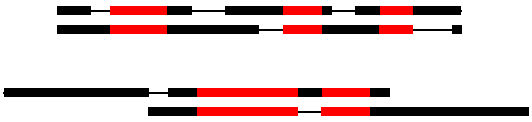
Srovnání (přiložení) dvou či více sekvencí (aminokyselinových, nukleotidových) na základě jejich vzájemné podobnosti.

Pairwise alignment – dvě sekvence

Multiple sequence alignment – více sekvencí

Pair-wise alignment

- Srovnání dvou sekvencí
- Sekvence mohou být seřazeny v celé své délce (global alignment) nebo jen v určitém regionu (local alignment).

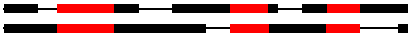


Local alignment

Hledá úseky dvou sekvencí, které si podle zvolených kritérií dobře odpovídají. Nesnaží se zahrnout celé sekvence, pokud si jejich některé části neopdovídají.

Global alignment

Vychází z předpokladu, že obě srovnávané sekvence jsou víceméně shodné v celé své délce. Alignment k sobě přikládá celé sekvence (od počátku do konce) a to včetně částí, které si nepříliš odpovídají.



Algoritmy

- Témeř výhradně se užívají **heuristické algoritmy** – nalezení výsledku v dostatečně krátkém čase
- Vývoj algoritmů je prováděn v návaznosti na srovnávání výsledků s tzv. zlatým standardem – alignment na základě známé 3D struktur

Vstupní data

Sekvence AK (nt) v určitém formátu – dnes desítky formátů, mnohé obsahují krom sekvence i doplňující data

Bližší např.
http://emboss.sourceforge.net/docs/themes/SequenceFormats.html

- **FASTA formát**

>název␣popis dle vlastní volby↵
SEKVENCESEKVENCESEKVENCESEKVENCESEKVENCESEKVENCE

Scoring matrix (skórovací matice)

- Dvě sekvence považujeme za **příbuzné**, vycházejí-li ze společného předka; pak dobu potřebnou k jejich evoluci můžeme odvodit z množství rozdílů mezi nimi

- **Záměna** aa je častější než inserce/delece. Pravděpodobnost změny jedné aminokyseliny na jinou **je** přímo **úměrná podobnosti** obou aminokyselin.
- **Matice** vzniká přiřazením hodnoty (pravděpodobnosti) jednotlivým dvojcím aminokyselin v závislosti na jejich vzájemné „zastupitelnosti" – pravděpodobnosti substituce

Typy matic

- **PAM** (Point Accepted Mutation) – založena na mutacích v rámci globálního alignmentu, tj. ve vysoce konzerovovaných i mutabilních oblastech.
- **BLOSSUM** (Blocks Substitution Matrix) – je odvozena z vysoce konzervovaných oblastí neobsahujících mezery → lepší pro lokální alignment. Je využívána v blastp, vhodná pro identifikaci neznámé nukleotidové sekvence.
- **GONNET** – vytvořena 1992, postupným opakováním cyklu: pairwise alignment – nová matice – nový pairwise alignment – nová matice…
- **DNA identity** matrix

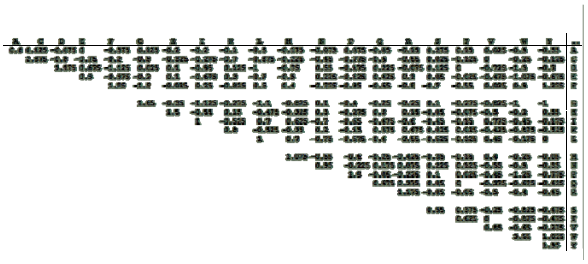
V rámci jednoho typu existuje **více** jednotlivých **matic** založených na stejném principu, které se však liší konkrétními hodnotami a tedy i **oblastí použití** (vysoce příbuzné nebo naopak velmi vzdálené sekvence).

Matice PAM 250

C	12																
G	-3	5															
P	-3	-1	6														
S	0	1	1	1													
A	-2	1	1	1	2												
T	-2	0	0	1	1	3											
D	-5	1	-1	0	0	0	4										
E	-5	0	-1	0	0	0	3	4									
N	-4	0	-1	1	0	0	2	1	2								
Q	-5	-1	0	-1	0	-1	2	2	1	4							
H	-3	-2	0	-1	-1	-1	1	1	2	3	6						
K	-5	-2	-1	0	-1	0	0	0	1	1	0	5					
R	-4	-3	0	0	-2	-1	-1	-1	0	1	2	3	6				
V	-2	-1	-1	-1	0	0	-2	-2	-2	-2	-2	-2	4				
M	-5	-3	-2	-2	-1	-1	-3	-2	0	-1	-2	0	0	2	6		
I	-2	-3	-2	-1	-1	0	-2	-2	-2	-2	-2	-2	4	2	5		
L	-6	-4	-3	-3	-2	-2	-4	-3	-3	-2	-2	-3	-3	2	4	2	6		
F	-4	-5	-5	-3	-4	-3	-6	-5	-4	-5	-2	-5	-4	-1	0	1	2	9	
Y	0	-5	-5	-3	-3	-3	-4	-4	-2	-4	0	-4	-5	-2	-2	-1	-1	7	10	
W	-8	-7	-6	-2	-6	-5	-7	-7	-4	-5	-3	-3	2	-6	-4	-5	-2	0	0	17
	C	G	P	S	A	T	D	E	N	Q	H	K	R	V	M	I	L	F	Y	W

Matrice BLOSSUM vypadá analogicky, liší se hodnoty.

GONNETova matice



DNA matice

A	1			
T	-10000	1		
G	-10000	-10000	1	
C	-10000	-10000	-10000	1
	A	T	G	C

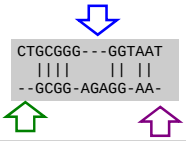
Jako pozitivní je uvažována pouze shoda, jakákoliv substituce je vysoce penalizována; jsou však povoleny mezery.

