Whole Genome Alignments

Dynamic programming
Local aligners
Global aligners
Glocal aligners

EMBL-EBI

Javier Herrero

Dynamic Programming

Needleman-Wunsch (1970) for global alignments

Smith-Watermann (1981) for local alignments

Need to define:

- Matches/mismatches scores
- Gap penalties

	A	**G**	**C**	**T**
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Match = +2

Mismatch = -1

Gap = -1

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0										
G											
A											
G											
A											
C											
G											
T											

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5										
A	-10										
G	-15										
A	-20										
C	-25										
G	-30										
T	-35										

Add gap penalties in the 1st row and 1st column

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5	x									
A	-10										
G	-15										
A	-20										
C	-25										
G	-30										
T	-35										

$x_1 = 0 + 7$ (Match)

$X_2 = -5 + -5$ (Insertion)

$X_3 = -5 + -5$ (Deletion)

$x = \max(x_1 ; x_2 ; x_3)$

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5	7									
A	-10										
G	-15										
A	-20										
C	-25										
G	-30										
T	-35										

$x_1 = 0 + 7$ (Match)

$X_2 = -5 + -5$ (Insertion)

$X_3 = -5 + -5$ (Deletion)

$x = \max(x_1 ; x_2 ; x_3)$

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5	7	x								
A	-10										
G	-15										
A	-20										
C	-25										
G	-30										
T	-35										

$x_1 = -5 + -1$ (Match)

$x_2 = 7 + -5$ (Insertion)

$x_3 = -10 + -5$ (Deletion)

$x = \max(x_1 ; x_2 ; x_3)$

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5	7	2								
A	-10										
G	-15										
A	-20										
C	-25										
G	-30										
T	-35										

$x_1 = -5 + -1$ (Match)
$x_2 = 7 + -5$ (Insertion)
$x_3 = -10 + -5$ (Deletion)

$x = \max(x_1 ; x_2 ; x_3)$

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5	7	2	-3	-8	-13	-18	-23	-28	-33	-38
A	-10										
G	-15										
A	-20										
C	-25										
G	-30										
T	-35										

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch (1970)

	-	G	A	C	T	A	G	T	T	A	C
-	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50
G	-5	7	2	-3	-8	-13	-18	-23	-28	-33	-38
A	-10	2	17	12	7	2	-3	-8	-13	-18	-23
G	-15	-3	12	12	9	6	9	4	-1	-6	-11
A	-20	-8	7	9	8	19	14	9	4	9	4
C	-25	-13	2	16	11	14	14	14	9	4	18
G	-30	-18	-3	11	13	10	21	16	11	8	13
T	-35	-23	-8	6	19	14	16	29	24	19	14

```
- G A C T A - G T T A C
- G A G - A C G T - - -
```

Start from the bottom-right corner and trace back the alignment

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Smith-Waterman (1981)

	-	G	A	C	T	A	G	T	T	A	C
-	0	0	0	0	0	0	0	0	0	0	0
G	0										
A	0										
G	0										
A	0										
C	0										
G	0										
T	0										

T. Smith: "My contribution was to add exactly 0"

$x_1 = 0 + 7$ (Match)
$x_2 = 0 + -5$ (Insertion)
$x_3 = 0 + -5$ (Deletion)

$x = \max(x_1 ; x_2 ; x_3 ; 0)$

Start with all 0's in the 1st column and in the 1st row

				T
				-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Smith-Waterman (1981)

	-	G	A	C	T	A	G	T	T	A	C
-	0	0	0	0	0	0	0	0	0	0	0
G	0	7	2	0	0	0	7	2	0	0	0
A	0	2	17	12	7	10	5	3	0	10	5
G	0	7	12	12	9	6	17	12	7	5	5
A	0	2	17	12	8	19	14	13	8	17	12
C	0	0	12	26	21	16	14	14	13	12	26
G	0	7	7	21	23	20	23	18	13	12	21
T	0	2	3	16	29	24	19	31	26	21	16

```
    G A C T A G T T A C
G A G A C - - G T
```

Find the maximum score and trace back the alignment

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

Gap = -5

Needleman-Wunsch vs Smith-Waterman

Needleman-Wunsch result:

```
- G A C T A - G T T A C
- G A G - A C G T - - -
```

Smith-Waterman result:

```
      G A C T A G T T A C
G A   G A C - - G T
```

Scoring scheme:

- Needleman-Wunsch can use pretty much anything
- Smith-Waterman requires negative scores for mismatches and/or gaps

A few notes

Needleman-Wunsch and Smith-Waterman guarantee the best score (given a matrix + gap penalty)

- There might be more than one alignment with that score:

```
ACTTGGAT      ACTTGGAT
ACTTC-AT      ACTT-CAT
```

Other gap scoring models are possible: affine gap

- gap open: -5 and gap extension: -1

The algorithms are not very efficient: *O(mn)*

- Several heuristics are required to align whole genomes

Building multiple alignments only makes the problem more complex

Whole Genome Alignments

Local aligners
Global aligners
Glocal aligners

EMBL-EBI

Whole Genome Alignment strategies

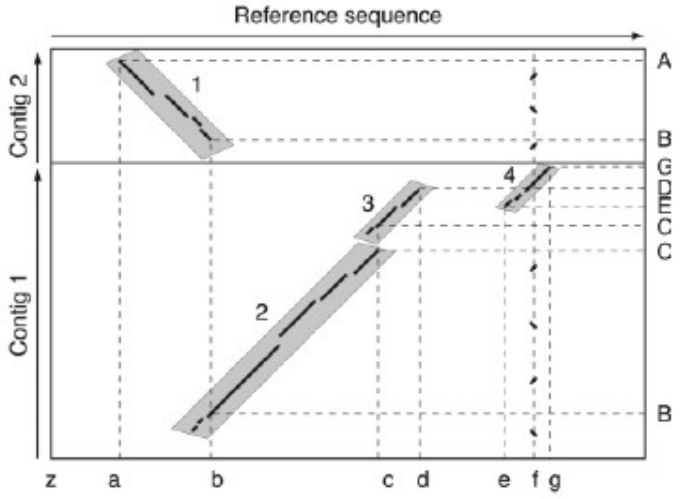
- Local aligners
  - Work by “stacking” pairwise alignments
  - High specificity
  - MultiPipMaker, TBA, MultiZ
- Global aligners
  - Need to pre-define collinear segments
  - Better sensitivity
  - AVID/MAVID, LAGAN/MLAGAN, Pecan
- 'Glocal' aligners
  - Combine both approaches
  - Shuffle-LAGAN, MAUVE

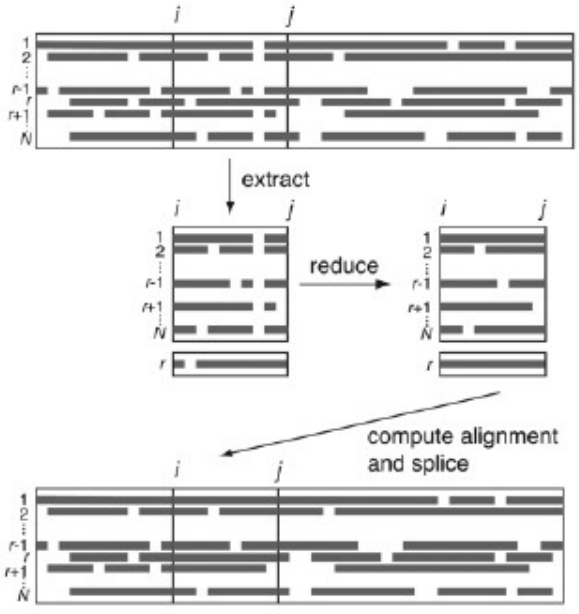
Local Alignments

EMBL-EBI

MultiPipMaker

- Uses a reference genome (non-draft)
- Builds blastz pairwise alignments
- prunes overlaps
- Refine the multiple alignment





Scott Schwartz et al., **MultiPipMaker and supporting tools: alignments and analysis of multiple genomic DNA sequences**, *Nucl. Acids Res*. 31, no. 13 (July 1, 2003): 3518-3524.

