Fast Local RNA Alignment

Andrey Mironov
(Moscow State University)

Benasque 2015

Motivation

- Simultaneous alignment and structure prediction is very slow (Sankoff algorithm and its modifications)

- Most algorithms (especially for multiple alignment) produces global alignments: we need exact boundaries of the sequences

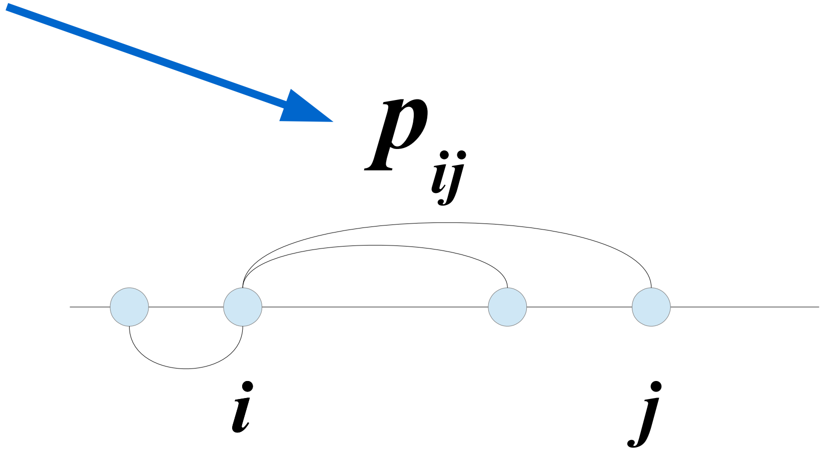
Idea1. Not to work with structure

- The bracket-dot presentation of the structure is a ***String***
- Analise the structure before the alignment
- Use the result of structure analysis in the alignment scoring
  - Use the probabilities of pairing (pFold)