Mezery (Gaps)

Příčiny vzniku mezer:

- **Bodová mutace** (velmi častá příčina)
- Nepřesný crossover při meióze (inzerce nebo delece řetězce bází)
- DNA slippage během replikace (vzniká repetice – opakující se sekvence v řetězci)
- Inzerce retroviru
- Translokace DNA mezi chromozomy

Mezery nacházíme na **začátku** řetězce, **uprostřed** nebo na jeho **konci**.



Mezery umožňují alignment sekvencí, kdy v jedné z nich došlo k delecí. Zvyšují však také možnost alignmentu náhodných sekvencí. Jejich přítomnost je proto vždy „**penalizována**“, a to více než substituce.

Čím nižší je penalizace mezer, tím lepší (dokonalejší) bude alignment, ovšem z biologického hlediska může jít o nesmysl.

Jednotlivé programy obvykle penalizují **přítomnost mezery** (gap open) a také zvyšují penalizaci s **délkou mezery** (gap ext).

Krátká mezera:

```
ATCTTCAGTGTTTCCCCTGTTTTGCCC.ATTTAGTTCGCTC
||||||||||||||||||||||||||| ||||||||||||||
ATCTTCAGTGTTTCCCCTGTTTTGCCCGATTTAGTTCGCTC
```

Dlouhá mezera:

```
ATCTTCAGTGTTTCCCCTGTTTTGCCC..................ATTTAGTTCGCTC
||||||||||||||||||||||||||| ||||||||||||||
ATCTTCAGTGTTTCCCCTGTTTTGCCCGCCCCCCCCCCCCCCCCCATTTAGTTCGCTC
```

Skóre

Každé dvojici sekvencí je ve výsledku přiřazeno číslo – skóre, které určuje míru jejich podobnosti

Čím vyšší je skóre, tím vyšší je podobnost.

Podle použité matice může být skóre i záporné.

Multiple sequence alignment - MSA

(mnohonásobné srovnání)

Multiple alignment slouží k:

- Nalezení „diagnostického vzoru“ (diagnostic patterns) na jehož základě jsou **charakterizovány proteinové rodiny**
- Odhalení či dokázání **homologie** mezi novou sekvencí a sekvencemi v databázích
- Určení vzájemné příbuznosti sekvenci v rámci skupiny – tvorba **fylogenetických stromů**
- **Predikci** sekundární a terciární **struktury** nových proteinů
- Navržení primerů (oligonukleotidů) pro PCR

Metody MSA

- Dynamické programování (dynamic programming) – rozšíření pairwise alignmentu
- **Progresivní alignment** (progressive sequence alignment) – nejčastěji používaný k vytvoření alignmentu; využívá fylogenetické informace
- Iterativní alignment (iterative sequence alignment) – odstraňuje problémy progresivního alignmentu pomocí opakování alignmentu pro podskupiny sekvencí

Dynamické programování

- **Simultánní alignment všech sekvencí** - analogické pairwise alignmentu
- Programové balíky: MSA (Lipman et al., 1989) a DCA (Stoye et al., 1997), založené na Carrilově a Lipmanově algoritmu (1988)
- Využívá skórovací matice, ale vytváří n-rozměrný prostor (n = počet sekvencí)
- Extrémně **náročný na výpočetní kapacity**
- I při zjednodušení nepoužitelné pro více než cca 20 sekvencí

Progresivní multiple alignment

- Používá ho většina programů
- Vznik – 1987

  Feng, D.-F. and Doolittle, R.F. (1987) J. Mol. Evol. 25, 351-360.

1) sestavení příbuzenského stromu (guide tree) z nepřiložených sekvencí

Guide tree vs. phylogenetic tree

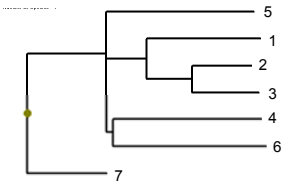
- **Guide tree** je vypočítán na základě matice vzdáleností (distance matrix) vytvořené podle skóre pairwise alignmentů. Výstupem je .dnd soubor.
- **Phylogenetic tree** je vypočten na základě vytvořeného MSA. Vzdálenosti mezi sekvencemi jsou vypočteny a uloženy jako .ph soubor. Následně je možno je využít pro konstrukci fylogenetického stromu (soubory .nj, .ph, .dst) pomocí zvolené metody (nj, phylip, dist).

.dnd soubor

```
(
(
PAIIL:0.16435,
RSIIL:0.13654)
:0.03384,
(
CVIIL:0.16563,
BCLB:0.26800)
:0.02264,
(
(
BCLA:0.17899,
BCLD:0.26633)
:0.18717,
BCLC:0.29707)
:0.03484);
```

