INDIANA UNIVERSITY

informatics

Protein Structure Comparison by Alignment of Distance Matrices

Liisa Holm and Chris Sander
J. Mol. Biol. (1993) 233, 123-138

Presented by Kenny Daily
IU School of Informatics
kmdaily [at] indiana [dot] edu

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Overview

- Overview of Holm & Sander, Science 1996
- Introduction
- Overview of method of structure comparison
- Methods of scoring
- Building/Refining data representation
- Assembly of alignment

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Holm & Sander 1996

- Comparison with sequence or structure?
    - Close evolutionary distance – use sequence
    - Farther away, use sequence profiles
    - Any farther, use sequence

Similarity of structures over long evolutionary time remains even if sequences highly diverge!

- General method of matching 3D shapes
    - Visually this is easy! (But what human can visually query 1000s of protein structures?)

    - We need five things:
        - Object Representation (Distance Matrix)
        - Objective function to optimize (Φ)
        - Comparison algorithm to do optimization (MCMC)
        - Results (alignment or superimposition)
        - Statistical significance of results (Harini)

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Introduction

- H&S's (1993) approach is an alignment problem
  - Find a subset of aligned (equivalent) residues in two proteins
  - Proteins are represented by Cα-Cα distance matrices – a 2D representation of a 3D structure
  - allows permutations of segments of aligned residues

- **"Similar 3D structures have similar inter-residue distance."**

  - We can find these by visualizing moving a transparent distance matrix on top of another one and look for similar submatrices

  - Patches on main diagonal are similar 2' structures
  - Patches short distance away from diagonal are similar 3' structures

  - Common structural motif easily seen by "collapsing" – removing residues that are not aligned (equivalent)