Threaded Block Aligner (TBA) & MultiZ

- Does not use a reference genome (at the beginning)
- Builds pairwise blastz alignments
- Removes overlaps to get a clean set of “blocks”
- MultiZ combines pairs of pairwise/multiple alignments:

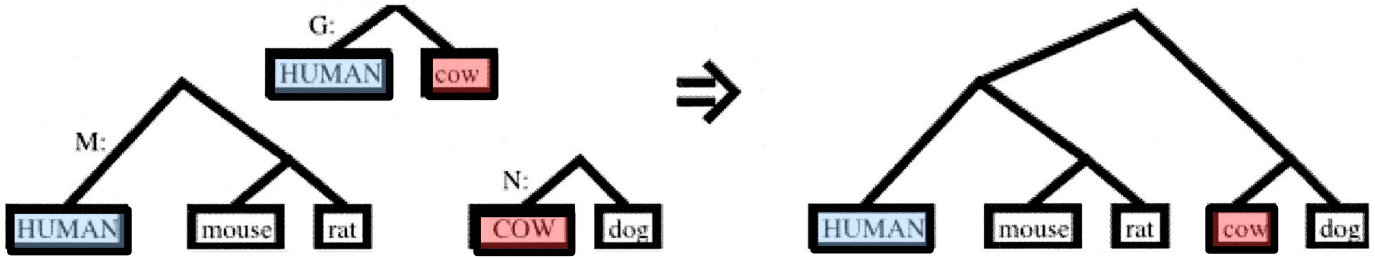


Figure 5 Pictorial representation of an application of MULTIZ. M is a human-ref blockset of human, mouse, and rat, whereas N is a cow-ref blockset of cow and dog. MULTIZ uses a pairwise human-ref blockset, G, of human and cow to guide the aligning process. The output is a human-ref blockset of human, mouse, rat, cow, and dog. The reference sequence for each blockset is indicated by capital letters.

- Dynamic programming is used to refine the multiple alignment (TBA only)
- The final alignment is usually “projected” on a reference species (as available on the UCSC Genome Browser)

Mathieu Blanchette et al., **Aligning multiple genomic sequences with the threaded blockset aligner**, *Genome Research* 14, no. 4 (April 2004): 708-715.

Global Alignments

EMBL-EBI

Global alignments

- Assumes that the homologous regions in the sequences appear in the same order and orientation
- The whole sequences are aligned
- To align whole genomes, they need an homology map

AVID

- Pairwise global aligner
- Uses suffix trees to find maximal matches
- Uses the maximal matches to build a set of anchors
- Anchors are selected such as they are non-overlapping and non-crossing
- Uses Needleman-Wunsch algorithm to align inter-anchor regions

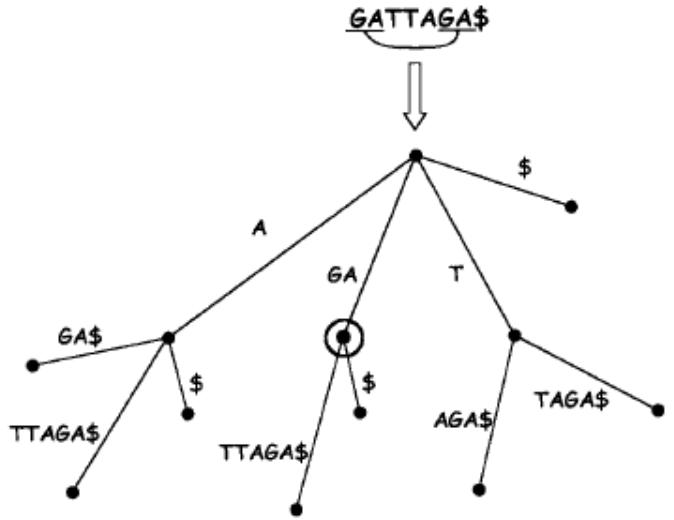


Figure 2 Finding maximal matches using a suffix tree: The suffixes of the word at the root are represented by the characters along the paths from the root to the leaves. Branchings in the tree correspond to locations where different suffixes shared the same prefix, and therefore are matches. Every internal node in the tree is therefore a match (with the matching sequence corresponding to the path characters along the path from the root). Maximal matches can be efficiently detected by considering some additional criteria.

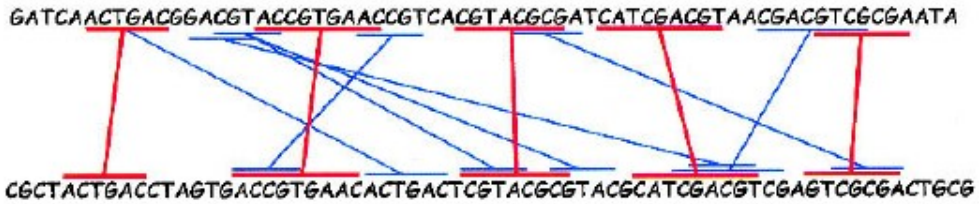
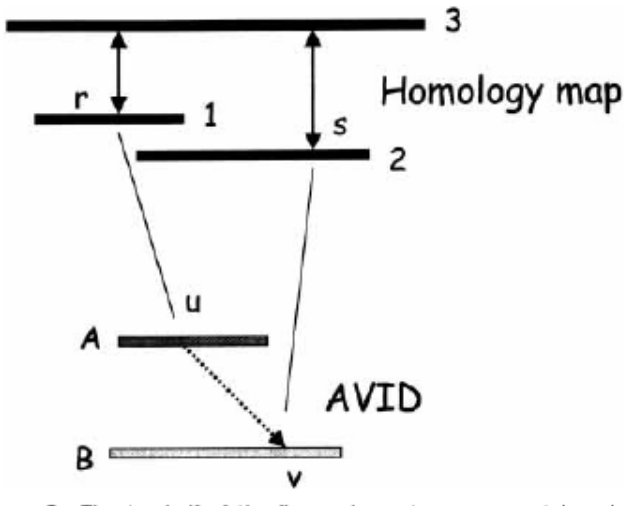


Figure 3 Selecting anchors from the set of matches. Every maximal match is shown in blue. A set of good anchors is shown in red.

Bray et al., **AVID: A Global Alignment Program**, *Genome Research* 13, no. 1 (January 1, 2003): 97-102.

