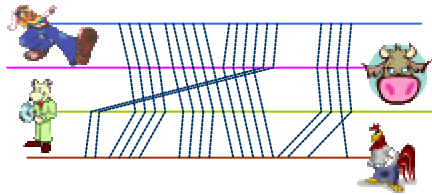


Multiple Sequence Alignments



Reading

Durbin's book:
Chapter 6.1-6.4

Gusfield's book:
Chapter 14.1, 14.2, 14.5, 14.6.1

Papers:
avid, lagan

Optional:
All of Gusfield chapter 14,
Papers: tcoffee, slagan, scl

Lecture 12, Tuesday May 13, 2003

Definition

Given N sequences x^1, x^2, \dots, x^N :

- Insert gaps (-) in each sequence x^i , such that
 - All sequences have the same length L
 - Score of the global map is maximum

The *sum-of-pairs* score of an alignment is the sum of the scores of all induced pairwise alignments

$$S(m) = \sum_{k < l} s(m^k, m^l)$$

$s(m^k, m^l)$: score of induced alignment (k, l)

Lecture 12, Tuesday May 13, 2003

Consensus

```

-AGGCTATCACCTGACCTCCAGGCCGA--TGCCC--
TAG-CTATCAC--GACCGC--GGTCGATTGCCCCGAC
CAG-CTATCAC--GACCGC----TCGATTGCTCGAC

CAG-CTATCAC--GACCGC--GGTCGATTGCCCCGAC
    
```

- Find optimal consensus string m^* to maximize

$$S(m) = \sum_i s(m^i, m)$$

$s(m^k, m)$: score of pairwise alignment (k, l)

Lecture 12, Tuesday May 13, 2003

Multiple Sequence Alignments

Algorithms

Multidimensional Dynamic Programming

- Example: in 3D (three sequences):



- 7 neighbors/cell

$$F(i, j, k) = \max \{ \begin{aligned} &F(i-1, j-1, k-1) + S(x_{ip}, x_{jp}, x_{kp}), \\ &F(i-1, j-1, k) + S(x_{ip}, x_{jp}, -), \\ &F(i-1, j, k-1) + S(x_{ip}, -, x_{kp}), \\ &F(i-1, j, k) + S(x_{ip}, -, -), \\ &F(i, j-1, k-1) + S(-, x_{jp}, x_{kp}), \\ &F(i, j-1, k) + S(-, x_{jp}, -), \\ &F(i, j, k-1) + S(-, -, x_{kp}) \end{aligned} \}$$

Lecture 12, Tuesday May 13, 2003

Progressive Alignment

- Multiple Alignment is NP-complete
- Most used heuristic: Progressive Alignment

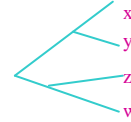
Algorithm:

1. Align two of the sequences x_i, x_j
2. Fix that alignment
3. Align a third sequence x_k to the alignment x_i, x_j
4. Repeat until all sequences are aligned

Running Time: $O(N^2)$

Lecture 12, Tuesday May 13, 2003

Progressive Alignment



- When evolutionary tree is known:
 - Align closest first, in the order of the tree

Example:

- Order of alignments:
1. (x,y)
 2. (z,w)
 3. (xy, zw)

Lecture 12, Tuesday May 13, 2003

Progressive Alignment: CLUSTALW

CLUSTALW: most popular multiple protein alignment

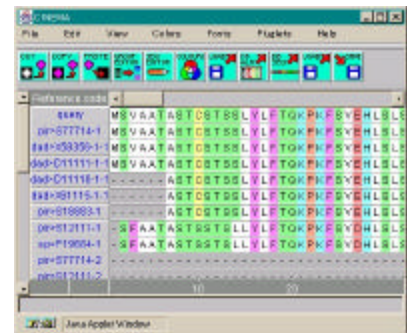
Algorithm:

1. Find all d_{ij} : alignment dist (x^i, x^j)
2. Construct a tree (Neighbor-joining hierarchical clustering)
3. Align nodes in order of decreasing similarity

+ a large number of heuristics

Lecture 12, Tuesday May 13, 2003

CLUSTALW & the CINEMA viewer



Lecture 12, Tuesday May 13, 2003

Iterative Refinement

One problem of progressive alignment:

- Initial alignments are "frozen" even when new evidence comes

Example:

x:	GAAGTT	}	Frozen!
y:	GAC-TT		
z:	GAAC TG	}	Now clear correct y = GA-CTT
w:	GTACTG		

Lecture 12, Tuesday May 13, 2003

Iterative Refinement

Algorithm (Barton-Stenberg):

1. Align most similar x^i, x^j
2. Align x^k most similar to ($x^i x^j$)
3. Repeat 2 until ($x^1 \dots x^N$) are aligned
4. For $j = 1$ to N , Remove x^j and realign to $x^1 \dots x^{j-1} x^{j+1} \dots x^N$
5. Repeat 4 until convergence