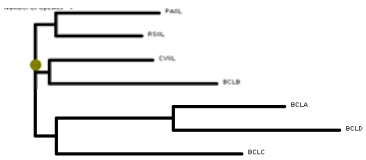
.nj soubor

```
DIST   = percentage divergence (/100)
Length = number of sites used in comparison
1 vs.  2  DIST = 0.6491;  length =   114
1 vs.  3  DIST = 0.6842;  length =   114
1 vs.  4  DIST = 0.9298;  length =   114
1 vs.  5  DIST = 0.9035;  length =   114
1 vs.  6  DIST = 0.9386;  length =   114
1 vs.  7  DIST = 0.9825;  length =   114
2 vs.  3  DIST = 0.3772;  length =   114
2 vs.  4  DIST = 0.9123;  length =   114
2 vs.  5  DIST = 0.8947;  length =   114
2 vs.  6  DIST = 0.9123;  length =   114
2 vs.  7  DIST = 0.9386;  length =   114
3 vs.  4  DIST = 0.9123;  length =   114
3 vs.  5  DIST = 0.9386;  length =   114
3 vs.  6  DIST = 0.9298;  length =   114
3 vs.  7  DIST = 0.9474;  length =   114
4 vs.  5  DIST = 0.9211;  length =   114
4 vs.  6  DIST = 0.9035;  length =   114
4 vs.  7  DIST = 0.9649;  length =   114
5 vs.  6  DIST = 0.9561;  length =   114
5 vs.  7  DIST = 0.9211;  length =   114
6 vs.  7  DIST = 0.9649;  length =   114
          Neighbor-joining Method
Saitou, N. and Nei, M. (1987) The Neighbor-joining Method:
A New Method for Reconstructing Phylogenetic Trees.
Mol. Biol. Evol., 4(4), 406-425
This is an UNROOTED tree
Numbers in parentheses are branch lengths
Cycle  1   = SEQ:  2 (  0.17807) joins  SEQ:  3 (  0.19912)
Cycle  2   = SEQ:  1 (  0.34101) joins Node:  2 (  0.13706)
Cycle  3   = SEQ:  5 (  0.44298) joins  SEQ:  7 (  0.47807)
Cycle  4   = SEQ:  4 (  0.44518) joins  SEQ:  6 (  0.45833)
Cycle  5 (Last cycle, trichotomy):
Node:  1 (  0.12171) joins
Node:  4 (  0.01864) joins
Node:  5 (  0.02083)
```



.ph soubor

```
(
(
PAIIL:0.34101,
(
RSIIL:0.17807,
CVIIL:0.19912)
:0.13706)
:0.12171,
(
BCLA:0.44518,
BCLC:0.45833)
:0.01864,
(
BCLB:0.44298,
BCLD:0.47807)
:0.02083);
```



.dst soubor

7

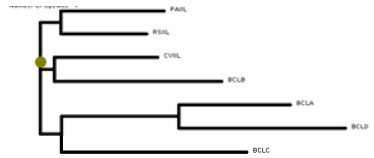
PAIIL	0.000	0.649	0.684	0.930	0.904	0.939	0.982
RSIIL	0.649	0.000	0.377	0.912	0.895	0.912	0.939
CVIIL	0.684	0.377	0.000	0.912	0.939	0.930	0.947
BCLA	0.930	0.912	0.912	0.000	0.921	0.904	0.965
BCLB	0.904	0.895	0.939	0.921	0.000	0.956	0.921
BCLC	0.939	0.912	0.930	0.904	0.956	0.000	0.965
BCLD	0.982	0.939	0.947	0.965	0.921	0.965	0.000

Phylogram a cladogram

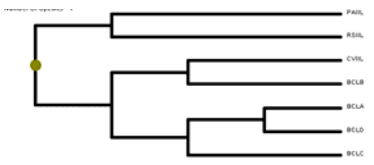
- **Phylogram** (phylogeny tree) – je rozvětvený diagram (strom), který naznačuje fylogenezi (postupný vývoj). Délka jednotlivých větví je úměrná **velikosti změny** v průběhu evoluce.
- **Cladogram** – rovněž strom, v němž však všechny větve mají **stejnou délku**. Ukazuje tak sice „společné předky“ pro jednotlivé sekvence, ale ne množství změn, jež od té doby prodělaly (evoluční dobu).

Phylogram a cladogram

Phylogram



Cladogram



Progresivní multiple alignment

- Používá ho většina programů
- Vznik – 1987
  Feng, D.-F. and Doolittle, R.F. (1987) J. Mol. Evol. 25, 351-360.

1) sestavení příbuzenského stromu (guide tree) z nepřiložených sekvencí
2) tvorba párových alignmentů postupně podle příbuznosti (topologie guide tree)

- Dnes obsahuje často iterativní smyčku

Iterativní přístup

(Gotoh, 1996; Notredame & Higgins, 1996)

Vzniklý strom i alignment jsou následně **optimalizovány** do konvergence. Jinak jsou chyby vzniklé při prvním alignmentu (tvorba stromu) zachovány i ve výsledku.

Nezaručuje nalezení nejlepšího výsledku, ale – na rozdíl od deterministických alternativ – je dostatečně **robustní** a dobře použitelný i pro velký počet sekvencí.

Kombinace local a global alignment

- S výhodou lze kombinovat lokální a globální alignment.
- Lokální alignment může být reprezentován sadou kotvících bodů v místě dobré shody
- Následný globální alignment pak tyto odpovídající úseky sekvencí zahrnuje (využito např. v ClustalW2)

Výstup

Výstupem je sada sekvencí (případně s vloženými mezerami)

Různé formáty, nejčastěji používán **.aln soubor**, ale též .fasta, aj.

Mnoho programů sloužících pro zobrazení a/nebo editaci
- Bioedit
- JalView
- CINEMA 2.1...
- JavaShade
- …