$$p_i^R = \sum_{j>i} p_{ij}$$

$$p_i^L = \sum_{j<i} p_{ji}$$

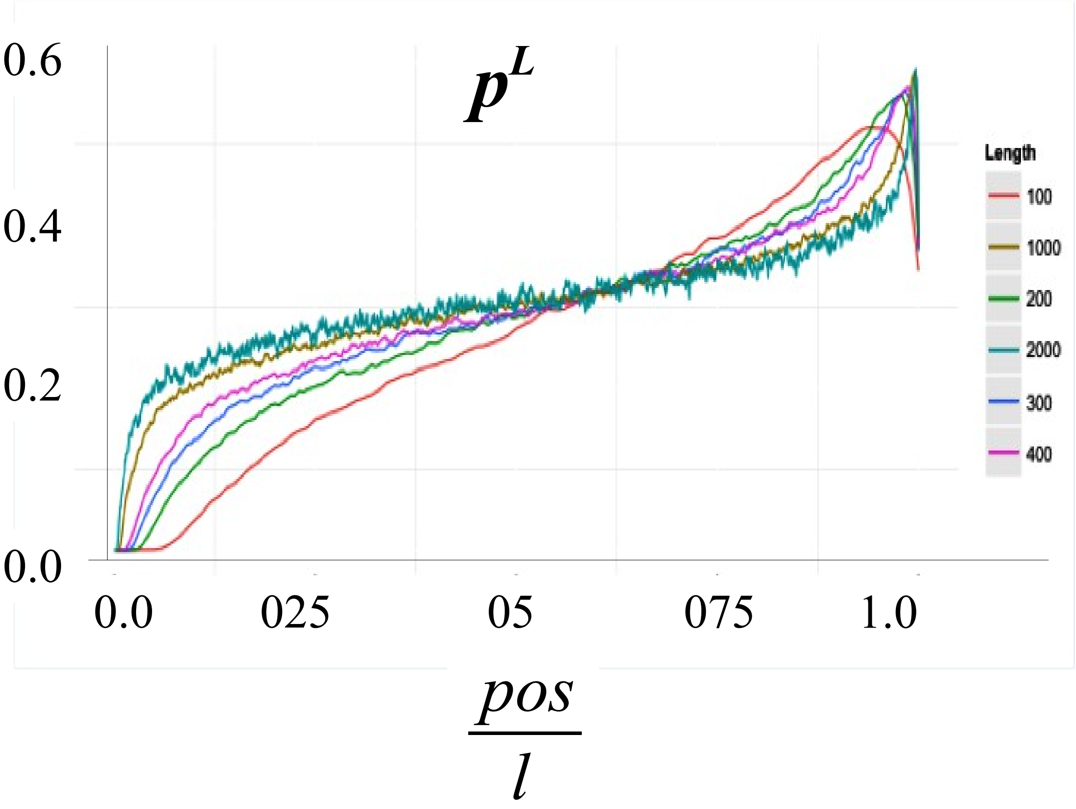
$$p_i^U = 1 - p_i^L \cdot p_i^R$$



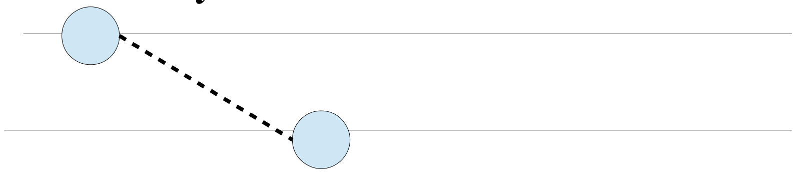
If $\{\boldsymbol{p}^L, \boldsymbol{p}^R\}$ are similar the structures seems to be similar

The p^L and p^R can not be used directly

- The distributions of the $\boldsymbol{p^L}$ and $\boldsymbol{p^R}$ depend on sequence length and positions



The $\boldsymbol{p_i^L}$ is small due position



The $\boldsymbol{p_j^L}$ can be small due structure

The $\boldsymbol{p_i^L}, \boldsymbol{p_i^L}$ are incomparable!

Rescaling of p^L, p^R

- We want to do a transformation of $\boldsymbol{p^L}$, $\boldsymbol{p^R}$ to get a values with standard distributions that do not depend on the position and length.
- The ***cdf*** is uniformly distributed!
- The ***cdf*** can be fitted by:

$$cdf(x) = \alpha x^{b1} + (1-\alpha)\left(1-(1-x)^{b2}\right)$$

$$\alpha = \alpha(pos); \quad b2 = b2(pos); \quad b1 = b1(pos, l)$$

Scoring

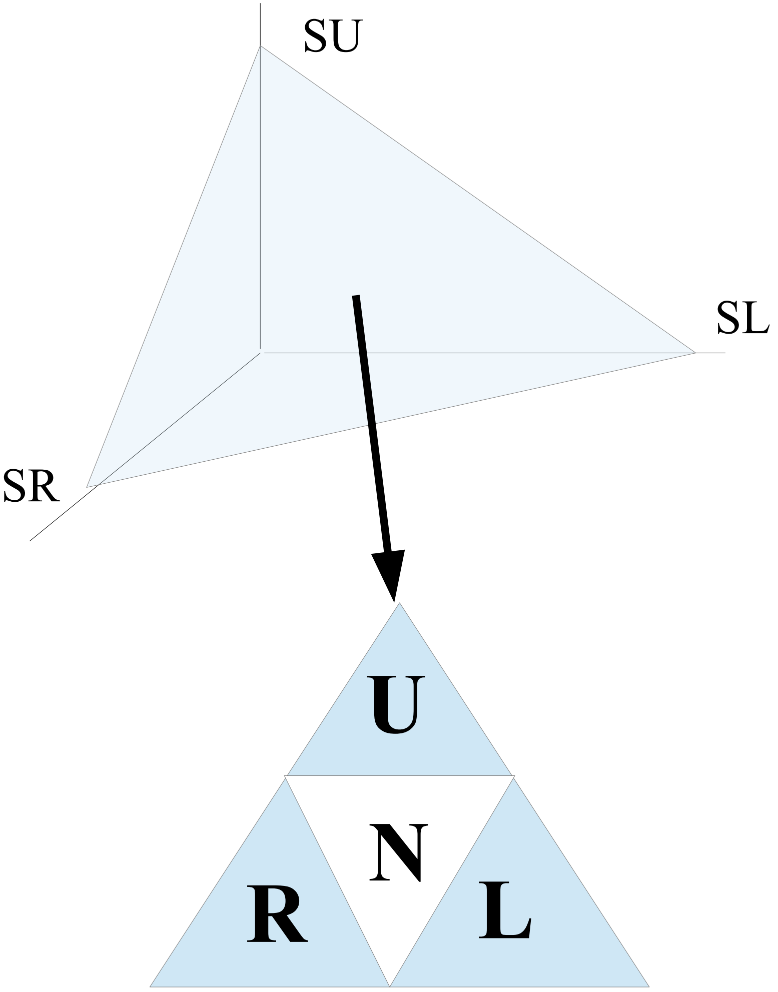
$$W_{ij} = \alpha \cdot S_{ij}^{seq} + (1-\alpha) \cdot S_{ij}^{str}$$
$$S_{ij}^{str} = SL_{ij} + SR_{ij} + SU_{ij}$$

- S_{ij} are calculated as log-likelihood:

$$S_{ij}^{seq} = \log\left(\frac{p(s_i, s'_j)}{p(s_i)\,p(s'_j)}\right)$$
$$SL_{ij} = \log\left(\frac{p(s_i^L, s'^L_j)}{p(s_i^L)\,p(s^{L\prime}_j)}\right)$$
$$etc\ldots$$

Idea 2. Non-progressive multiple alignment

- Do BLAST-like alignments between all sequences and find HSP
- Convert structure information to 4-letter alphabet
- Do BLAST-like alignments between all sequence structure signatures



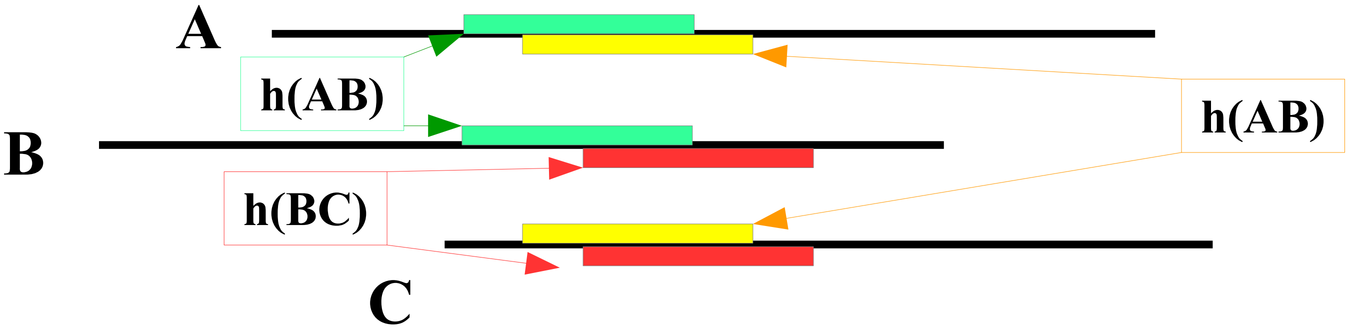
High Scoring Segments (HSP)