MAVID (based on AVID)

- Aligns the first two sequences
- Infers the ancestral sequence
- Aligns the ancestral sequence to the third sequence
- Global constraints are inferred from the homology map



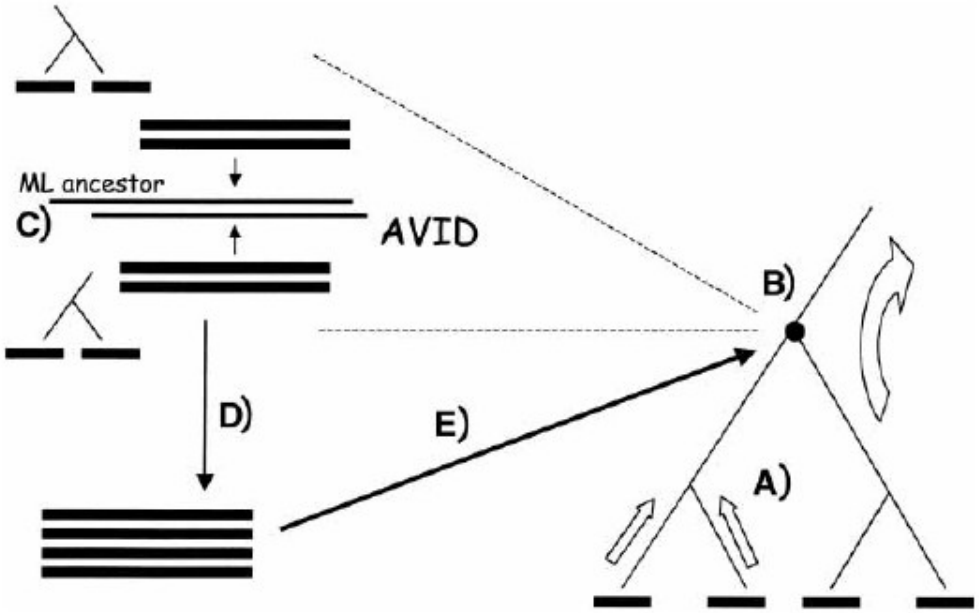


Figure 1 MAVID architecture overview. (*A*) Sequences are aligned upward along a guide tree and (*B*) alignments of alignments are performed at internal nodes. To align two alignments (*C*), maximum likelihood ancestor sequences are inferred from each of the separate alignments, and (*D*) the ancestor sequences are aligned with MAVID. The resulting multiple alignment (*E*) (corresponding to a subset of leaves of the tree) is then recorded at the internal node.

Bray & Pachter, **MAVID: Constrained Ancestral Alignment of Multiple Sequences**, *Genome Research* 14, no. 4 (April 2004): 693-699

LAGAN

- Global pairwise aligner
- Uses CHAOS to find
  - all high-scoring local alignments (B)
  - the rough global map (C)
- LAGAN uses CHAOS recursively
- Uses flexible anchors
- Banded Needleman-Wunsch alignment around and between anchors (D)

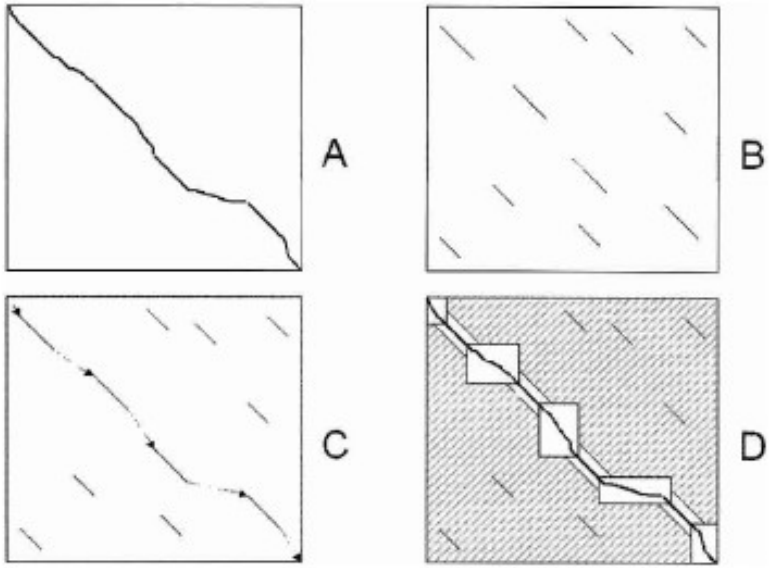


Figure 1 The LAGAN algorithm. (*A*) A global alignment between two sequences is a path between the top-left and the bottom-right corner of their alignment matrix. (*B*) LAGAN first finds all local alignments between the two sequences. (*C*) LAGAN computes a maximal-scoring ordered subset of the alignments, the *anchors*, and puts together a rough global map. (*D*) LAGAN limits the search for an optimal alignment to the area included in the boxes and around the anchors, and computes the optimal Needleman-Wunsch alignment limited to that area. LAGAN uses memory proportional to the area of the largest box plus the memory to hold the optimal alignment.

Michael Brudno et al., **LAGAN and Multi-LAGAN: Efficient Tools for Large-Scale Multiple Alignment of Genomic DNA**, *Genome Research* 13, no. 4 (April 1, 2003): 721-731.

MLAGAN or Multi-LAGAN

- Progressive global multiple aligner
- Aligns “multi-sequences” recursively
- Combines X-Z and Y-Z anchors to find the X/Y-Z anchors
- Optional iterative refinement step to overcome problems in the initial pairwise alignments

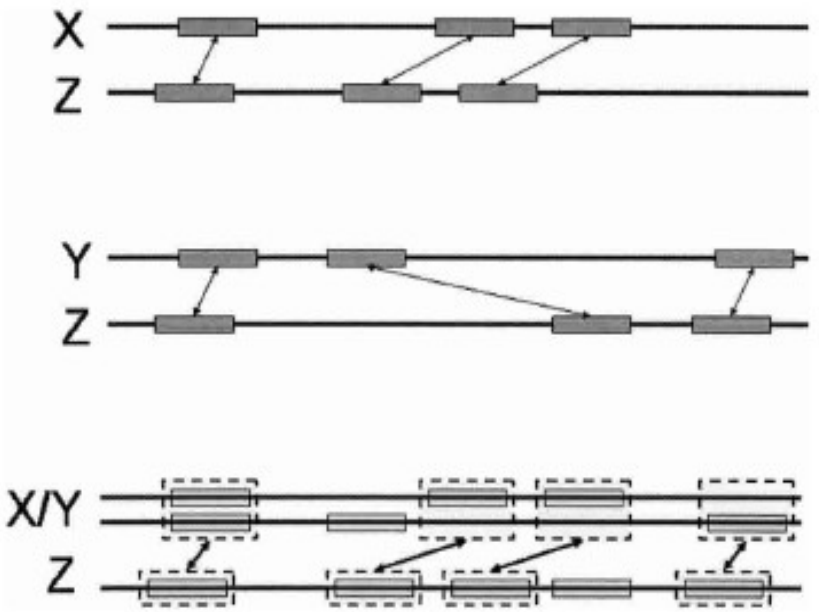


Figure 5 Generation of anchors during progressive alignment. Multi-sequence X/Y is aligned to sequence Z. Anchors between X and Z (*top*) and anchors between Y and Z (*middle*) are remapped to coordinates in the X/Y multi-sequence, and given a new score. Then, the Longest Increasing Subsequence algorithm is applied to select a subset of the remapped anchors, as the anchors between X/Y and Z.

Michael Brudno et al., **LAGAN and Multi-LAGAN: Efficient Tools for Large-Scale Multiple Alignment of Genomic DNA**, *Genome Research* 13, no. 4 (April 1, 2003): 721-731.