Výstup - .aln soubor

```
CLUSTAL 2.0.10 multiple sequence alignment


PAIIL           ------------------------------------------------------------
RSIIL           ------------------------------------------------------------
CVIIL           ------------------------------------------------------------
BCLB            ---LVEKLPQYDVFVDIATIPYSFDVGSWQNKVKTDAAGEVVACTVTWAGAPGVLPGAAA
BCLC            AIATNQGVVADGCFTYSSKVPESTGRMPFTLVATIDVGSGVTFVKGQWKSVRGSAMHIDS
BCLA            ------------------------------------------------------------
BCLD            LRETALALRAEVSVLFIRFALKDAGIVAPIELEVRDAATAVPDADDLLHPSCRPLKDHYW


PAIIL           ----------------------------------------------------ATQGVFT
RSIIL           ----------------------------------------------------AQQGVFT
CVIIL           ----------------------------------------------------AQQGVFT
BCLB            KFGVGAVVN----------------YFSKATPQPVQPAPVP--------TGGGERDGIFT
BCLC            YASLSAIWG----------------TAAPSSQGSGNQGAETGGTGAGNIGGGGERDGTFN
BCLA            ------------------------------------ADSQT---------SSNRAGEFS
BCLD            RSDVLAAGATTCTADFAVCDRDGTVSGYFRWETSIEIAGSQPDTKQPGFKPSSDRNGNFS
                                                     * *.

PAIIL           LPANTRFGVTAFANSSGTQTVNVLVNNETA--ATFSGQSTNNAVIGTQVLNSGSSGKVQV
RSIIL           LPANTSFGVTAFANAANTQTIQVLVDNVVK--ATFTGSGTSDKLLGSQVLNSGS-GAIKI
CVIIL           LPARINFGVTVLVNSAATQHVEIFVDNEPR--AAFSGVGTGDNNLGTKVINSGS-GNVRV
BCLB            LPPNIAFGVTALVNSSAPQTIEVFVDDNPKPAATFQGAGTQDANLNTQIVNSGK-GKVRV
BCLC            LPPHIKFGVTALTHAANDQTIDIYIDDDPKPAATFKGAGAQDQNLGTKVLDSGN-GRVRV
BCLA            IPPNTDFRAIFFANAAEQQHIKLFIGDSQEPAAYHKLTTRDGPRE--ATLNSGN-GKIRF
BCLD            LPPNTAFKAIFYANAADRQDLKLFIDDAPEPAATFVGNSEDGVRL--FTLNSKG-GKIRI
                :*..  * .  .:::  * :.: :.:     * .       .      ::*   * ::.
```

Programové balíky

- Existují programy pro pairwise alignment i pro MSA
- Využívají lokální nebo globální alignment nebo příp. kombinaci obou
- Neexistuje univerzální „nejlepší“ program – záleží na konkrétním použití

Pairwise alignment „programy“

Oblasti použití:

- Přímé porovnání dvou sekvencí
- Vyhledávání podobných sekvencí v databázích

emboss Needle & Water

- vytvořeny 1970
  Needleman S.B. and Wunsch C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology 48: 443-453.*
- využívají dynamic programming,
- umožňují vložení mezer

Needle – globální pairwise alignment, Needleman-Wunsch algoritmus

Water – lokální pairwise alignment, Smith-Waterman algoritmus

Globálně podobné sekvence

```
Needle
PA-IIL    1 ATQGVFTLPANTRFGVTAFANSSGTQTVNVLVNNETAATFSGQSTNNAVI 50
            |.||||||||||.|||||||::.|||:.|||:|...|||:|..|:::.::
RS-IIL    1 AQQGVFTLPANTSFGVTAFANAANTQTIQVLVDNVVKATFTGSGTSDKLL 50
PA-IIL   51 GTQVLNSGSSGKVQVQVSVNGRPSDLVSAQVILTNELNFALVGSEDGTDN 100
            |:|||||| ||.:::|||||:|||||.|.||.|:|||:|||||||||||
RS-IIL   51 GSQVLNSG-SGAIKIQVSVNGKPSDLVSNQTILANKLNFAMVGSEDGTDN 99
PA-IIL  101 DYNDAVVVINWPLG 114
                ||||.:.|:|||||
RS-IIL  100 DYNDGIAVLNWPLG 113

Water
PA-IIL    1 ATQGVFTLPANTRFGVTAFANSSGTQTVNVLVNNETAATFSGQSTNNAVI 50
            |.||||||||||.|||||||::.|||:.|||:|...|||:|..|:::.::
RS-IIL    1 AQQGVFTLPANTSFGVTAFANAANTQTIQVLVDNVVKATFTGSGTSDKLL 50
PA-IIL   51 GTQVLNSGSSGKVQVQVSVNGRPSDLVSAQVILTNELNFALVGSEDGTDN 100
            |:|||||| ||.:::|||||:|||||.|.||.|:|||:|||||||||||
RS-IIL   51 GSQVLNSG-SGAIKIQVSVNGKPSDLVSNQTILANKLNFAMVGSEDGTDN 99
PA-IIL  101 DYNDAVVVINWPLG 114
                ||||.:.|:|||||
RS-IIL  100 DYNDGIAVLNWPLG 113
```

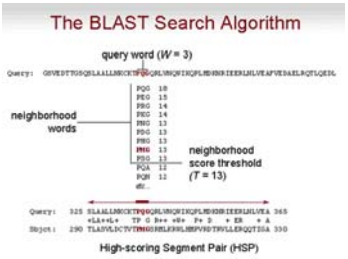
Lokálně podobné sekvence

```
Needle
PA-IIL   1 -------------------------------------------------- 0
BCLB     1 SQPFTHDDLYALLQLAGNDATAVQANGDQAVLDRMRQFMTAQLVEKLPQY 50
PA-IIL   1 -------------------------------------------------- 0
BCLB    51 DVFVDIATIPYSFDVGSWQNKVKTDAAGEVVACTVTWAGAPGVLPGAAAK 100
PA-IIL   1 ---------------------------ATQGVFTLPANTRFGVTAFANS 22
                                        ...|:|||||.|..||||||.||
BCLB   101 FGVGAVVNYFSKATPQPVQPAPVPTGGGERDGIFTLPPNIAFGVTALVNS 150
PA-IIL  23 SGTQTVNVLV--NNETAATFSGQSTNNAVIGTQVLNSGSSGKVQVQVSVN 70
             |..||:.|.| |.:.||||.|..|.:|.:.||::||| .|||:|.|:.|
BCLB   151 SAPQTIEVFVDDNPKPAATFQGAGTQDANLNTQIVNSG-KGKVRVVVTAN 199
PA-IIL  71 GRPSDLVSAQVILTNELNFALVGSEDGTDNDYNDAVVVINWPLG 114
             |:||.:.|.||.:..:..|.|||||||.|.||||.:.::|||||
BCLB   200 GKPSKIGSRQVDIFKKTYFGLVGSEDGGDGDYNDGIAILNWPLG 243

Water
PA-IIL   4 GVFTLPANTRFGVTAFANSSGTQTVNVLV--NNETAATFSGQSTNNAVIG 51
             |:||||.|..|||||..|||..||:.|.| |.:.||||.|..|.:|.:.
BCLB   132 GIFTLPPNIAFGVTALVNSSAPQTIEVFVDDNPKPAATFQGAGTQDANLN 181
PA-IIL  52 TQVLNSGSSGKVQVQVSVNGRPSDLVSAQVILTNELNFALVGSEDGTDND 101
             ||::||| .|||:|.|:.||:||.:.|.||.:..:..|.|||||||.|.|
BCLB   182 TQIVNSG-KGKVRVVVTANGKPSKIGSRQVDIFKKTYFGLVGSEDGGDGD 230
PA-IIL 102 YNDAVVVINWPLG 114
             |||.:.::|||||
BCLB   231 YNDGIAILNWPLG 243
```

