Protein sequence alignment via anti-translation

Marta Gîrdea, Gregory Kucherov and Laurent Noé

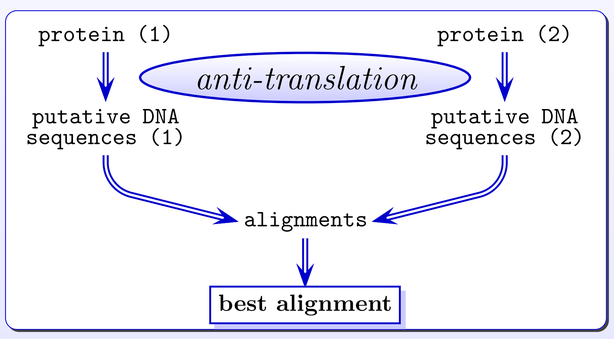
LIFL/CNRS and INRIA-Nord Europe

{Marta.Girdea|Gregory.Kucherov|Laurent.Noe}@lifl.fr

Objective

The design of a method which can detect protein homologies based on similarities between their putative DNA sequences.

Approach



Motivation

Traditional protein alignment methods, which consist of aligning amino-acid pairs, fail to reveal protein relations when the divergence is caused by **frameshift mutations**. By aligning the putative DNA, we take into account both

- point mutations which only affect one codon
- **frameshift mutations** (insertions / deletions of bases) that can alter the reading frame of the ribosomes, affecting all the amino-acids coded after the frameshift.

Related work

- (Leluk, 2000) the alignment score of an amino-acid pair is computed from the base pair alignment of the respective codons
- (Pellegrini et al, 1999) several substitution matrices were designed for aligning proteins with a frameshift

The plus of our approach

More flexibility for managing point and frameshift mutations in the putative DNA sequences for an expressive alignment

Data structures

There are too many putative DNA sequences for a given protein

- their number increases exponentially with the protein's length;
- an explicit enumeration and pairwise alignment is not an option.

⇒ The protein's **"anti-translation"** (set of putative DNA sequences) is represented as **a graph**, GRAPH EXAMPLES ⟶

- which can be seen as a generalised sequence:
  several symbols on each position,
  precedence constraints (marked by arcs) between symbols on consecutive positions.

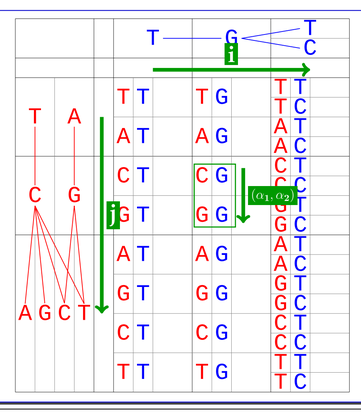
Alignment Algorithm

Smith-Waterman extended for bi-dimensional data structures (anti-translation graphs).

Dynamic programming matrix

Given the input graphs A and B, it fills a *3D alignment matrix* M, where, for $M[i, j, (\alpha_1, \alpha_2)]$:

- i and j iterate on the columns of the first and second graph
- (α_1, α_2) enumerates on all possible pairs of symbols (nodes) from $A[i]$, and $B[j]$