Note: Guaranteed to converge

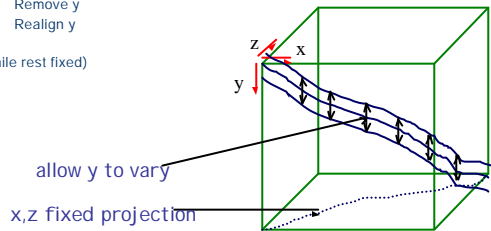
Lecture 12, Tuesday May 13, 2003

2. Iterative Refinement (cont'd)

For each sequence y

1. Remove y
2. Realign y

(while rest fixed)



Lecture 12, Tuesday May 13, 2003

Iterative Refinement

Example: align (x,y), (z,w):

```
x: GAAGTTA
y: GAC-TTA
z: GAACTGA
w: GTACTGA
```

After realigning y:

```
x: GAAGTTA
y: G-ACTTA
z: GAACTGA
w: GTACTGA
```

+ 3 matches

Lecture 12, Tuesday May 13, 2003

Iterative Refinement

Example not handled well:

```
x: GAAGTTA
y1: GAC-TTA
y2: GAC-TTA
y3: GAC-TTA
z: GAACTGA
w: GTACTGA
```

Realigning any single y_i changes nothing

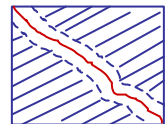
Lecture 12, Tuesday May 13, 2003

Restricted MDP

• Here is a final way to improve a multiple alignment:

1. Construct progressive multiple alignment m
2. Run MDP, restricted to radius R from m

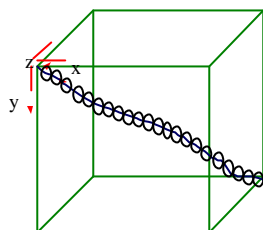
Running Time: $O(2^N R^{N-1} L)$



Lecture 12, Tuesday May 13, 2003

1. Restricted MDP

Run MDP,
restricted to
radius R from m



Running Time: $O(2^N R^{N-1} L)$

Lecture 12, Tuesday May 13, 2003

Restricted MDP (2)

```
x: GAAGTTA
y1: GAC-TTA
y2: GAC-TTA
y3: GAC-TTA
z: GAACTGA
w: GTACTGA
```

• Within radius 1 of the optimal

⇒ Restricted MDP will fix it.

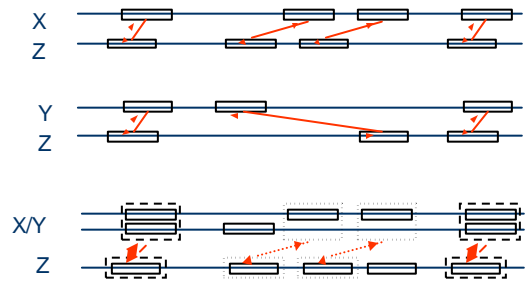
Lecture 12, Tuesday May 13, 2003

MLAGAN: Multiple Alignment

1. Multi-Anchoring
2. Progressive Alignment
3. Iterative Refinement

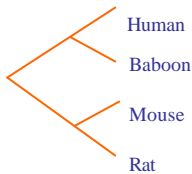
1. Multi-anchoring

To anchor the (X/Y), and (Z) alignments:



Lecture 12, Tuesday May 13, 2003

2. Progressive Alignment



Given N sequences, phylogenetic tree

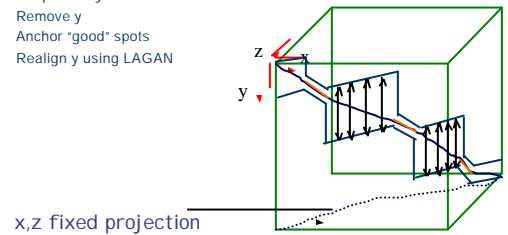
Align pairwise, in order of the tree (LAGAN)

Lecture 12, Tuesday May 13, 2003

3. Iterative Refinement

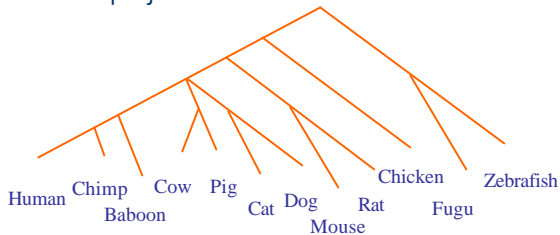
For each sequence y

1. Remove y
2. Anchor "good" spots
3. Realign y using LAGAN



Lecture 12, Tuesday May 13, 2003

Cystic Fibrosis (CFTR), 12 species The "zoo" project



- Human sequence length: 1.8 Mb
- Total genomic sequence: 13 Mb

Lecture 12, Tuesday May 13, 2003

Performance in the CFTR region

		Exons Perfect	Exons > 90%	TIME (sec)	MAX MEMORY (Mb)
MULTIMER	Mammals	25%	40%	45	40
	Chicken & Fishes	0%	0%	7	40
AVID	Mammals	95%	95%	1563	600
	Chicken & Fishes	23%	27%	212	387
LAGAN	Mammals	95%	99.7%	550	90
	Chicken & Fishes	80%	84%	862	90
MLAGAN	Mammals	95%	99.8%	4547	670
	Chicken & Fishes	82%	91%		