BLAST algoritmus

Heuristický algoritmus jehož základem je **hledání slov** (několikapísmenných sekvencí), s dostatečnou podobností (poskytují dostatečně vysoké skóre v substituční matici)



- **Tvorba k-písmenných slov ze vstupní sekvence**
  pro proteiny typicky 3-písmenných (v případě DNA 11-písmenných)
- **Porovnání slov na základě substituční matice**
  algoritmus BLAST hledá na základě vloženého skóre slova, která jsou podobná každému slovu v zadané sekvenci. Vyhovující slova jsou následně uspořádána.
- **Prohledání databázových sekvencí**
  Je hledána shoda s nalezenými vysoce podobnými slovy.
- **Rozšíření slov na segmenty**
  Přesné shody slov s databázovými sekvencemi jsou rozšiřovány oběma směry. To pokračuje dokud skóre pro tuto dvojici sekvencí je dostatečně vysoké.

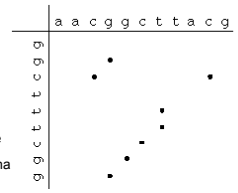
Novější verze BLASTu (BLAST2) má mj. níže nastavenu hladinu pro hledání podobných slov, což rozšiřuje možnost nalezení vzdálenějších homologů.

FASTA algoritmus

FastA algoritmus nejprve provádí rychlé prohledání pro nalezení odpovídajících sekvencí, následuje přesnější porovnání zadané sekvence s databázovou sekvencí. Na rozdíl od algoritmu BLAST jsou zde tolerovány mezery.

Proces:
Obě porovnávané sekvence tvoří horizontální a vertikální osu grafu.
Následně jsou jednotlivá slova z jedné sekvence porovnávána se slovy sekvence druhé. Odpovídající páry pak vytvoří sadu bodů. Body na úhlopříčce signalizují významnou shodu (či podobnost) v daném úseku. Cílem je nalezení nejdelšího shodného úseku (úseku s nejvyšším skóre).

V dalších krocích jsou zahrnuty konzervativní změny pro nejlepší úseky z prvního prohledání. Program pak vyhledává možnost spojení více takových úseků (může mezi nimi být mezera, či jsou na různých diagonálách) a tyto spojené úseky jsou posouzeny z hlediska zadaných kriterií



Příklad porovnání sekvencí GGCTTTCGG a AACGGCTTACG

MSA „programy“

- Za poslednich 10 let vzniklo pres 50 MSA programových balíků (Wallace, I. M., O'Sullivan, O., Higgins, D. G. and Notredame, C. (2006). M-Coffee: combining multiple sequence alignment methods with T-Coffee. Nucleic Acids Res. **34**, 1692-1699.)

- Clustal W (Thompson et al., 1994)
- Clustal X (Thompson et al., 1997)
- Dialign2 (Morgenstern, 1999)
- T-Coffee (Notredame et al., 2000)
- MAFFT (Katoh et al., 2002)
- MUSCLE (Edgar, 2004)
- Kalign (Lassmann, 2005)
- …

Clustal http://www.ebi.ac.uk/clustalw/

- V současné době nejužívanější program
- První verze 1988
  Higgins,D.G. and Sharp,P.M. (1988) CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. Gene, 73, 237–244.
- Dnes používané verze:
  Clustal W (Thompson et al., 1994)
  Clustal X (Jeanmougin et al., 1998)
- Využívá progresivní alignment

ClustalW: Jednotlivým sekvencím přiřazuje váhy (weight – W) podle četnosti zastoupení (čím více jsou si sekvence podobné, tím nižší mají váhu a naopak) a penalizuje přítomnost mezer v závislosti na jejich pozici (position-specific gap penalties)

ClustalW2 – postup

1. Provedení **pairwise alignmentů** pro každou dvojici sekvencí a určení jejich podobnosti – v závislosti na množství neodpovídajících residuí a mezer

2. Sestavení **příbuzenského stromu** (similarity tree)

3. **Kombinace** alignmentů (viz. 1.) v pořadí dle příbuznosti – od nejvíce podobných k nejméně příbuzným (viz. 2.). Jednou vložené mezery jsou zachovány.