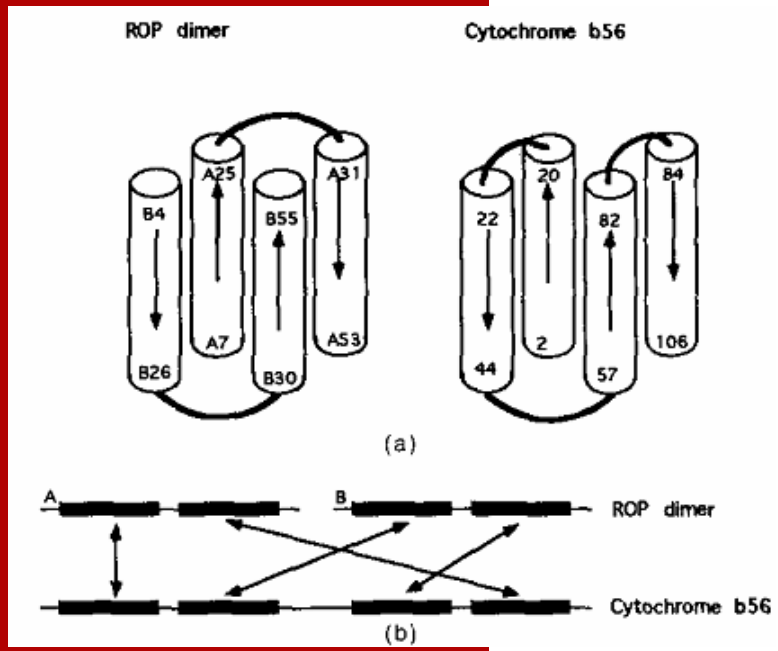
kmdaily@indiana.edu

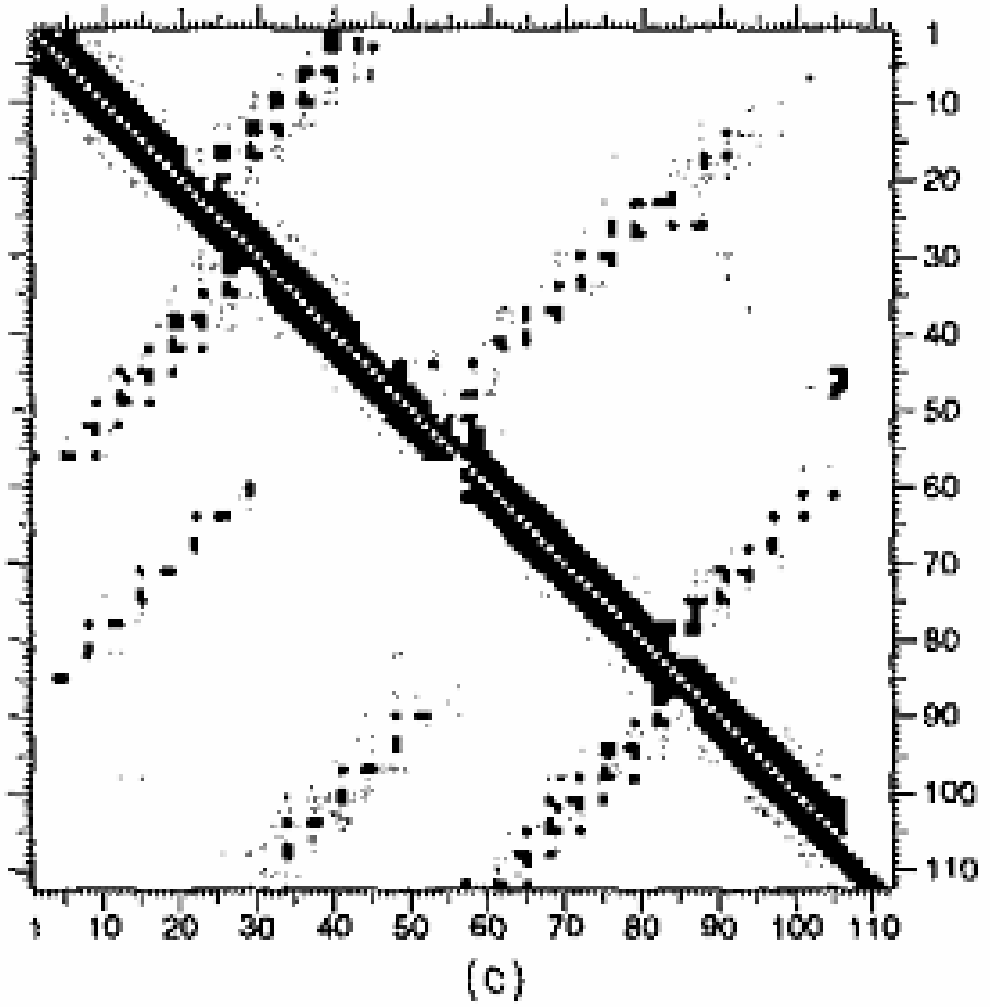
INDIANA UNIVERSITY
informatics

Introduction

Darker = better (shorter distance)

