Heurističke metode pretraživanja baza podataka

1. BLAST
2. FASTA

Obnavljanje
Poravnanje sekvenci – egzaktne metode

- Globalno
  - Poredimo dve sekvence celom dužinom
  - Pojava uzastopnih praznina je loša
  - Pogodno kada poredimo sekvence približnih dužina

Needleman-Wunsch algoritam

- Lokalno
  - Poredimo delove sekvenci
  - Pojava uzastopnih praznina ne mora da bude loša
  - Pogodno kada poredimo manju sekvencu sa većom sekvencom

Smith-Waterman algoritam

Heurističke metode poravnanja sekvenci

- Kada je neophodno da izvršimo poravnanje jedne sekvence sa celom bazom sekvenci, egzaktne metode su suviše spore – *polinomijalna složenost O(mn), gde su m i n dužine sekvenci koje poravnavamo*
- Rešenje: korišćenje heurističkih
- Dva algoritma za heurističko poravnanje:
  - FASTA, 1988 (prva verzija)
  - BLAST, 1990 (prva verzija)

FASTA i BLAST – osnovna ideja

- *Osnovna ideja*: dobro poravnanje između dve sekvence sadrži kratke podsekvence koji su identični

- Opšti postupak kod oba algoritma:
  - Odrediti kratka identična poklapanja između dve sekvence
  - Od svih takvih poklapanja, izabrati najbolje na osnovu nekog skora; najbolja poklapanja proširiti sa obe strane dokle god je moguće
  - Optimizovati najbolje pogotke

FASTA

Ulaz: dve sekvence, dužina reči k, matrica supstitucije

- Konstrukcija tačkastog dijagrama (*dot plot*)
- Računanje dijagonalnih suma
- Računanje skora dijagonala na osnovu matrica supstitucije
- Spajanje dijagonala
- Izgradnja lokalnog poravnanja

Izlaz: init1, initn, opt

FASTA – KORAK 1

- Preprocesiramo datu sekvencu:

S=ATCGTATCG, k=3

ATCGTATCG

1 ATC

2 TCG

3 CGT

4 GTA

5 TAT

6 ATC

7 TCG

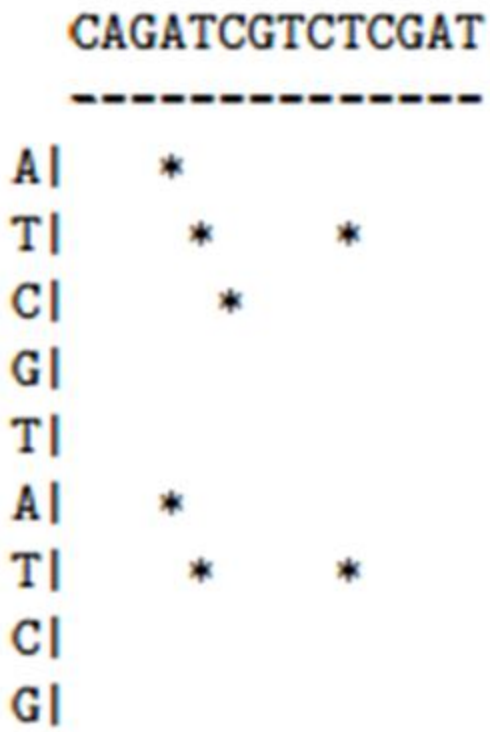
$L_S[ATC] = \{1,6\}$
$L_S[CGT] = \{3\}$
$L_S[GTA] = \{4\}$
$L_S[TAT] = \{5\}$
$L_S[TCG] = \{2,7\}$

FASTA – KORAK 1

- Potražimo detektovane *k*-torke u sekvenci iz baze

T = CAGATCGTCTCGAT

$L_S[ATC] = \{1,6\}$ $L_T[ATC] = \{4\}$
$L_S[CGT] = \{3\}$ $L_T[CGT] = \{6\}$
$L_S[GTA] = \{4\}$ $L_T[GTA] = \{\}$
$L_S[TAT] = \{5\}$ $L_T[TAT] = \{\}$
$L_S[TCG] = \{2,7\}$ $L_T[TCG] = \{5,10\}$



FASTA – KORAK 2

- Izračunati dijagonalne sume

3-tuple	L_S	L_T	Diagonals
ATC	$\{1, 6\}$	$\{4\}$	$1-4=-3;\ 6-4=+2$
TCG	$\{2, 7\}$	$\{5, 10\}$	$2-5=-3;\ 2-10=-8;\ 7-5=+2;\ 7-10=-3$
CGT	$\{3\},$	$\{6\}$	$3-6=-3$

d	Diag(d)
-8	1
-3	4
2	2

FASTA – KORAK 2

```
Algorithm DiagonalScores(S, T, L_S[], L_T[], k)
begin
  declare Diag[−n..m];          // array of diagonal scores
  for i = −n to m do
    Diag[i] := 0;               // initialize all diagonal scores to 0
  end for
  for each k such that L_S[k] ≠ ∅ and L_T[k] ≠ ∅ do
                                // find all matches in the dot plot
    for each i ∈ L_S[k] do
      for each j ∈ L_S[k] do
        d := i − j;             // determine the diagonal for the match
        Diag[d] := Diag[d] + 1; // update the diagonal score
      end for
    end for
  end for
  return Diag[];
end
```