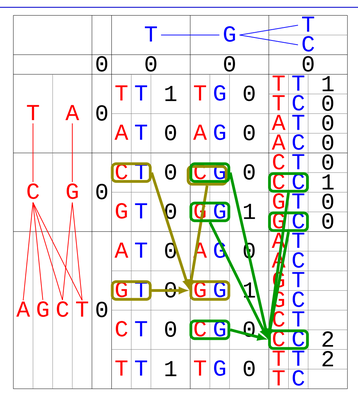


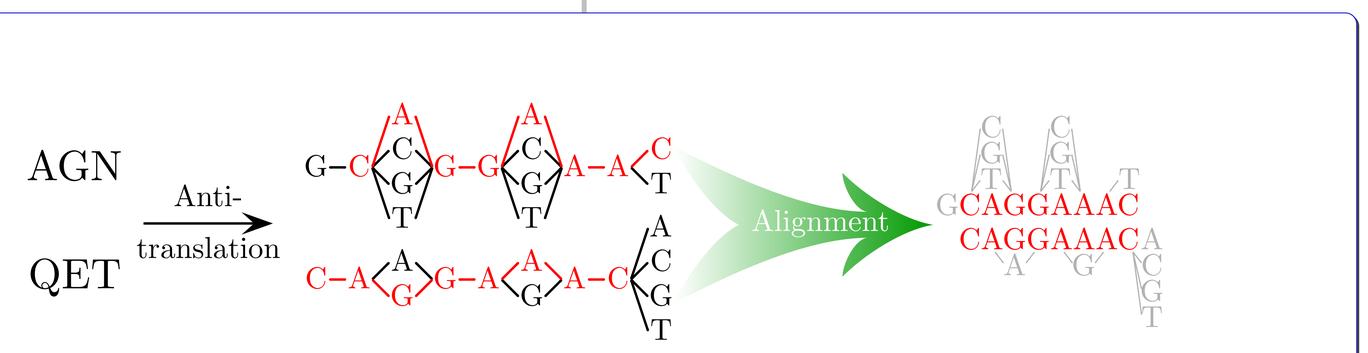
Partial score computation formula

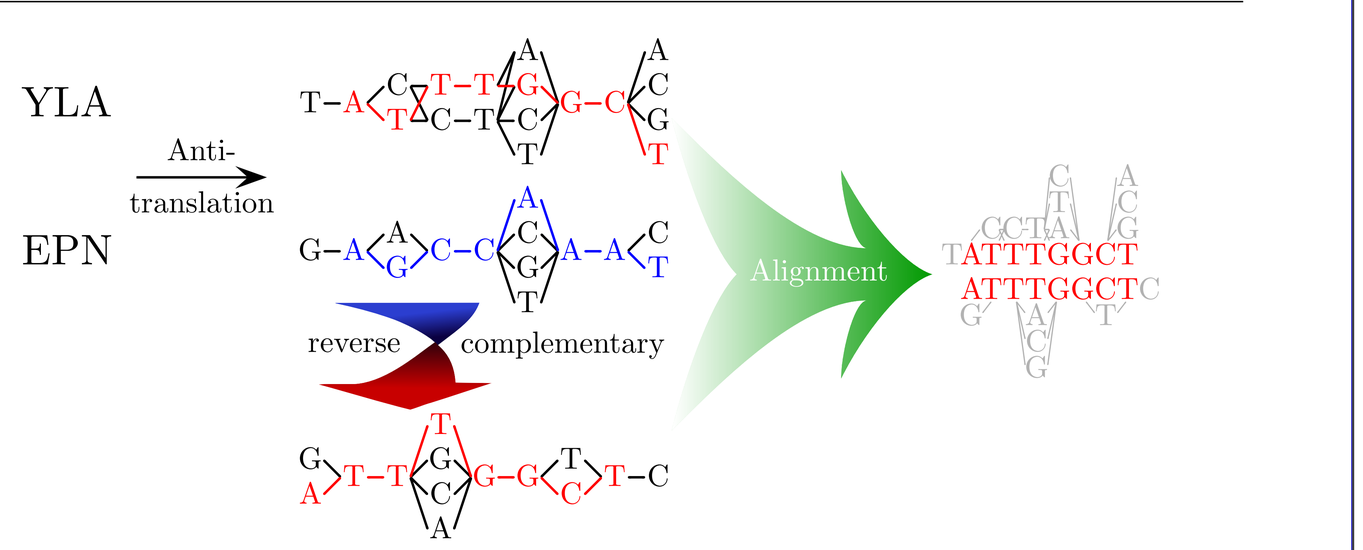
$$M[i,j,\alpha_1\alpha_2] = \begin{cases} 0 \\ M[i-1,j-1,(\beta_1,\beta_2)] + score(\alpha_1,\alpha_2)), \\ \qquad \beta_k \in pred(\alpha_k), k \in 1,2 \\ M[i-1,j,(\alpha_1,\beta_2)] + gap\_penalty, \\ \qquad \beta_2 \in pred(\alpha_2) \\ M[i,j-1,(\beta_1\alpha_2)] + gap\_penalty, \\ \qquad \beta_1 \in pred(\alpha_1) \end{cases} = max$$