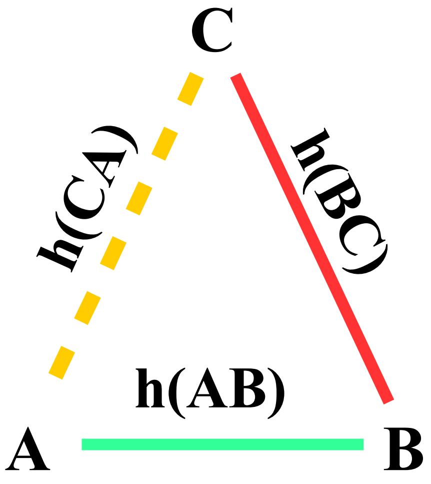
<u>Definition1</u>

$$HSP(A,B)=\{fA, tA; fB, tB\}$$
$$diag(HSP)=fA-fB$$

<u>Definition2.</u> *Two HSPs h(AB),h2(BC) for sequence pairs A~B and B~C are **<u>compatible</u>** if there exist HSP h(CA) for pair C~A that:*

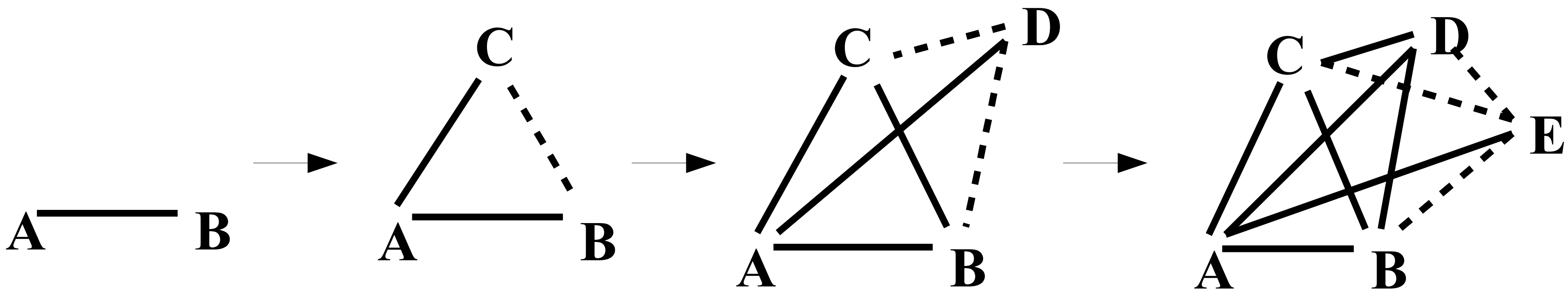
$$diag(h(AB))+diag(h(BC))+diag(h(CA))=0$$
$$iv(A,h(AB))\ overlaps\ iv(A,h(CA))$$
$$iv(B,h(AB))\ overlaps\ iv(B,h(BC))$$
$$iv(C,h(CA))\ overlaps\ iv(C,h(BC))$$