FASTA – KORAK 3

- Svaka dijagonala pronađena u koraku 2 predstavlja jedno lokalno poravnanje u kom se mogu pojaviti neslaganja (*mismatches*) ali se ne mogu pojaviti praznine (*gaps*)
- Izračunamo skor dobijenih poravnanja na osnovu zadate matrice supstitucije – **init1** skor
- Na primer, neka je kod datih nukleotidnih sekvenci match skor 5, a mismatch -4

FASTA – KORAK 3

- Dijagonala 3:

```
    ATCGTATCG
     | | | | |   | | |
cagATCGTCTCGat
```

5+5+5+5+5−4+5+5+5 =36

- Dijagonala -2:

```
atcgtATCG
      | | | |
  cagATCGtctcgat
```

5 + 5 + 5 + 5 = 20

- Najbolji takav skor: **init1**

FASTA – KORAK 4

- Pokušavamo spajanje datih dijagonala – uvodimo praznine u poravnanja i računamo skor tako dobijenog poravnanja:

  sum(init1) – skor_praznine*#praznina

- Najbolji takav skor: **initn**
- Optimizacioni problem

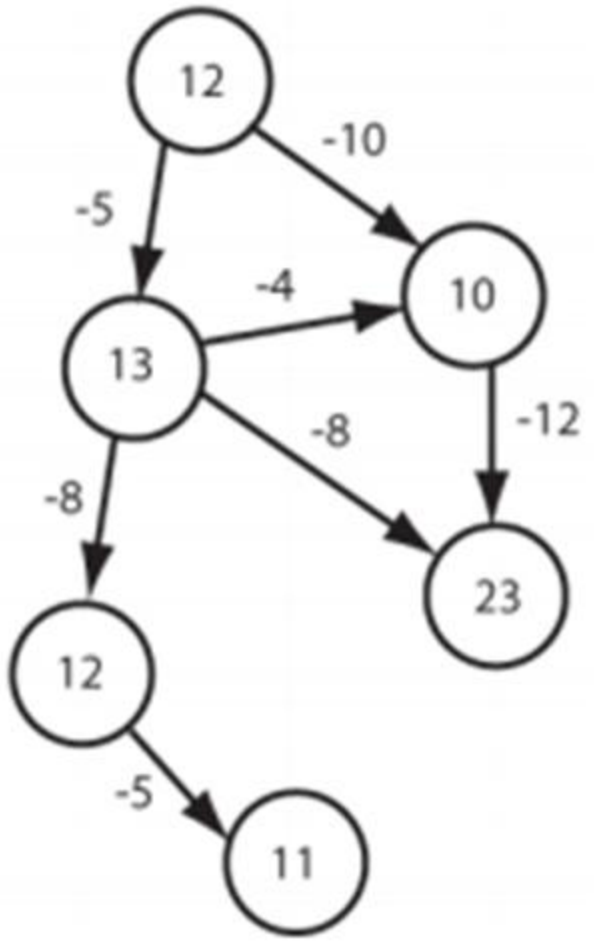
koji se može formulisati

grafovski

FASTA – KORAK 4

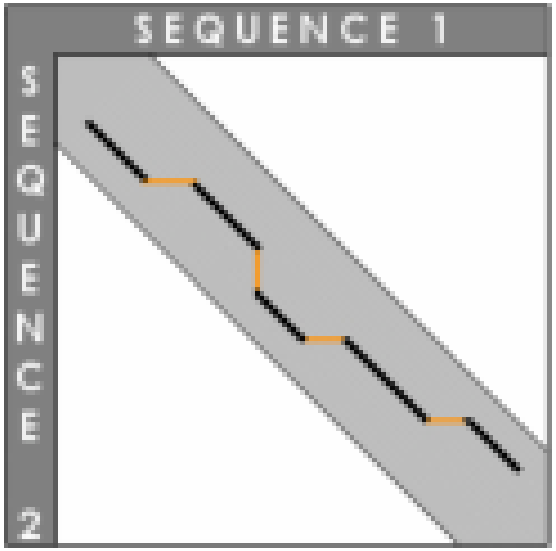
Usmereni aciklički graf (čvorovi – poravnanja dobijena u prethodnim koracima, grane – ako je poravnanje v nakon poravnanja u, spojiti ih granom i dodeliti im težinu praznina*skor_praznine)

Problem traženja puta najveće težine



FASTA – KORAK 5

- Primeniti algoritam lokalnog poravnanja (Smith-Waterman) u nekoj okolini (najviše K pozicija) poravnanje iz koraka 2– **opt** skor



FASTA

- http://fasta.bioch.virginia.edu/fasta_www2/fasta_list2.shtml
- http://www.ebi.ac.uk/Tools/sss/fasta/