Pecan

- Global consistency-based multiple aligner
- Performs a banded alignment
- Uses the forward-backward algorithm for pairwise alignments
- Breaks the pairwise alignments into small fragments (sequence progressive aligner)
- Uses all pairwise alignments to build a consistent multiple alignment
- Uses “transitive” anchors to speed up the anchoring step

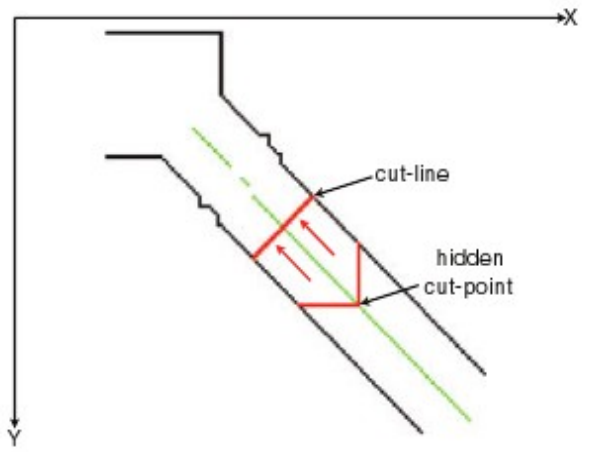
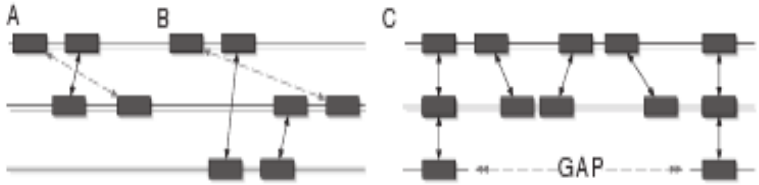
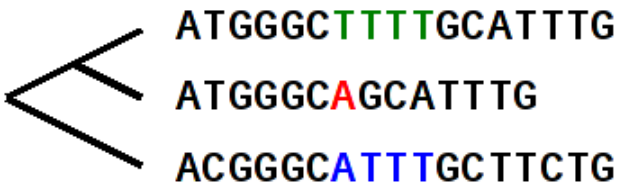


Fig. 1. Banded alignment. A schematic edit graph for two sequences (X and Y) is shown. Anchors, representing ungapped alignments are shown as green diagonal lines. Around these anchors a relaxed alignment band is computed (shown as thin black lines). Within the alignment band the structure of a cut-point is shown.



Paten et al., **Sequence progressive alignment, a framework for practical large-scale probabilistic consistency alignment**, *Bioinformatics* 25, no. 3 (February 1, 2009): 295-301.

Progressive vs consistency approach



One sequence after the other

Initial errors are difficult/expensive to avoid

Progressive aligner

ATGGGCTTTTGCATTTG
ATGGGC---AGCATTTG
vs
ATGGGCTTTTGCATTTG
ATGGGCA---GCATTTG

ATGGGCTTTTGCATTTG
ATGGGC---AGCATTTG
ACGGGCATTTGCTTCTG
or
ATGGGCTTTTGCATTTG
ATGGGCA---GCATTTG
ACGGGCATTTGCTTCTG

Consistency-based aligner

ATGGGCTTTTGCATTTG
ATGGGCATTTGCATTTG
and
ATGGGCATTTGCATTTG
ATGGGCA---GCATTTG

ATGGGCTTTTGCATTTG
ATGGGCA---GCATTTG

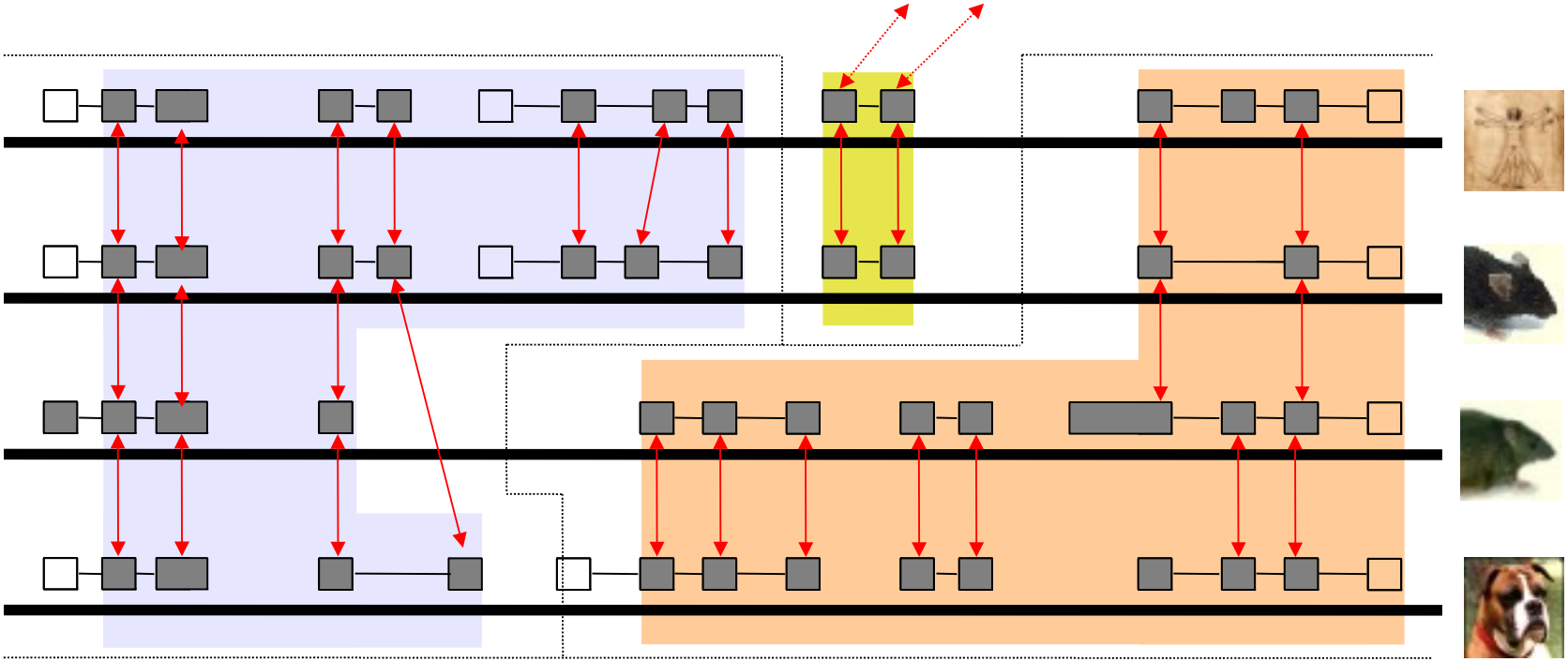
ATGGGCTTTTGCATTTG
ATGGGCA---GCATTTG
ACGGGCATTTGCTTCTG

Starts with all pw-alignments

Computationally v. expensive

Mercator (orthology map)