Lecture 12, Tuesday May 13, 2003

Alignment & Rearrangements

Evolution at the DNA level

SEQUENCE EDITS

Deletion Mutation

...ACGGTCCAGTACCA...

...AC---CAGTCCACCA...

REARRANGEMENTS

Inversion

Translocation

Duplication

Lecture 12, Tuesday May 13, 2003

Local & Global Alignment

Local

Global

Lecture 12, Tuesday May 13, 2003


Global Alignment Problem

Find least cost transformation of one sequence into another using new operations

- Sequence edits
- Inversions
- Translocations
- Duplications
- Combinations of above

Lecture 12, Tuesday May 13, 2003

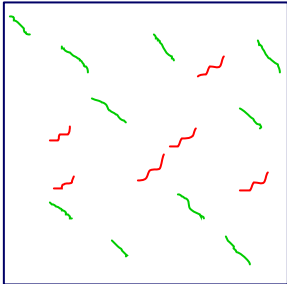
Shuffle-LAGAN



A global aligner for long DNA sequences

Lecture 12, Tuesday May 13, 2003

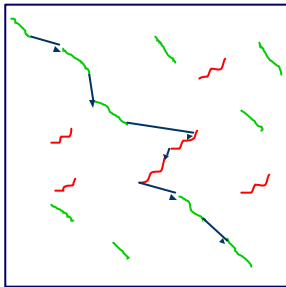
S-LAGAN: Find Local Alignments



1. Find Local Alignments
2. Build Rough Homology Map
3. Globally Align Consistent Parts

Lecture 12, Tuesday May 13, 2003

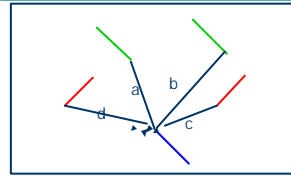
S-LAGAN: Build Homology Map



1. Find Local Alignments
2. Build Rough Homology Map
3. Globally Align Consistent Parts

Lecture 12, Tuesday May 13, 2003

Building the Homology Map



Chain (using Eppstein Gallil); each alignment gets a score which is MAX over 4 possible chains.

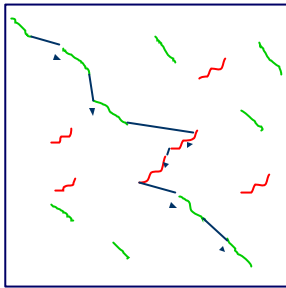
Penalties are affine (event and distance components)

Penalties:

- | | |
|------------------|---------------------------|
| a) regular | c) inversion |
| b) translocation | d) inverted translocation |