Clustal W/Clustal X

Pod alignmentem je uváděn tzv. **consensus** – dohodnuté symboly vyjadřující „konzervovanost“ každého sloupce:

* - identické residuum ve všech sekvencích
: - silně konzervovaný sloupec
. - slabě konzervovaný sloupec

```
IPPNTDFRAIFFANAAEQQHIKLFIGDSQEPAAYHKLTTRDGPRE--ATLNSGNGKIRFE
LPPNTAFKAIFYANAADRQDLKLFIDDAPEPAATFVGNSEDGVRL--FTLNSKGGKIRIE
LPPNIAFGVTALVNSSAPQTIEVFVDDNPKPAATFQGAGTQDANLNTQIVNSGKGKVRVV
LPPHIKFGVTALTHAANDQTIDIYIDDDPKPAATFKGAGAQDQNLGTKVLDSGNGRVRVI
:**:  *.   .:::  * :.:::.*  :*** .     :. .     ::*  *::*.
```

MUSCLE

(**MU**ltiple **S**equence **C**omparison by **L**og-**E**xpectation)
http://www.drive5.com/muscle

Rychlejší určení „vzdálenosti“ dvou sekvencí
Tzv. log-expectation skórovací funkce
Refinement metodou restricted partitioning

Vhodný i pro velký počet sekvencí (5000 seq po 350 bp za 7 min na PC – rok 2004)

Postup:
- Sestavení matice pro každou dvojici sekvencí, určení jejich „vzdálenosti“ a sestavení matice vzdáleností (distance matrix)
- Na základě distance matrix je sestaven první příbuzenský strom (tree1)
- Skládání sekvencí v pořadí dle tree1 od větví ke kmenu – v každém rozvětvení je vytvořen profil, který při dalším porovnávání nahrazuje původní sekvence – výsledkem je první MSA

Algoritmus MUSCLE (podobne PRRP a MAFFT)

4. Přepočítání vzdáleností sekvencí na základě vzniklého MSA1 – tvorba druhé distance matrix (D2)
5. Na základě D2 sestaven vylepšený příbuzenský strom (tree2)
6. Progresivní alignment (viz bod 3) na základě tree2 – vytvoření druhého MSA

---

7. **Refinement** – rozdělení vzniklého stromu na dvě části a vytvoření MSA pro každou z nich. Pokud je výsledný alignment lepší, je zachován. Toto se opakuje do konvergence (žádná další změna nevede k lepšímu výsledku) nebo do určeného počtu kroků

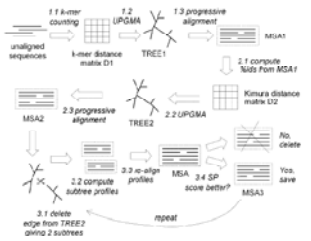


Figure 2. This diagram summarizes the flow of the MUSCLE algorithm. There are three main stages: Stage 1 (draft progressive), Stage 2 (improved progressive) and Stage 3 (refinement). A multiple alignment is available at the completion of each stage, at which point the algorithm may terminate.

Další skórovací schémata (scoring schemes) pro pairwise alignment

Algoritmy založené na matici (matrix-based algorithms) – např. ClustalW, MUSCLE; pomocí substituční matice je příslušné dvojici (AK) přiřazena hodnota. Rozhoduje pouze **identita** těchto dvou **AK**, případně jejich **nejbližší okolí** (viz. např. BLAST)

Schémata založená na konzistenci (consistency-based schemes) – poprvé v T-Coffee, dále v PCMA, ProbCons, MUMMALS, MAFFT, aj. Vychází z nejlepších možných alignmentů každé dvojice sekvencí. Využívá často i **data z různých zdrojů** (např. strukturní informace). Cílem je dosáhnout maximální konzistence (vnitřní shody). Výsledek je přesnější, ale výpočet je časově náročnější.

T-Coffee

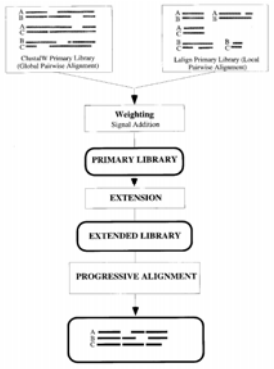
http://www.tcoffee.org/Projects_home_page/t_coffee_home_page.html
(Tree-based Consistency Objective Function for alignment Evaluation)

- Pomalejší ale výrazně přesnější než ClustalW
- Je schopen kombinovat data z více předchozích alignmentů, které mohly být vytvořeny různými postupy (lokální, globální, strukturní podobnost,…)

Hlavním rozdílem oproti tradičním metodám progresivního alignmentu je použití pozičně specifického skórovacího schématu (**extended library**) namísto substituční matice.

T-Coffee

1) Provedení pairwise alignmentů pro všechny dvojice sekvencí pomocí globálního a pomocí lokálního alignmentu (dve primární knihovny).
2) Jednotlivým pairwise alignmentům je přiřazena váha podle poměru počtu identických residuí k celkovému počtu residuí.
3) Kombinace obou knihoven. Pokud je rozdíl v globalním a lokálním alignmentu, jsou zachovány oba s příslušnou váhou. Vzniká pozičně specifická matice (extended library), která je dále použita pro vlastní progresivní alignment.



Zlepšení přesnosti – strukturní informace

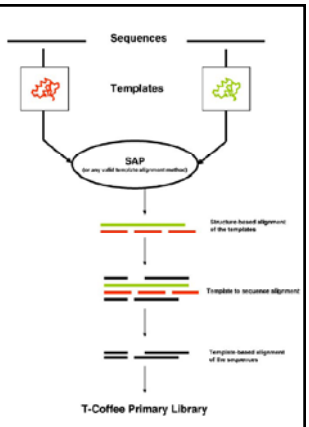
- Sekvence s vyšší homologií (>40%) – vysoká přesnost alignmentu
- Bez homologie – nepoužitelné
- Tzv. twilight zone – málo podobné sekvence (nižší než 20% homologie) = špatná (méně než 30%) přesnost alignmentu

Řešení: nejčastěji využití znalosti strukturní podobnosti (2D nebo 3D), která se behem evoluce zachovává více než sekvence AK.