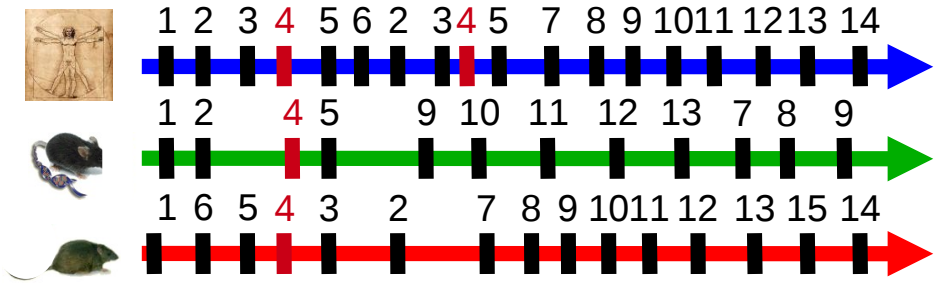
- Use all coding exons
- Get sets of best reciprocal hits
- Create orthology maps

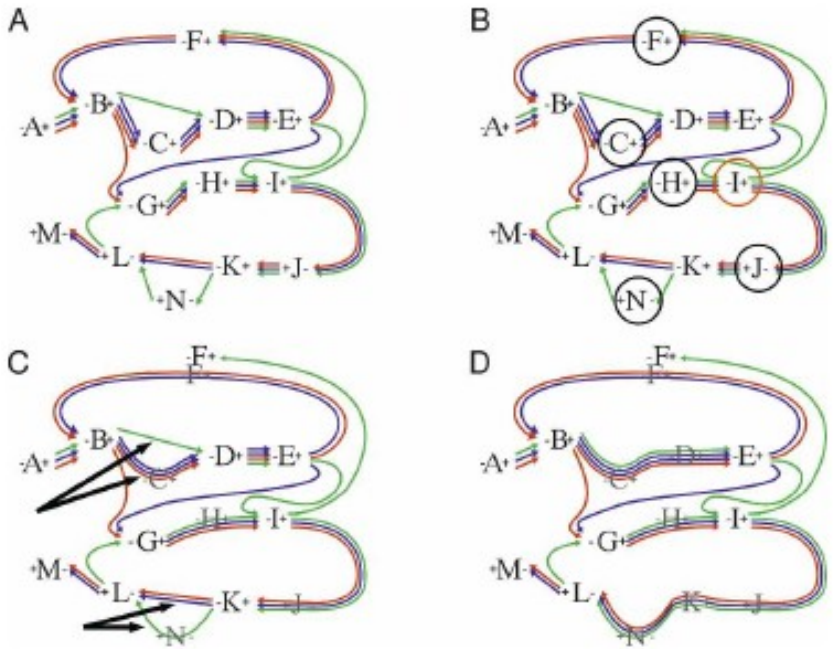


Dewey, **Aligning multiple whole genomes with Mercator and MAVID**, *Methods in Molecular Biology (Clifton, N.J.)* 395 (2007): 221-236.

Enredo (collinear segments)

- Maps a set of anchors on the genomes
- Anchors are obtained from regions of high homology
- Remove spurious hits
- Build a graph
- Simplify the graph: joining and annealing
- Other graph edits
- Supports duplications





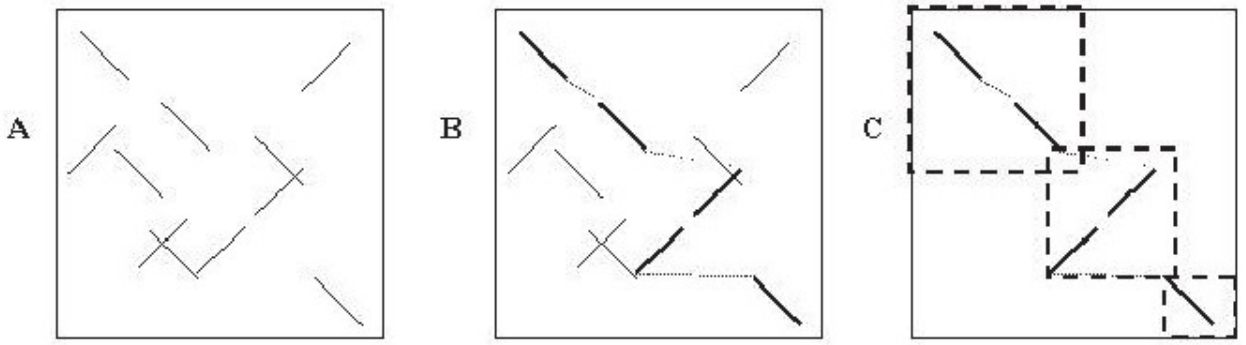
Paten et al., **Enredo and Pecan: genome-wide mammalian consistency-based multiple alignment with paralogs**, *Genome Research* 18, no. 11 (November 2008): 1814-1828.

Glocal Alignments

EMBL-EBI

SLAGAN or Shuffle-LAGAN

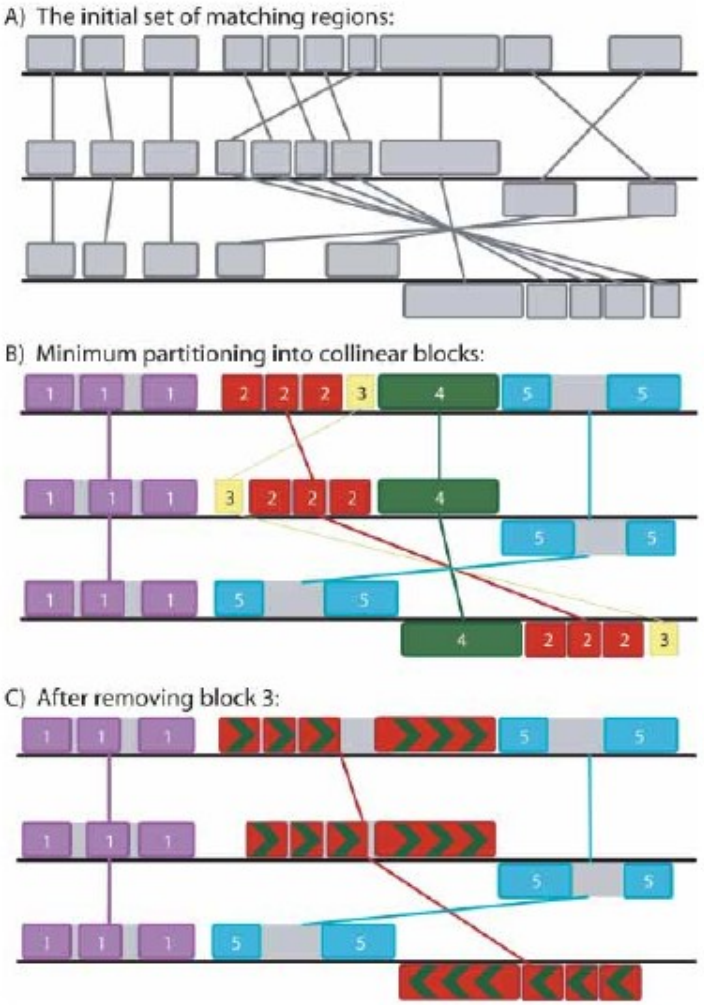
- Pairwise “glocal” aligner
- Performs a global alignment while allowing for some rearrangements
- Uses CHAOS to find a 1-monotonic map (monotonic on one sequence only)
- Sub-segments are aligned with LAGAN



Michael Brudno et al., **Glocal alignment: finding rearrangements during alignment**, *Bioinformatics* 19, no. suppl_1 (July 3, 2003): i54-62.