b56





ROP

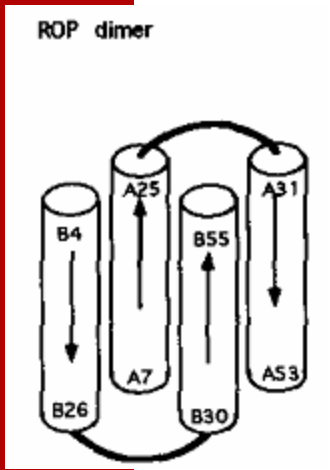
kmdaily@indiana.edu

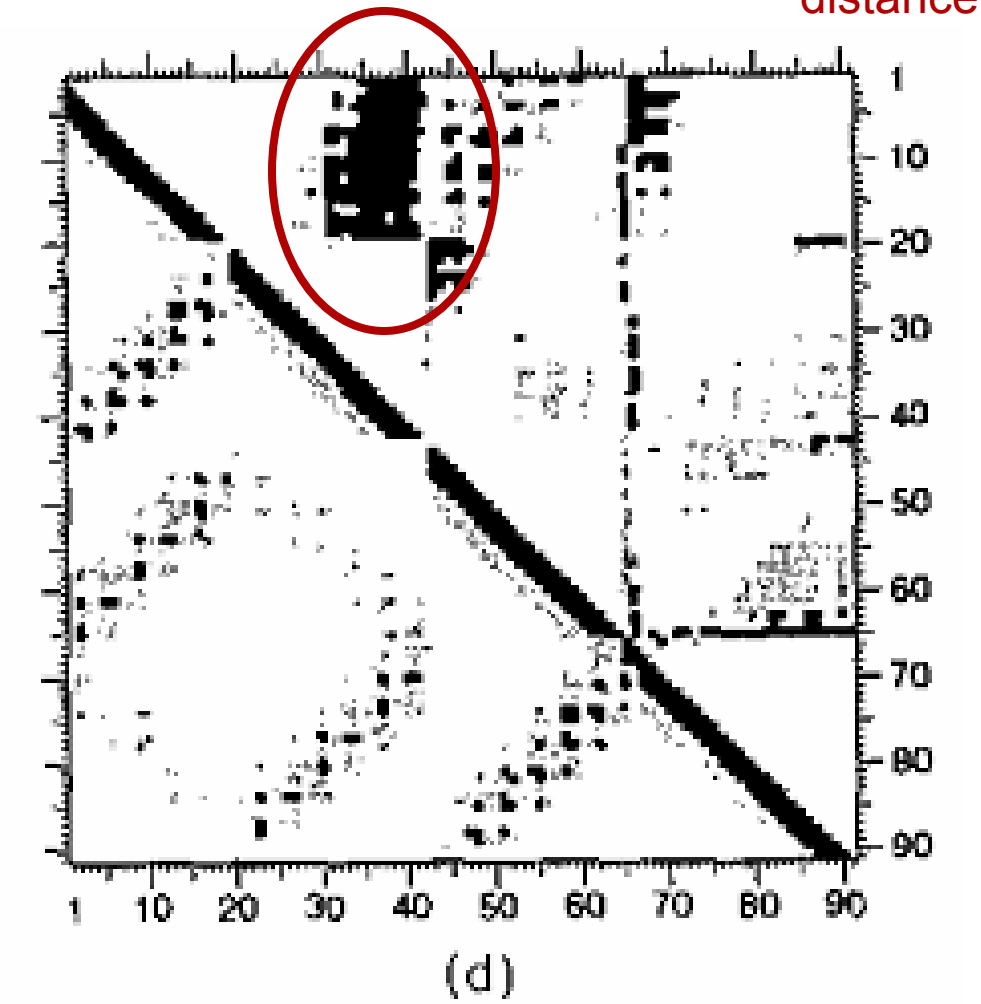
INDIANA UNIVERSITY
informatics

Introduction

Lighter = better (smaller deviation)

Difference distance matrix





Collapsed Distance Matrix of ROP

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Overview of Method

- Finding best subset of equivalent residues is non-trivial problem!
- An alignment is scored by sum of pairwise similarities

- Divide distance matrices into overlapping regions
  - hexapeptide-hexapeptide contact patterns - **within a protein**

- Search for pairs of similar contact patterns between two proteins
  - Contact pairs - implies a subalignment involving two fragments in each protein

- Starting from a few aligned residues, chain together contact pattern pairs that share previously equivalenced fragment
  - (a,b) – (b,c) – (c,d) : share b and c