Rozšíření konzistentního modelu

Template-based alignment metody – vytváří alignment na základě strukturní informace známých homologů jednotlivých sekvencí v PDB databázi nebo získava "profil" sekvence na základě homologních sekvencí (pomocí BLASTu)

Výhoda: vyšší přesnost



Expresso

- Je založeno na 3DCoffee

Expresso je MSA server, který srovnává sekvence za užití strukturní informace. Po zadání sekvencí vyhledá v databázi struktur (PDB) pomocí BLASTu homology a použije je jako templáty pro následný alignment zadaných sekvencí pomocí metod MSA založených na struktuře (např. SAP, Fugue).

Benchmark (srovnávací testy)

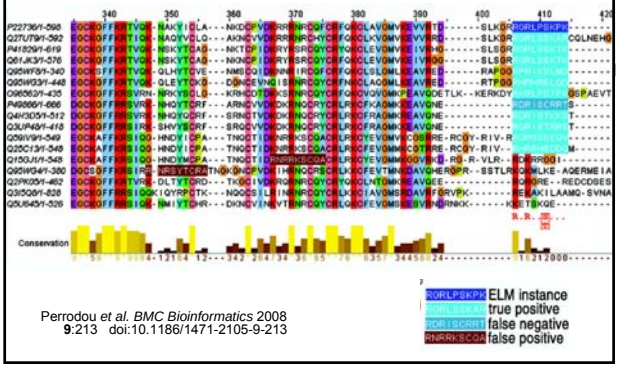
BAliBASE - První vytvořená sada benchmarkových testů pro multiple alignment programy (Thompson et al., 1999) – byla vytvořena pomocí manuálně provedeného alignmentu

Na základě srovnání 3D struktur byly vytvořeny další sety:
HOMSTRAD [Mizuguchi *et al.*, 1998].
OxBench [Raghava *et al.*, 2003]
PREFAB [Edgar, 2004]

Existují i specificky zaměřené benchmarkové sety, např.

IRMBASE [Subramanian *et al*, 2005] – náhodné (nepřiložitelné) sekvence s vloženými motivy. Slouží k testování metod pro lokální alignment

BaliBASE – ukázka alignmentu



Perrodou *et al. BMC Bioinformatics* 2008 **9**:213 doi:10.1186/1471-2105-9-213

Zopakování / shrnutí

- **Alignment** – přiložení sekvencí (2 nebo více) na základě podobnosti
- **Využití** pro hledání příbuznosti sekvencí, tvorba profilů proteinových rodin, aj.
- Řada **programů** využívajících rozdílné přístupy – použití závisí na vstupních datech a účelu
- Nejčastěji používaný (ClustalW) neznamená nejpřesnější – každý program je **kompromisem mezi přesností a rychlostí**
- Každý alignment potřebuje **lidskou kontrolu** !!!

Local alignment

- For two-sequence comparisons, there is the well-known Smith and Waterman (1981) algorithm. Here we use Lalign
- For multiple sequences, the Gibbs sampler (Lawrence et al., 1993) and Dialign2 (Morgenstern, 1999) are the main automatic methods. These programs often perform well when there is a clear block of ungapped alignment shared by all of the sequences. They perform poorly, however, on general sets of test cases when compared with global methods

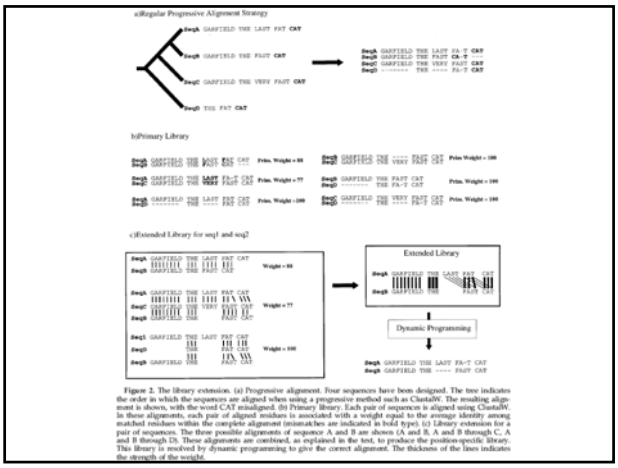


Figure 2. The library extension. (a) Progressive alignment. Four sequences have been designed. The tree indicates the order in which the sequences are aligned when using a progressive method such as ClustalW. The resulting alignment is shown, with the word CAT misaligned. (b) Primary library. Each pair of sequences is aligned using ClustalW. In these alignments, each pair of aligned residues is associated with a weight equal to the average identity among matched residues within the complete alignment (mismatches are indicated in bold type). (c) Library extension for a pair of sequences. The three possible alignments of sequence A and B are shown (A and B, A and B through C, A and B through D). These alignments are combined, as explained in the text, to produce the position-specific library. This library is resolved by dynamic programming to give the correct alignment. The thickness of the lines indicates the strength of the weight.

BAliBASE [Thompson et al., 1999] contains eight reference sets, each dealing with a different type of alignment problem. Ref1 deals with test cases containing small numbers of equidistant sequences, and is further subdivided by percent identity. Ref2 alignments contain "orphan", or unrelated, sequences. Ref3 test cases contain a pair of divergent subfamilies, with less than 25% identity between the two groups. Ref4 is concerned with long terminal extensions, while Ref5 test cases contain large internal insertions and deletions. Test sets from references 6-8 deal with problems like transmembrane regions, inverted domains, and repeat sequences. In previous versions of BAliBASE, test cases were confined to homologous regions. In practice, the boundaries of such regions may be unknown. The current version [Thompson et al., 2005] now also provides duplicate test cases containing full-length sequences. Only the first five reference sets are used here, as they have been corrected and verified in the latest release.

OxBench [Raghava et al., 2003] comprises 3 related datasets. Test cases in the MASTER set deal with isolated domains derived exclusively from sequences of known structure. The FULL set was generated from suitable MASTER test cases, using full-length sequence data. High scoring homologous sequences were added to each MASTER test case to generate the EXTENDED set. The results from this third set, however, are not used here. It was found that some of the test cases in the EXTENDED set proved too large for some programs, and aborted due to excessive memory requirements. Of the 276 test cases selected from EXTENDED, T-COFFEE returned 235 alignments, and Align-m was only able to align 107, using a single processor with 4GB of RAM.