Lecture 12, Tuesday May 13, 2003

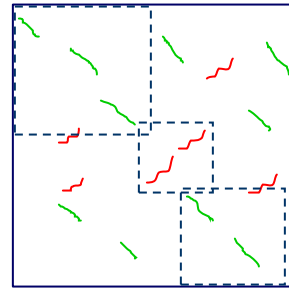
S-LAGAN: Build Homology Map



1. Find Local Alignments
2. Build Rough Homology Map
3. Globally Align Consistent Parts

Lecture 12, Tuesday May 13, 2003

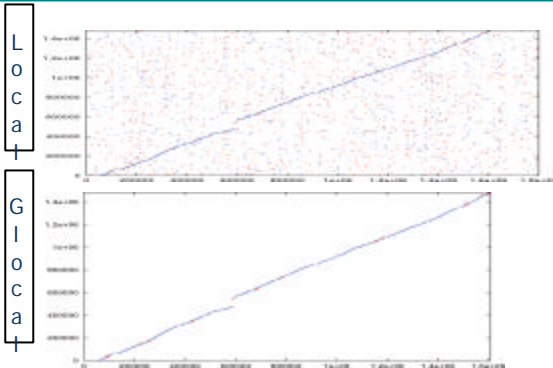
S-LAGAN: Global Alignment



1. Find Local Alignments
2. Build Rough Homology Map
3. Globally Align Consistent Fragments

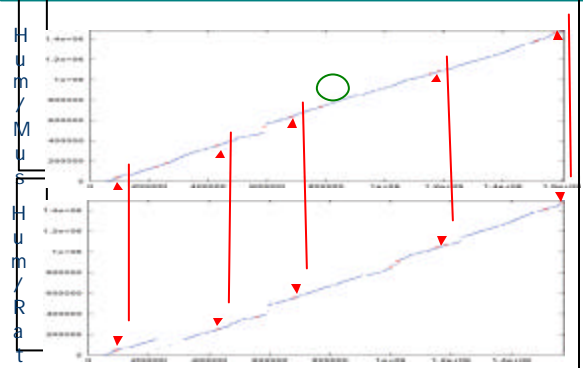
Lecture 12, Tuesday May 13, 2003

S-LAGAN alignments



Lecture 12, Tuesday May 13, 2003

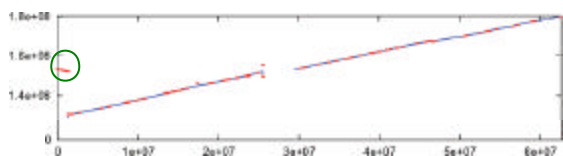
S-LAGAN alignments



Lecture 12, Tuesday May 13, 2003

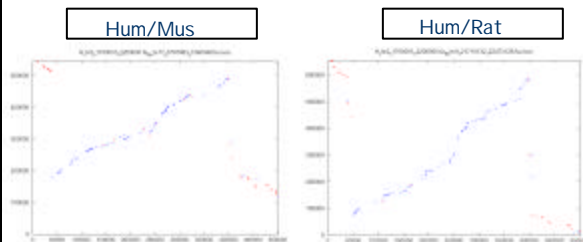
S-LAGAN alignments (Chr 20)

- Human Chr 20 v. homologous Mouse Chr 2.
- 270 Segments of conserved synteny
- 70 Inversions



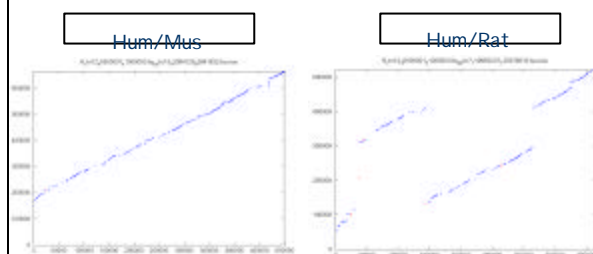
Lecture 12, Tuesday May 13, 2003

Some more examples



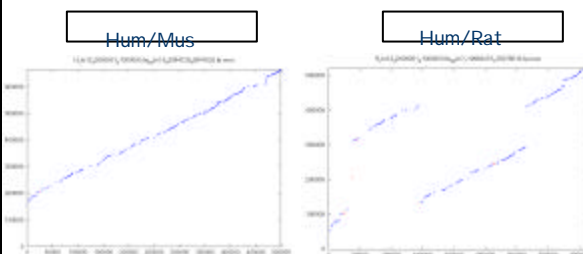
Lecture 12, Tuesday May 13, 2003

Some more examples



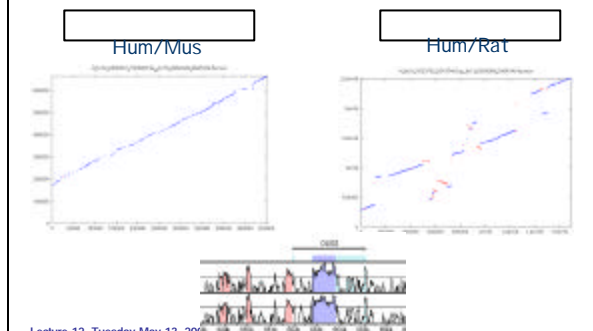
Lecture 12, Tuesday May 13, 2003

Some more examples



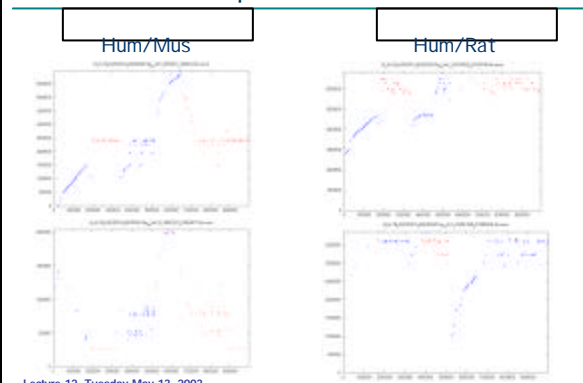
Lecture 12, Tuesday May 13, 2003

Some more examples



Lecture 12, Tuesday May 13, 2003

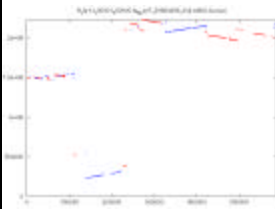
Some more examples



Lecture 12, Tuesday May 13, 2003

Some more examples

Hum/Mus



Hum/Rat