MAUVE

- Multiple “glocal” aligner
- Only addresses translocations and inversions
- Uses multi-MUMs as seeds
- Calculates a guide tree
- Removes spurious seeds
- Runs a ClustalW progressive alignment on the resulting blocks
- Performs well for closely related genomes
- New version addresses some of the shortcomings.



Darling et al., **Mauve: Multiple Alignment of Conserved Genomic Sequence With Rearrangements**, *Genome Research* 14, no. 7 (July 2004): 1394-1403.

Alignment Assessment

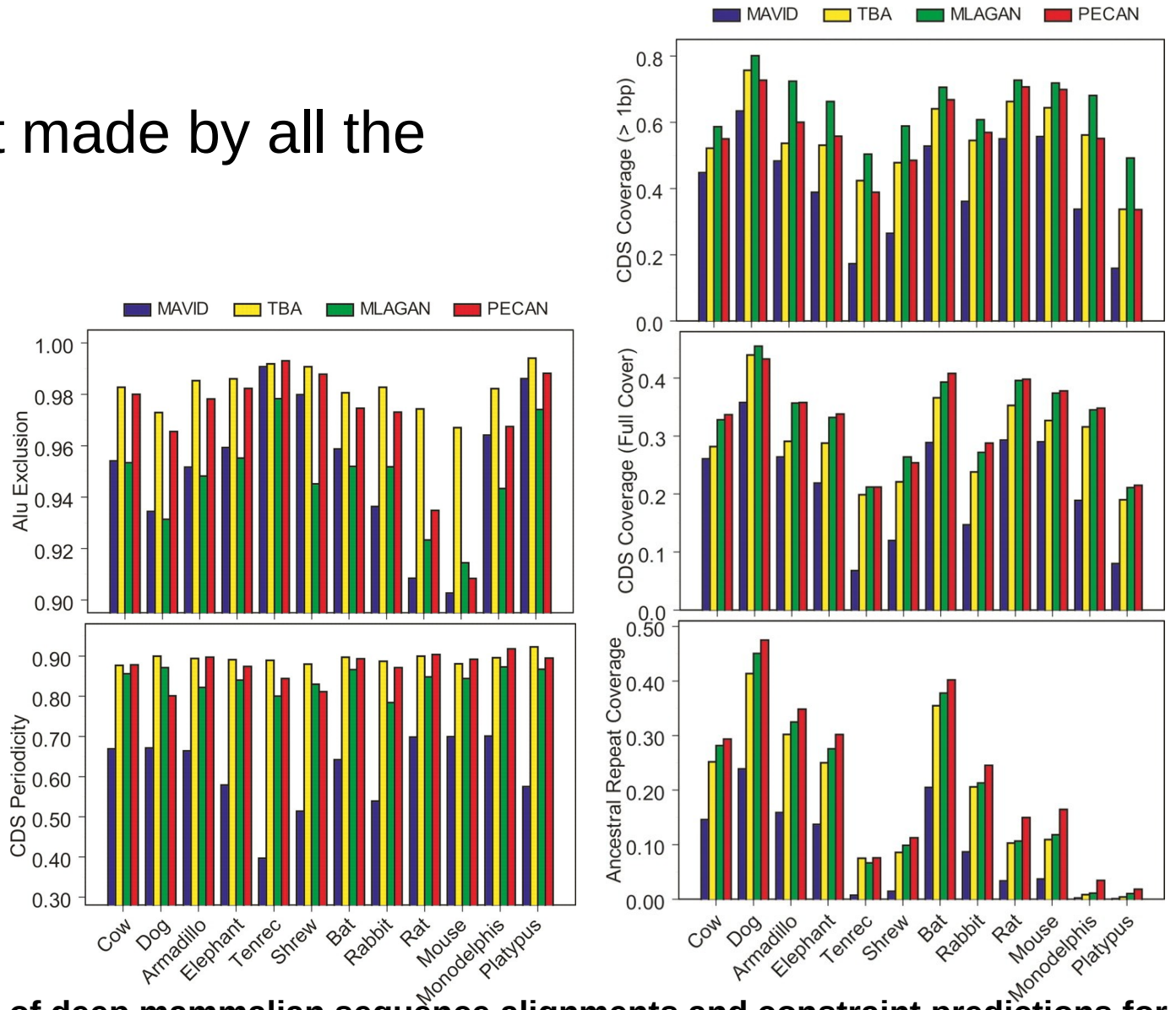
EMBL-EBI

Assessing genomic alignments

- Problem: golden sets only exist for protein coding regions
- Several approaches
  - Genomic annotations
  - Evol. models to create simulated sequences
  - Looking for misalignments
- “There are many wrong ways to compare alignments”

ENCODE assessment

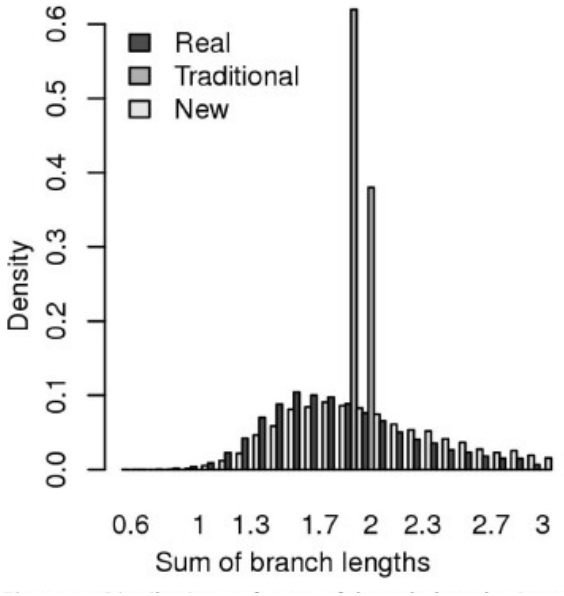
- Assessment made by all the authors
  - MAVID
  - TBA
  - MLAGAN
  - PECAN



Margulies et al., **Analyses of deep mammalian sequence alignments and constraint predictions for 1% of the human genome**, *Genome Research* 17, no. 6 (June 2007): 760-774.

New simulation method (Kim & Sinha)

- Independent study
- Usual simulation methods are too simplistic
- The quality of the alignments degrades more rapidly with insertion than with deletions