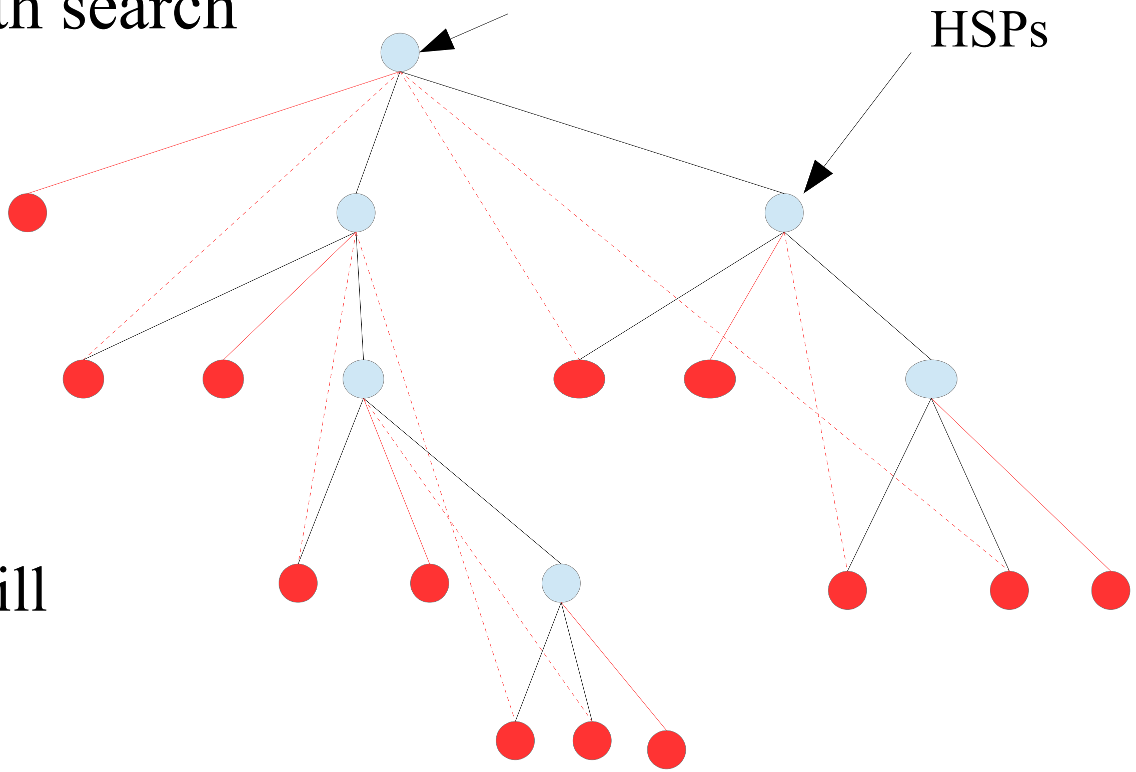
Search for sets of compatible HSPs (consensus set)

- Select a pair of the sequences (A,B)
- Select next HSP h1(A,B)
  - Select next HSP h2(AC) that is compatible with h1
    - Select next HSP h3(AD) that is compatible with h1 and h2
- …



This is a clique problem (NP-hard), BUT...

The algorithm is in-depth search

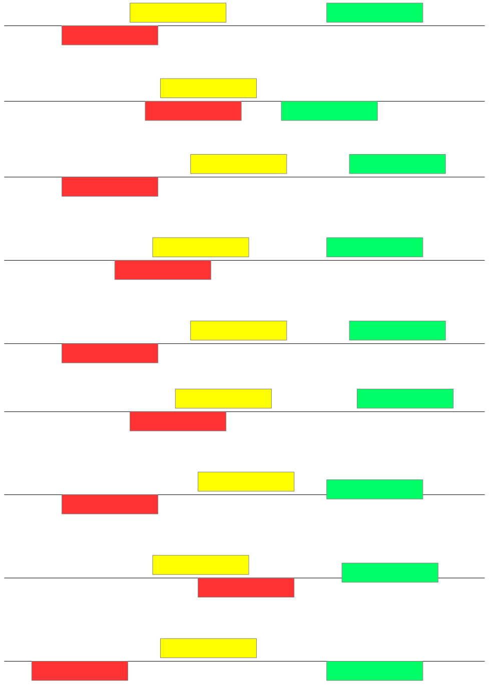


On every level we do more comparisons and have more chances to kill the recursion

Theoretically the expected number of iterations on a random sequences tends to a constant when number of sequences tends to infinity