- Find maximal chain – this is optimal alignment

kmdaily@indiana.edu

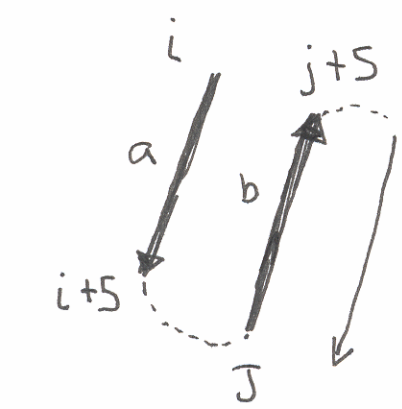
INDIANA UNIVERSITY

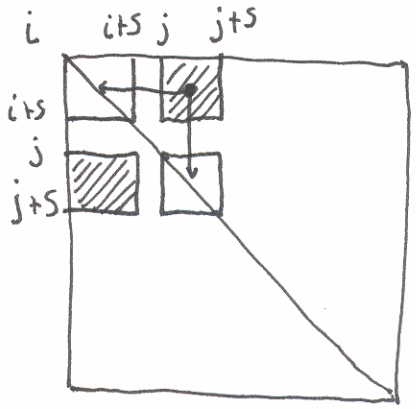
informatics

kmdaily@indiana.edu

a: (i,i+5), b:(j,j+5),
a´: (i´,i´+5), b´: (j´,j´+5)

What do arrows mean?