PREFAB [Edgar, 2004] test cases are generated by taking a pairwise alignment of sequences of known 3D structure, and adding up to 24 high scoring homologues for each sequence. Accuracy is assessed on the structural alignment of the original pair alone.

SABmark [Van Walle et al., 2005] is divided into two subsets. Each test group in the SUPERFAMILY set represents a SCOP superfamily, whose sequences are 25-50% identical. Each test group in the TWILIGHT set represents a common SCOP fold and sequences are 0-25% identical. In addition, these two subsets are also provided with non-homologous (false positive) sequences included within each group. Instead of a single alignment acting as a reference, SABmark provides multiple pairwise references for each test, and it is the average score from each of these references that is taken here as a score for each test case.

IRMBASE [Subramanian et al., 2008] test cases contain a number of simulated motifs [Sheye et al., 1998] inserted into otherwise random (unalignable) sequences, and as such is entirely different to the other benchmarks used in this study. Test cases are designed to examine whether a method can detect isolated motifs within sequences, and so are tailored to a local alignment approach.

HOMSTRAD [Mizuguchi et al., 1998] is a database exclusively based on protein structures derived from the PDB, arranged into homologous protein families. It was not specifically designed as a benchmark database, although it is regularly employed as such.

Method	Score	Templates	Validation Values		Server
			Prefab	HOMSTRAD	
ClustalW [14]	Matrix	—	61.66 [12]	—	http://www.ebi.ac.uk/clustalw/
Kalign	Matrix	—	63.00 [18]	—	http://msa.cgb.ki.se/
MUSCLE [8]	Matrix	—	66.00 [16]	45.0 [9]	http://www.drive5.com/muscle/
T-Coffee [10]	Consistency	—	69.97 [12]	44.0 [9]	http://www.tcoffee.org/
ProbCons [7]	Consistency	—	70.54 [12]	—	http://probcons.stanford.edu/
MAFFT [8]	Consistency	—	72.20 [12]	—	http://align.genome.jp/mafft/
M-Coffee [12]	Consistency	—	72.91 [12]	—	http://www.tcoffee.org/
MUMMALS [16]	Consistency	—	73.10 [16]	—	http://predata.swmed.edu/mummals/
DbClustal [24]	Profiles	—	—	—	http://bips.u-strasbg.fr/PipeAlign/
PRALINE [9]	Matrix	Profiles	—	50.2 [9]	http://zeus.cs.vu.nl/programs/pralinewww/
PROMALS [16]	Consistency	Profiles	79.00 [16]	—	http://prodata.swmed.edu/promals/
SPEM [26]	Matrix	Profiles	77.00 [26]	—	http://sparks.informatics.iupui.edu/Software-Services_files/spem.htm
Expresso [13]	Consistency	Structures	—	71.9 [11]*	http://www.tcoffee.org/
3-Lara [29]	Consistency	Structures	—	—	https://www.mi.fu-berlin.de/w/LiLA/

Validation values were compiled from several sources, and selected for comparability. Prefab validations were made using Prefab version 3. HOMSTRAD validations were made on datasets having less than 30% identity. The source of each value is indicated by the accompanying reference citation.
*The Expresso value comes from a slightly more demanding subset of HOMSTRAD (HOM29) made of sequences less than 25% identical.

Table 1: Programs used in this investigation.

Method	OVERVIEW
Align_m (2.3) [Van Walle et al., 2004]	http://bioinformatics.vub.ac.be/software/software.html Local, specialised for highly divergent sequences.
ClustalW (1.8) [Thompson et al., 1994]	http://www.ebi.ac.uk/clustalw/ Globa , progressive alignment package.
Dialign2 (2.2) [Morgenstern, 1999]	http://bibiserv.techfak.uni-bielefeld.de/dialign/ Local, aligns segments of sequences rather than individual residues.
Dialign-t (0.1.3) [Subramanian et al., 2005]	http://dialign-t.gobics.de/ Local, progressive alignment. Recent re-implementation of Dialign2.
MAFFT (6.531) [Katoh et al., 2002]	http://www.biophys.kyoto-u.ac.jp/~katoh/programs/align/mafft/ Suite of alignment programs:
FFTNS	Globa , uses Fast Fourier Transform to generate tree.
FFTNSi	As FFTNS, but with iteration step to refine alignment.
NWNS	Globa , uses traditional Needleman-Wunsch algorithm.
NWNSi	As NWNS, but with iteration step to refi
FINSi	Local, iterative, uses local pairwise alig
GINSi	Global, iterative uses global pairwise alignment information.
MUSCLE (3.0) [Edgar, 2004]	http://www.drive5.com/muscle/ Global, iterative progressive alignment program that uses Log Expectation as scoring function
ProbCons (1.09) [Do et al., 2005]	http://probcons.stanford.edu/ Global, uses posterior-probabilities from HMMs and pairwise alignment consistency.
PCMA (2.0) [Pei et al., 2003]	ftp://cole.swmed.edu/pub/PCMA/ Global, switches alignment strategies dependent on sequence data Clustalw is used to align highly simiar sequences and to form pre-aligned groups. T-COFFEE is used to align the more divergent groups.
POA (v2) [Lee et al., 2002]	http://www.bioinformatics.ucla.edu/poa/ Local; uses Partial Order graphs.
T-COFFEE (1.37) [Notredame et al., 2000]	http://igs-server.cnrs-mrs.fr/~cnotred/Projects_home_page/t_coffee_home_page.html Combines both global and local methods; uses consistency.

Blackshield 2006 oznacil ProbCons jako nejlepsi na zaklade 6 benchmarkovych testu