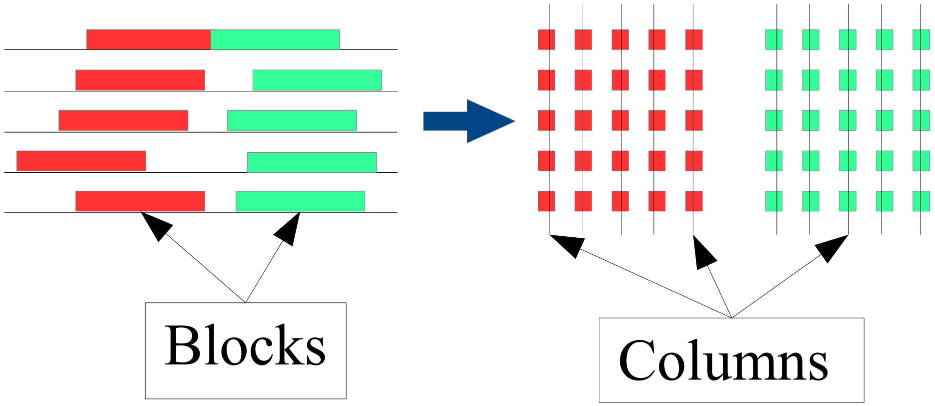
Algorithm: search for blocks

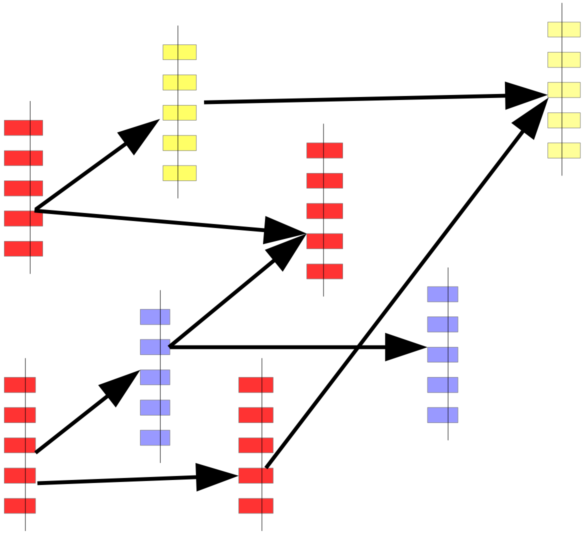
- Calculate $\boldsymbol{p}^L$, $\boldsymbol{p}^R$
- Transform structure information to structure alphabet
- Do BLAST-like search using sequences and structure
- Select combinations of HSPs that are compatible for all pairs of the sequences
- Search for consensus HS blocks (HSB)



Algorithm: alignment

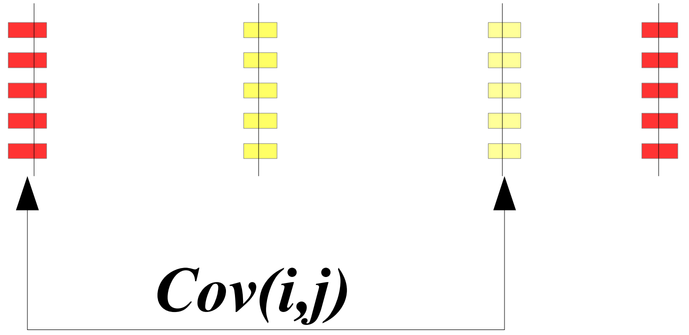
- Decompose HSBs to a set of columns
- Column Graph (CG):
  - Vertices = columns
  - Edge $\boldsymbol{e=(u\rightarrow v)}$ if for all sequences position $\boldsymbol{i}$ $\boldsymbol{v}_i > \boldsymbol{u}_i$; $\boldsymbol{v}_i$, $\boldsymbol{u}_i$ position on sequence #$\boldsymbol{i}$
- Do Dynamic Programming on CG and find the optimal alignment





If you have enough time

- Calculate covariance between columns



- Reconstruct optimal common structure and produce the alignment simultaneously (to be done)

Preliminary results

Without covariance

- tRNA with random flanks
- Identity 30-60%
- Quality (number of correctly aligned positions) = 80%
- Time for 20 sequences 2 s.

Variants

Variant 1

- Find HSS
- Near found diagonals do ProbCons-like alignment

Variant 2

- Do Nussinoff-style algorithm on columns

Variant 3

- Do hash-based partition function calculation

Team

- Svetlana Vinoradova (MSU)
- Michkael Roytberg (IMPB RAS, Puschino)
- Andrey Mironov

Funding:

Russian science foundation