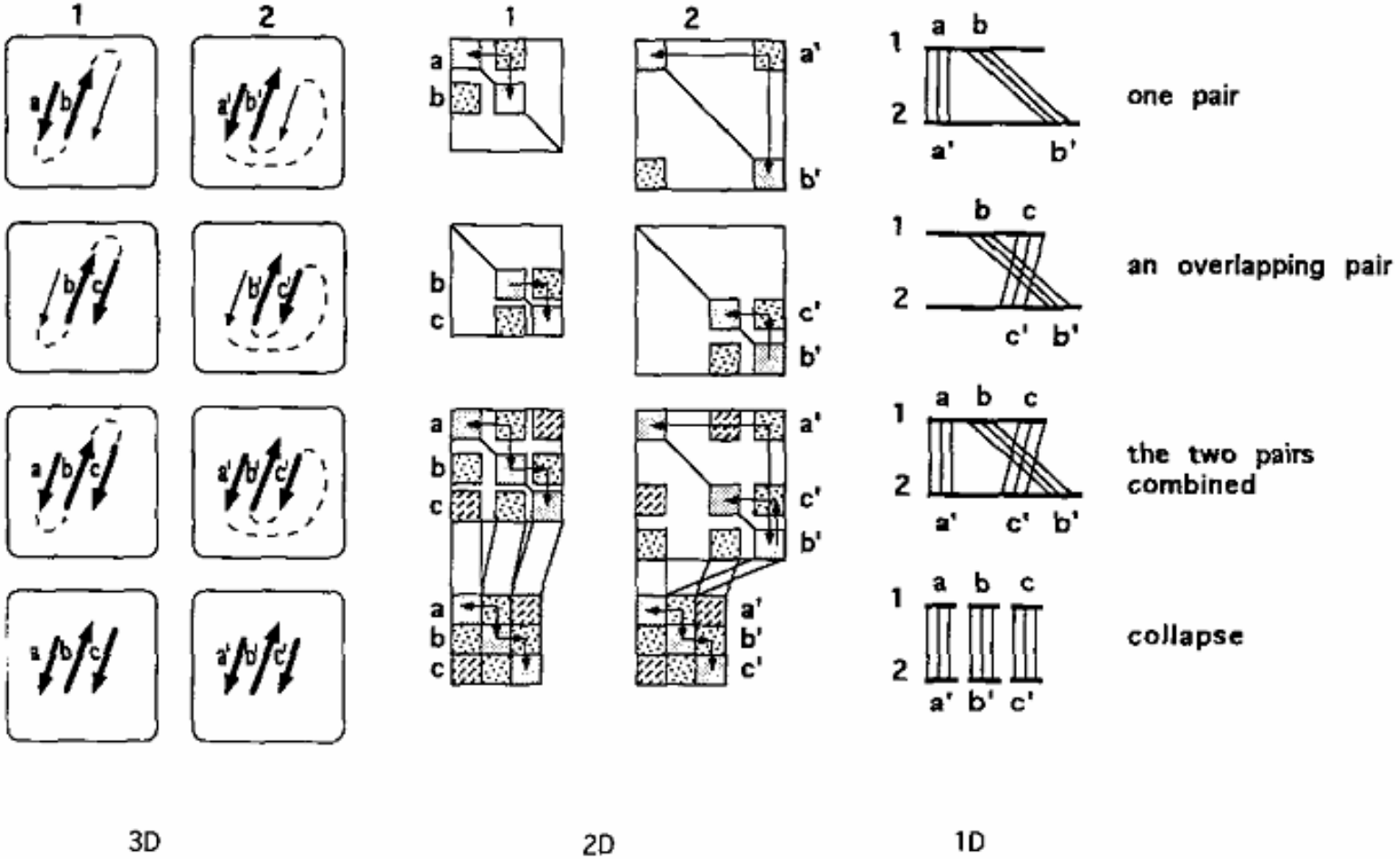


INDIANA UNIVERSITY

informatics

a: (i,i+5), b:(j,j+5),
a´: (i´,i´+5), b´: (j´,j´+5)

What do arrows mean?



kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

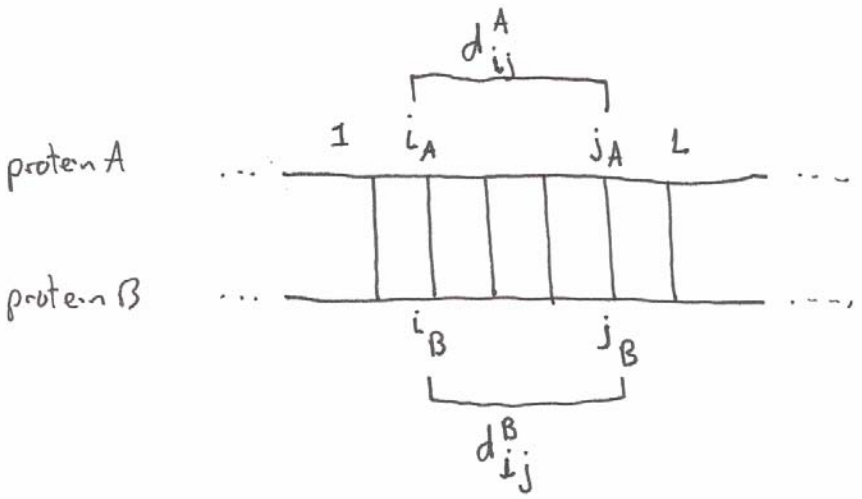
Methods

Given two proteins A and B:
Match of two substructures is the additive similarity S, defined as:

$$S = \sum_{i=1}^{L} \sum_{j=1}^{L} \phi(i, j)$$

i and *j* label pairs of aligned residues:

$$i = (i_A, i_B),\ j = (j_A, j_B)$$



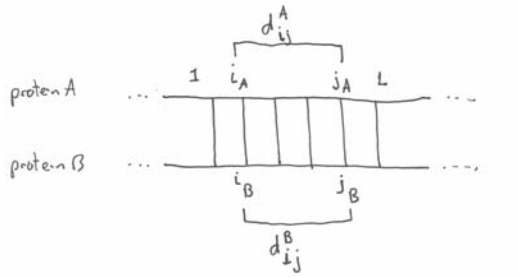
$\phi(i, j)$ Is a similarity measure based on Cα-Cα distance matrices

It is the sum of differences in distance of two aligned residue pairs

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics



kmdaily@indiana.edu

Similarity measures

H&S discuss two (only use one?)

Rigid similarity score (similar to rigid body 3D superimposition)

$$\phi^R(i,j) = \Theta^R - |\, d_{ij}^A - d_{ij}^B \,|$$

$$\Theta^R = \text{zero level similarity (highly/exactly similar)}$$

$$\text{if } d_{ij}^A = d_{ij}^B, \text{ then distance} = 0,$$

$$\phi^R = \Theta^R = 1.5A$$

As difference of the distance between two pairs of residues increases, similarity score goes below 0.