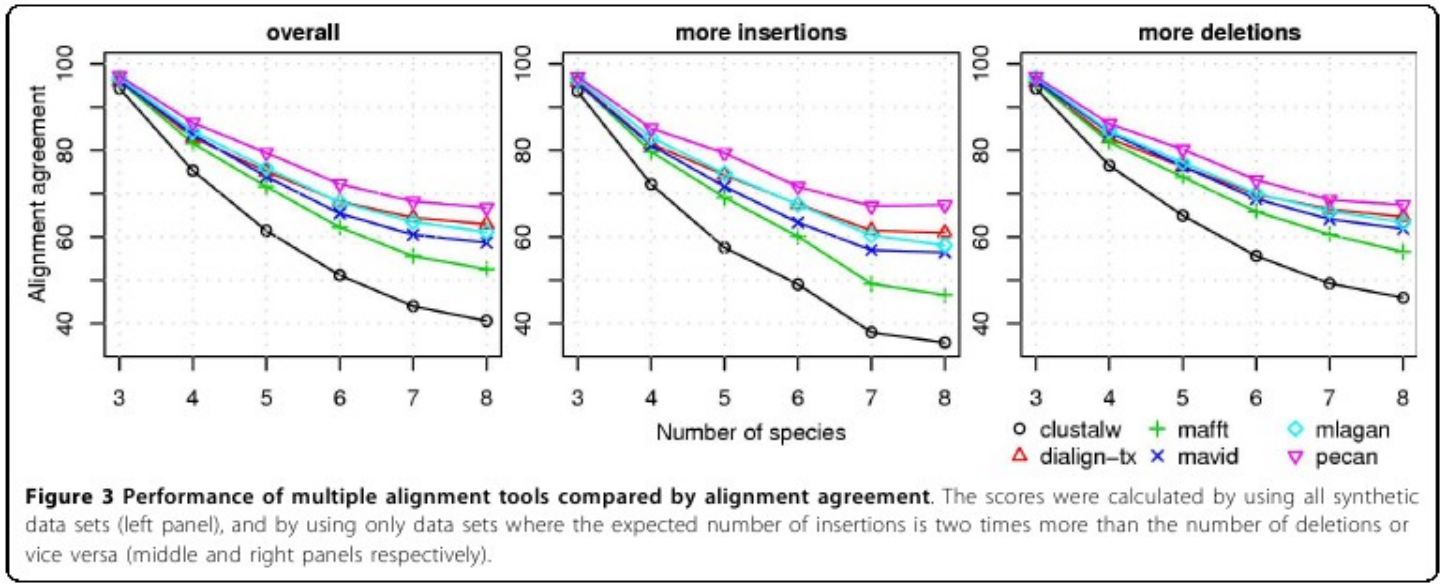


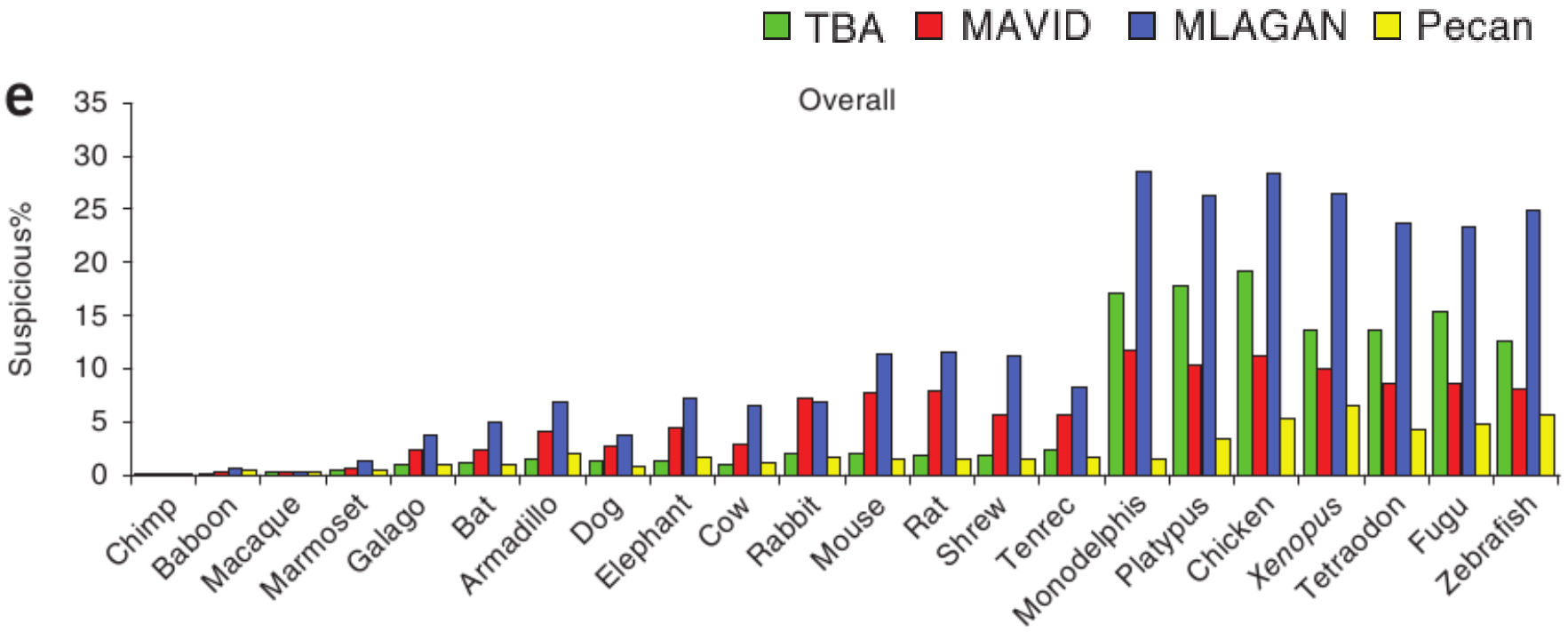
Figure 3 Performance of multiple alignment tools compared by alignment agreement. The scores were calculated by using all synthetic data sets (left panel), and by using only data sets where the expected number of insertions is two times more than the number of deletions or vice versa (middle and right panels respectively).

Kim & Sinha, **Towards realistic benchmarks for multiple alignments of non-coding sequences**, *BMC Bioinformatics* 11, no. 1 (2010): 54.

MSA: external assessment

Use StatSigMA-w to look for suspicious regions

i.e. region not better aligned than a random sequence



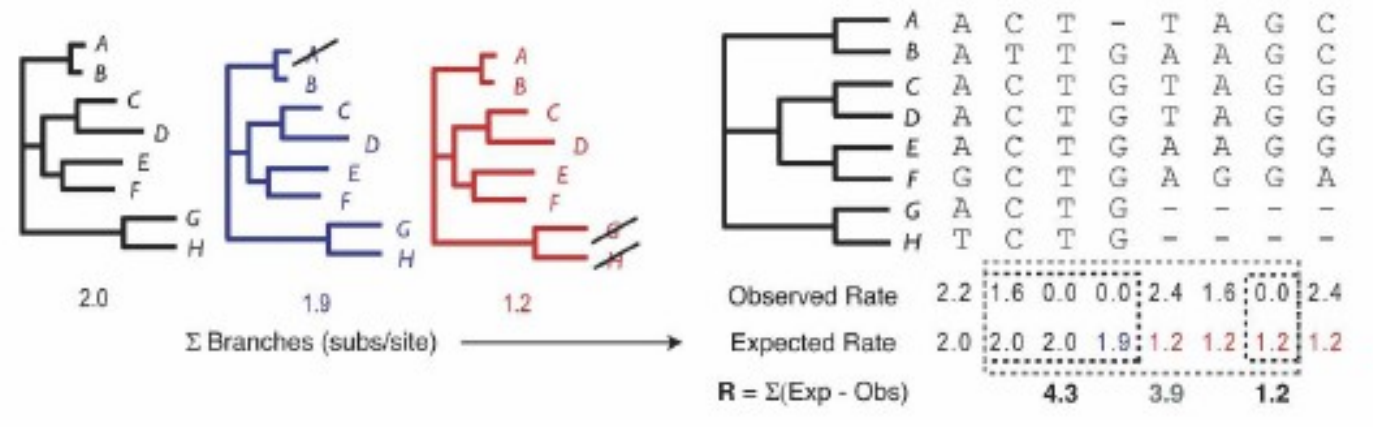
Chen & Tompa, **Comparative assessment of methods for aligning multiple genome sequences**, *Nat. Biotech* 28, no. 6 (June 2010): 567-572.