FASTA

- Izračunati dijagonalne sume za sekvence

I: CGTATGCG

J: ATGCGTATGTAC

$S_l := 0$ for all $1 - m \leq l \leq n - 1$

Compute $L_w(J)$ for all words w

for i := 1 to n – k – 1 do

  $w := I_i I_{i+1} \ldots I_{i+k-1}$

  for $j \in L_w(J)$ do

    $l := i - j$

    $S_l := S_l + 1$

  end

end

BLAST

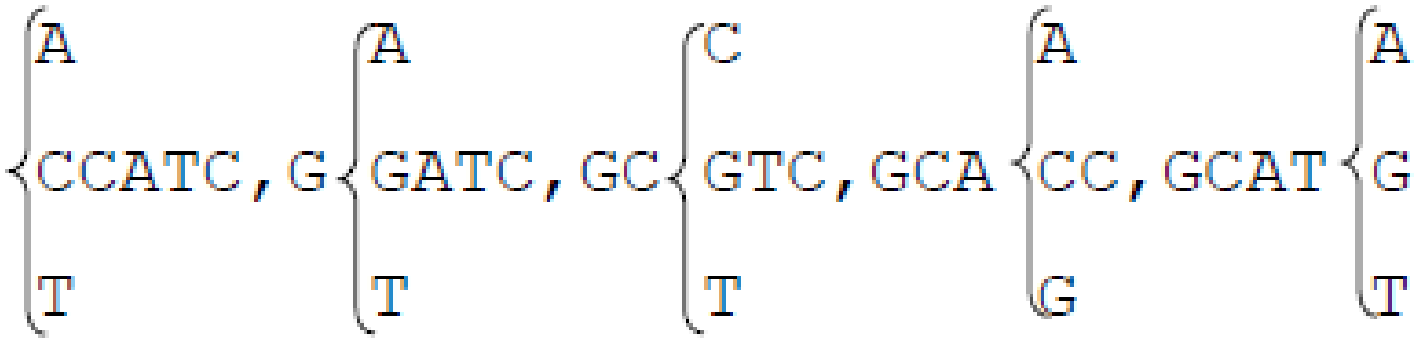
1. Naći lokalna poravnanja između upitne sekvnce i sekvence iz baze podataka (*semeni pogoci*)
2. Proširiti semene pogotke u lokalna poravnanja najvišeg mogućeg skora (*segmentni parovi*)
3. Rangirati dobijena lokalna poravnanja

BLAST – TRAŽENJE UPITNIH REČI

- Pronaći za upitnu sekvencu I=GCATCGGC sve upitne reči dužine k=5 ako je match skor 1, mismatch skor 0 i prag skora poravnanja T=4.

BLAST – TRAŽENJE UPITNIH REČI

- k-reči iz upitne sekvence su: GCATCGGC, GCATCGGC, GCATCGGC, GCATCGGC
- Tražimo susedne sekvence za reč GCATC; s obzirom na vrednost match skora, mismatsh skora i praga, u obzir dolaze sekvence koje se za najviše 1 razlikuju od GCATC:



BLAST – PROŠIRENJE PORAVNANJA

- Ako je upitna reč "*T*" a prag proširenja 5, proširiti poravnanje udesno između dve date sekvence

The quick brown fox jumps over the lazy dog.

The quiet brown cat purrs when she sees him.

BLAST - PROŠIRENJE

The quick brown fox jump

The quiet brown cat purr

123 45654 56789 876 5654 <- score

000 00012 10000 123 4345 <- drop off score

BLAST – EVALUACIJA DOBIJENIH PORAVNANJA

- Poravnanja se ocenjuju na osnovu date matrice skora (za proteinske sekvence) ili na osnovu datih match/mismatch skorova (za nukleotidne sekvence)
- Za svako poravnanje procenjujemo koliko je statistički značajno na osnovu *e-vrednosti* – očekivanje da ćemo u bazi slučajnih sekvenci te veličine naići na pogodak sa sličnim skorom; e-vrednost zavisi od veličine baze sekvenci (što je veća baza, to je veća verovatnoća pronalaženja poravnanja sa datim skorom) i od dužine upitne sekvence (što je kraća sekvenca, to je opet veća verovatnoća pronalaženja)
- Što je e-vrednost manja, to je poravnanje statistički značajnije

BLAST - NCBI

- Blast-ovanje proteina gi|129295
- Izlaz: **Descriptions**

Max score – maksimalni skor svih lokalnih poravnanja dobijenih iz date sekvence

Total score – ukupni skor poravnanja – zbir svih skorova lokalnih poravnanja dobijenih iz date sekvence (ako postoji samo jedno lokalno poravnanje upitne sekvence i sekvence iz baze, onda je max score jednak sa total score)

Query coverage – procenat upitne sekvence pokriven datim poravnanjem

Max ident – procenat sekvence iz baze koji je identičan sa upitnom sekvencom

BLAST - NCBI

- Izlaz: **Alignments**

Get selected sequence, distance tree of results, Multiple alignment