Good score is between 0 – 1.5 A, Bad score below 0

INDIANA UNIVERSITY
informatics

Similarity measures

Elastic Similarity Score – a relative score rather than absolute, like rigid score. The relative distance between two pairs of residues is analyzed.

$$\phi^E(i,j) = \begin{cases} \left(\Theta^E - \dfrac{|\, d_{ij}^A - d_{ij}^B \,|}{d_{ij}^*} \right) w(d_{ij}^*), i \neq j \\ \Theta^E, i = j \end{cases}$$

if $d_{ij}^A = d_{ij}^B$, distance $= 0$,

$\phi^E = \Theta^E = 20\%$

Θ^E = zero level similarity (highly/exactly similar)

$w(d_{ij}^*)$ weights down longer range distances (less discriminative)

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Building/Refining Data Representation

• What we have: two lists of contact patterns

• What we want: a “pair list” of contact patterns, with one contact pattern from each protein making a pair, such that the pairs are highly similar to each other

• First, reduce the number of contact patterns, and then the number of possible pairs:

$$\text{Size of distance matrices}: N_A^2, N_B^2$$

• If we consider any size contact pattern, then product of above sizes is number of comparisons (but alignment only has $N_A N_B$)

• However, we know the size of the contact patterns and can count how many there are (N – 5)

possible contact patterns per protein = $\binom{N-5}{2}$

Max number of pairs = $\binom{N_A-5}{2}\binom{N_B-5}{2}$

kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

Building/Refining Data Representation

- Reduce within distance matrices
  - exclude contact patterns that are:
    - overlapping, and
    - highly similar
  - represent this subset of patterns by the most compact one (lowest mean intra-pattern distance)

- Reduce in pair list
  - Sort both sets of contact patterns by mean intra-pattern distance (from short range to long range interactions)

  - remove redundant pairs by comparing between distance matrices

$$d^{A_{reduced}} \text{ to } d^{B_{full}}, \text{ and } d^{B_{reduced}} \text{ to } d^{A_{full}}$$

- This results in a new pair list, and remove the redundant ones
- Allow permutations of hexapeptides (reverse their orientation)
- Use speed-up methods to reduce the number of times the similarity score needs to be calculated – screen for extreme results

kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

Finally, a list of pairs (whew!

- The sorted, reduced contact patterns are added to the final list, and this list is closed when:

  1. The Mean intra-pattern distance reaches 25 A, or
  2. 80,000 contact pairs with a positive similarity score are added

- These are again sorted by similarity score, and 40,000 are used for the alignment.

kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

Data analysis

Table 1

Simplifying combinatorial complexity in the comparison of hen egg-white lysozyme (1lyz) with T4 lysozyme (2lzm)

A. Distance matrices	
1lyz	
No. of overlapping hexapeptides	124
Total no. of contact patterns	7626
No. of contact patterns in reduced distance matrix	5332
2lzm	
No. of overlapping hexapeptides	159
Total no. of contact patterns	12,561
No. of contact patterns in reduced distance matrix	4709
B. *Pair list*	
Total no. of pairs of contact patterns	96×10^6
Total no. of pairs of contact patterns after reduction	71×10^6
No. of checks by filters on row/column sums†	9×10^6
No. of residue-by-residue similarity score calculations	2×10^5
No. of kept pairs of contact patterns after ranking by score	4×10^4

Length of proteins
1lyz = 129 residues
2lzm = 164 residues

$$7626 = \binom{129}{2} \quad (\text{not } N_A^2)$$

$$12{,}561 = \binom{159}{2}$$

$$96 \times 10^6 = 7{,}626 \times 12{,}561$$

A 2-fold reduction!

kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

Assembly of alignment

If you are confused, just remember what we have now is a list of pairs of contact patterns that we know are similar to each other!