Sequence Conservation

EMBL-EBI

GERP

- Scores each position of the alignment according to the weighted number of “rejected substitutions”

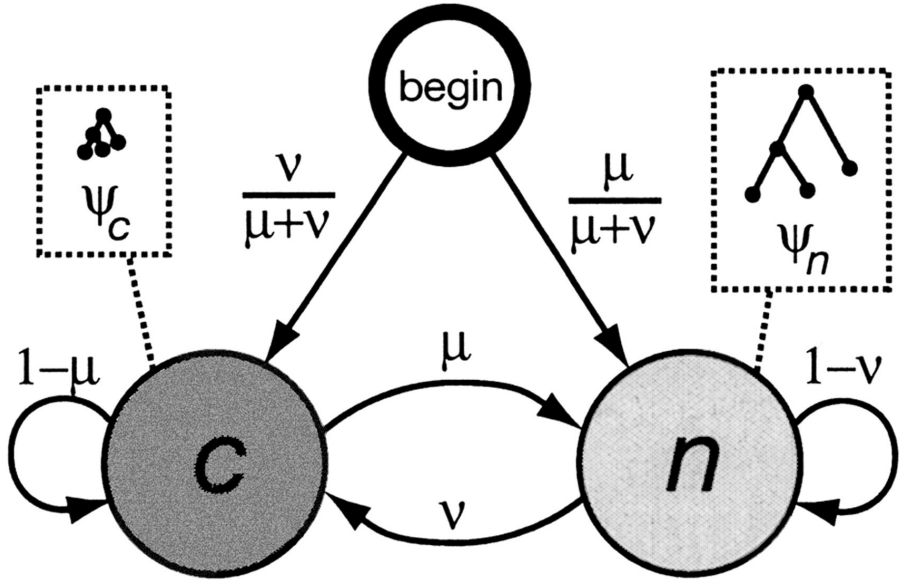


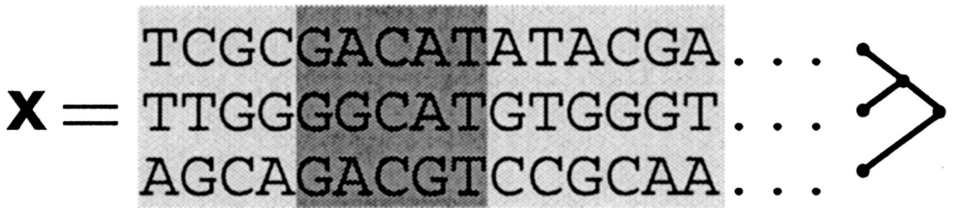
- Elements are defined as stretches of the alignment with high scores
- GERP v2 uses a permutation test to assign p-values and obtain significant elements only

Cooper et al., **Distribution and intensity of constraint in mammalian genomic sequence**, *Genome Research* 15, no. 7 (July 2005): 901-913.

PhastCons

- Uses a 2-states HMM to detect conserved regions:
  - neutral tree
  - scaled down neutral tree
- Also provides scores: the probability of that position to be in the “conserved” state
- Does not work well with global alignments because of the amount of gaps

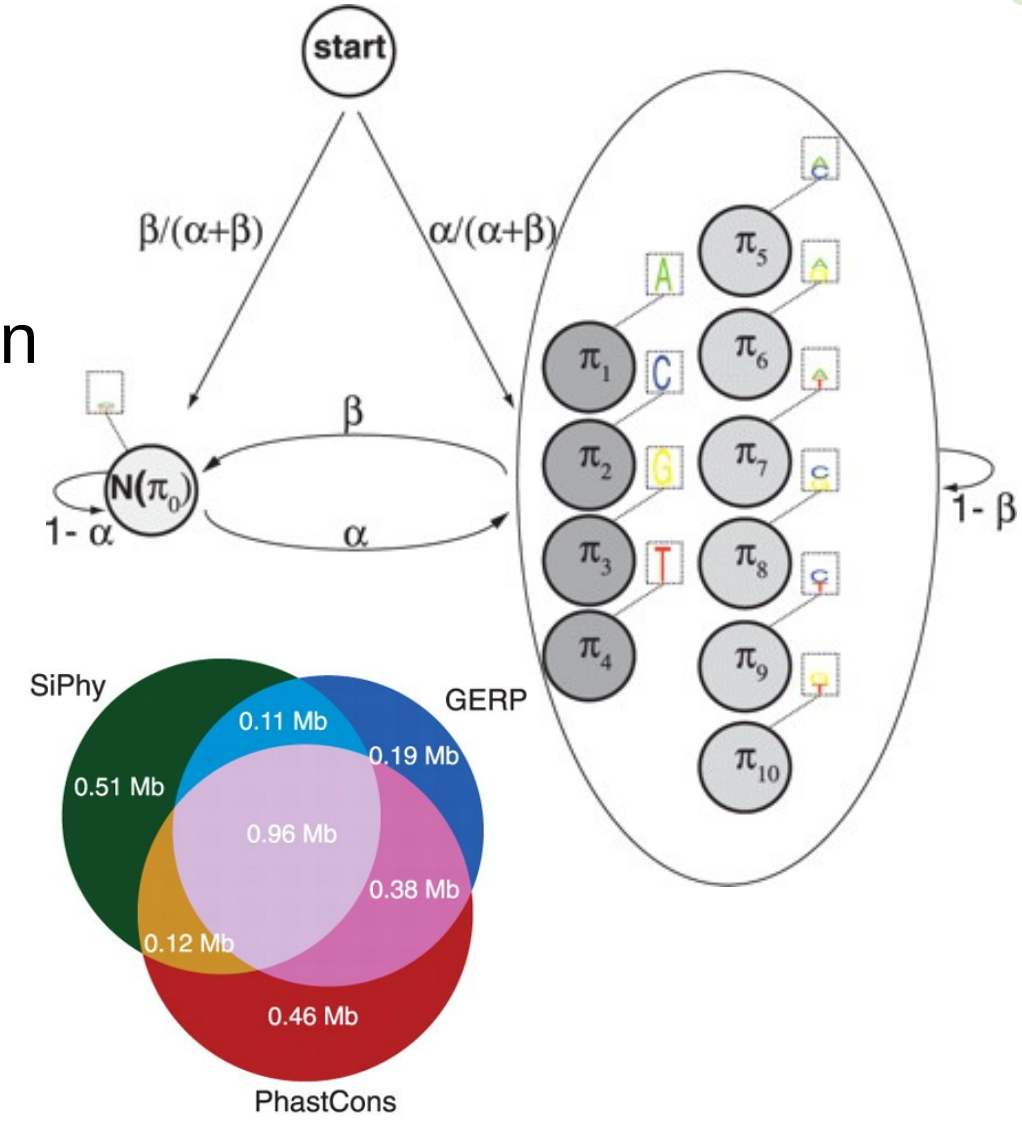




Siepel et al., **Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes**, *Genome Research* 15, no. 8 (2005): 1034-1050.

SiPhy-HMM

- Similar to PhastCons
- Uses biased substitution patterns to find conservation
- Works better especially on coding region, because of the 2D sites.



Garber et al., **Identifying novel constrained elements by exploiting biased substitution patterns**, *Bioinformatics* 25, no. 12 (2009): i54-62.