Complexity: quadratic in the size of the input

Examples







Scoring system

Scores: Several variants

1. **simple scoring system**, where a match score and transition/transversion mutation penalties are used
2. the above system, with **corrections** applied to these scores in order **to discourage the anti-translation into very improbable codons**
   each node n has a weight w_n computed according to a codon usage table
   the score for aligning n_1 and n_2 is corrected as follows:

$$score(n_1, n_2) = s(n_1, n_2) + \log(w_{n_1} w_{n_2})$$

3. a **scoring matrix** estimated from artificial data consisting of proteins perfectly aligned with a frameshift, that suffered mutations according to a codon mutation rate matrix
   this score matrix can be readjusted as soon as enough real data (correct alignments) is gathered

Gaps

Non-monotonic gap penalty function

- favours the insertion/deletion of full codons
- the gaps that corespond to frameshifts are strongly penalised and their number is restricted

RESULTS

Validation: Perfect alignments detected between viral proteins (phage PhiX174, Influenza A virus), and *E.coli* plasmid proteins that are known to be encoded by overlapping genes.

Identification of frameshifts mutations in mouse and human proteins. Example:

```
CD2A2_MOUSE Cyclin-dependent kinase inhibitor 2A , CDN2B_HUMAN Cyclin-dependent kinase 4 inhibitor B
[L][L][R][L][E][R][I][L ][R][R][G][P][H][R][N][P][G][   P][G][D][D][D][G][Q][R][S][R][S][   S][S][S][A][Q][L][R][C][R][F][E][L][R][G][P][H][Y][L][L][P][P][G][A][R][R][S]
 TCTTGAGGCTGGAGCGGATCCT-ACGGCGTGGACCGCATCGGAATCCGGGCC---CAGGTGATGATGATGGGCAGCGCTCGCGTAGCT---CTTCTTCTGCTCAGCTACGGTGCCGATTCGAACTGCGCGGACCCCACTACCTTCTCCCGCCCGGTGCACGACGCAGC
 |||||||||||||||||||| ||||||||:||||||.||||||..|||||||   ||||||||||||||||||||||||||||||||||||   ||||||||||   |:||||||||||:||||||||||.||||||:||||||:||.||||||||||||||||||||
 TCTTGAGGCTGGAGCGGATCCTAACGGCGTGAACCGCTTCGGACGCCGGGCCATCCAGGTGATGATGATGGGCAGCGCTCGCGTAGCTGAACTTCTTCTGCT---CCACGGTGCCGAGCCGAACTGCGCGGACCCCGCTACCCTCACCCGCCCGGTGCACGACGCAGC
L][L][E][A][G][A][D][P][N][G][V][N][R][F][G][R][R][A][I][Q][V][M][M][M][G][S][A][R][V][A][E][L][L][L][L   ][H][G][A][E][P][N][C][A][D][P][A][T][L][T][R][P][V][H][D][A][A

[A][G][R][L][P][G][H][A][G][G][A][A][R][V][R][G][S][A][G][C][A][R][C][L][G][S][P][A][A][R][L][G][P][R][A][G][T][S][R][H][R][A][I][F][A][
GCGGGAAGGCTTCCTGGACACGCTGGTGGTGCTGCACGGGTCCGGGGCTCGGCTGGATGTGCGCGATGCCTGGGGTCGCCTGCCGCTCGACTTGGCCCAAGAGCGGGGACATCGCGACATCGCGCGATATTTGCGT
||||||||||||||||||||||||||||||||||||||||:||.||||||||||||||||||||||||||||||||||||||||||||||||||.||||||||||.||||||||||||||||||:||||||.||||||||||
GCGGGAAGGCTTCCTGGACACGCTGGTGGTGCTGCACAGGGCCGGGGCTCGGCTGGATGTGCGCGATGCCTGGGGTCGCCTGCCGGTCGACTTGGCCGAAGAGCGGGGACATCGCGACGTCGCGGGATATTTGCGT
][R][E][G][F][L][D][T][L][V][V][L][H][R][A][G][A][R][L][D][V][R][D][A][W][G][R][L][P][V][D][L][A][E][E][R][G][H][R][D][V][A][G][Y][L][R]
```

REMARKS

- The method offers the possibility of correcting existing alignments
- Gives clues about the ancestral sequence of two related proteins
- Can be efficient in identifying common origins even when they are no longer obvious by simple protein or DNA sequence analysis, because of the alteration by frameshifts and synonymous mutations