• H&S introduce a heuristic method to find the optimal alignment, given a similarity scoring function and a set of equivalenced (aligned) contact pairs

• A Monte Carlo Optimization (more specifically, this is a Markov Chain Monte Carlo Optimization, MCMC)

• Think of an inebriated person randomly flying through high dimensional space (in this case, the space of all alignments of two proteins), and we assign him/her a score based on the path he/she has flown through

• Generally, we want to take a path that is always increasing the score, but sometimes its better to take a penalty and possibly reach a higher score

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Assembly of alignment

• We want an iterative improvement by random walk exploration through the possible alignment space

• A move through this space can be one of two things, an addition or removal of some set of residues to/from the current alignment

• The addition or deletion changes the similarity score of the current alignment (since it is a sum of pairwise alignment scores)

•The move is randomly chosen and then accepted according to a given probability distribution:

$$\text{prob(accepting a move)} = e^{(\beta \times (S' - S))},$$

$$S = \text{old score}, S' = \text{new score}, \beta = \text{parameter}$$

$$\text{As } \beta \text{ increases, probability of accepting a move that decreases score increases}$$

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

Assembly of alignment

• The resulting alignments generated from a set of moves (additions or deletions) is a chain (a Markov Chain) that H&S term a **trajectory**

• The best alignment (state) through the trajectory is remembered

• Only moves with tetrapeptides are allowed (to avoid spurious matches of single residues)

• The addition of tetrapeptides may overlap (add 1-4 residues)

• For each hexapeptide, there are 3 tetrapeptides to look at

kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

MCMC Optimization Modes

- Expansion
  - look for contact patterns that overlap with current alignment
  - this gives us some subset of contact pairs that can possibly be added
  - a cycle of expansion tests all of these possibilities in random order
  - sometimes the addition of a tetrapeptide may cause some inconsistencies in the alignment – these are removed if the new alignment is accepted out of all the possible choices
  - Any additions/removals changes similarity score!
- Trimming
  - Remove any tetrapeptide fragments whose removal increases the overall similarity score

kmdaily@indiana.edu

INDIANA UNIVERSITY
informatics

MCMC Optimization Stages

• 1 expansion, trim, 5 expansions, trim, 5 expansions, trim, etc…

• Beta changes as well (initally 50 and after every 5), otherwise changed based on current score to give a constant acceptance ratio for alignments of different lengths (As L and S increase, beta decreases)

•Stage 1 – coarse grained
• Stage 2 – found an optimum
• Stage 3 – refine final alignment

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

MCMC Optimization Stage 1

- Screen the pair list for non-overlapping triplets (remove them!)
  - (a,b)-(a'b') , (a,c)-(a',c'), (b,c)-(b',c') = (a,b,c) – (a',b',c')
- Seeds are generated with singlets (a,a')
- Start with large number of seed alignments, ~100
- Then some process of merging singlets that overlap and have same relative sequence shift (WHAT?)
- Each seed undergoes one expansion/trim cycle which passes through (at most) the pair list once
- Compare pairs of trajectories - >50% sequence identity, the one with the lower score is removed
- Sort the alignments based on similarity score and **keep the top 10**

kmdaily@indiana.edu

INDIANA UNIVERSITY

informatics

kmdaily@indiana.edu

MCMC Optimization Stage 2 and 3

• **Stage 2**

•Have 10 alignments, now continue expansion/trim cycles until all the scores don't change through 20 cycles

• Again, trajectories are eliminated when:

- >80% sequence identity, lower scoring trajectory is removed
- Or if any score lags too far behind the current best (n/n+1)

• **Stage 3**

- Refine the best alignment by taking the best from Stage 2 and randomly removing 30% of the alignment to seed 10 more
- Continue like Stage 2
- Re-seeding his is done after every 20 cycles until no improvement in score

Yay, now we have an alignment that can now be represented as an alignment or as