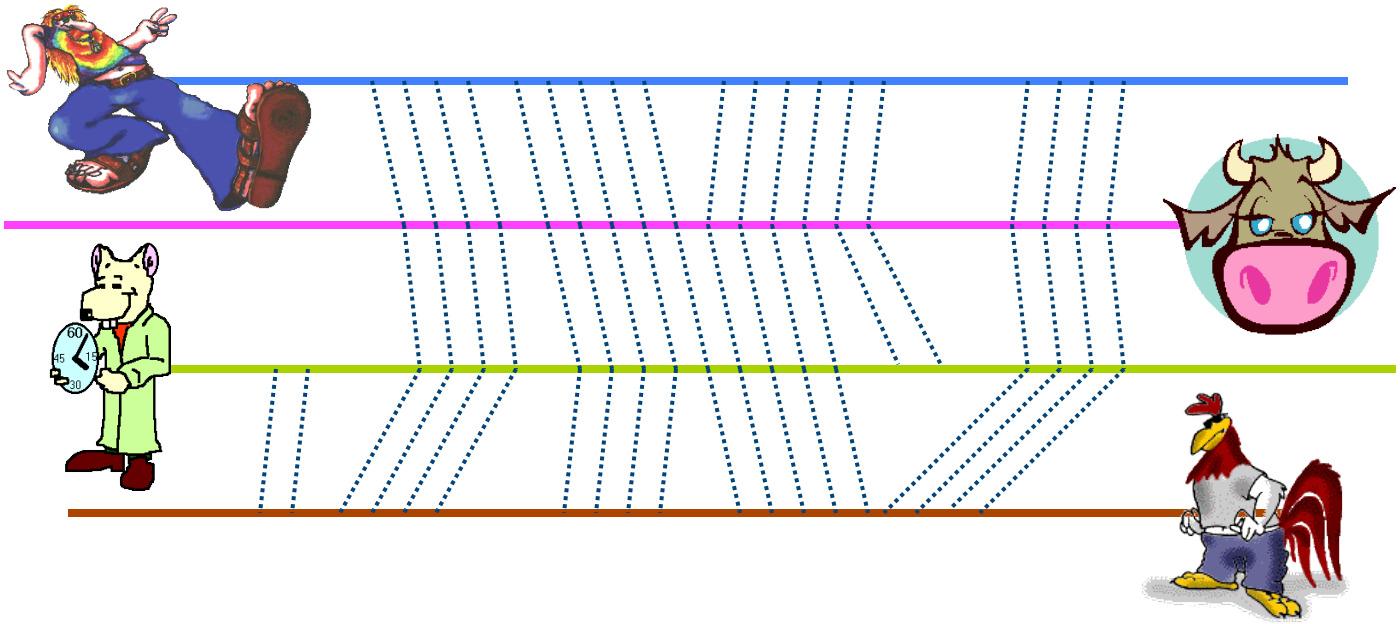
Multiple Sequence Alignment

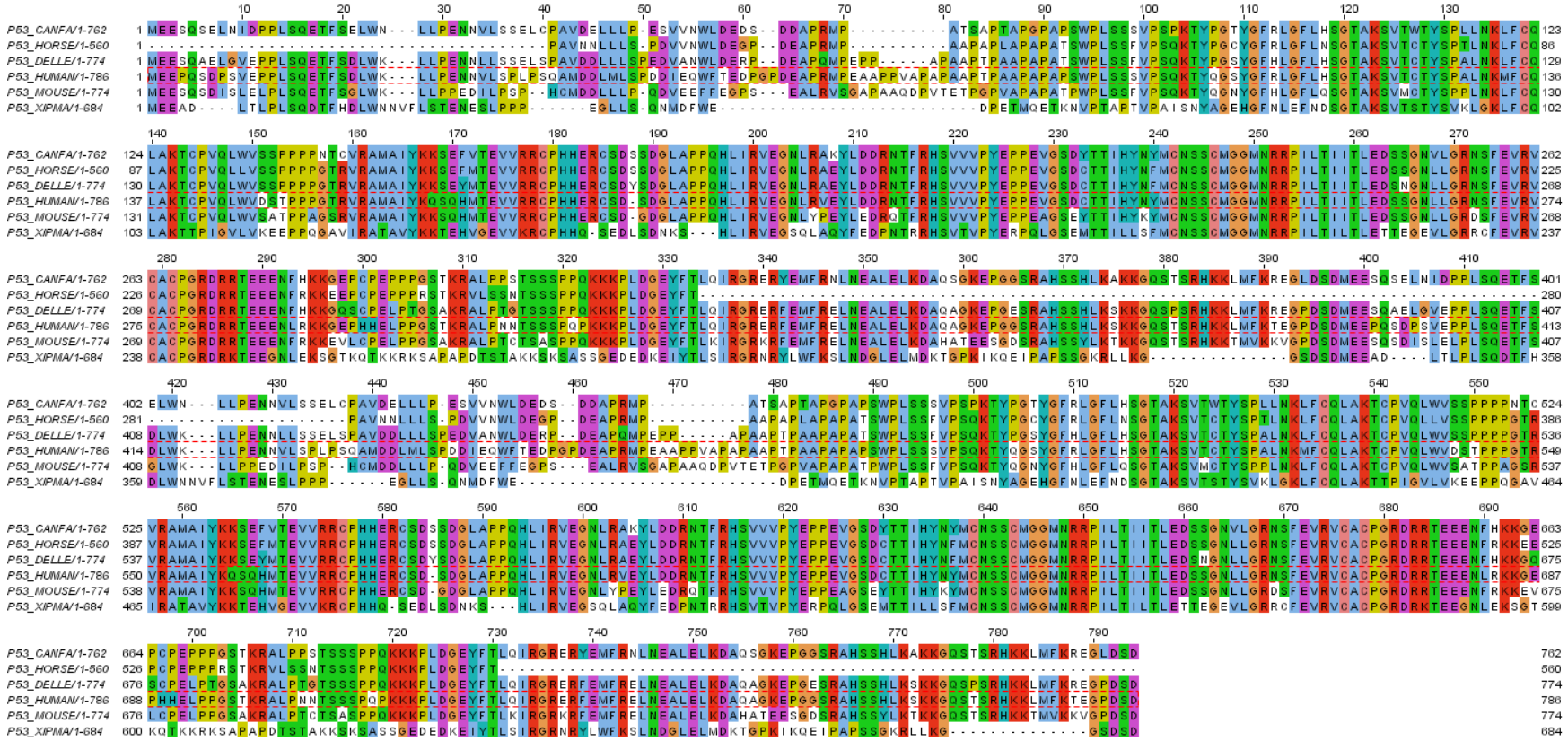
Gene Finding, Conserved Elements



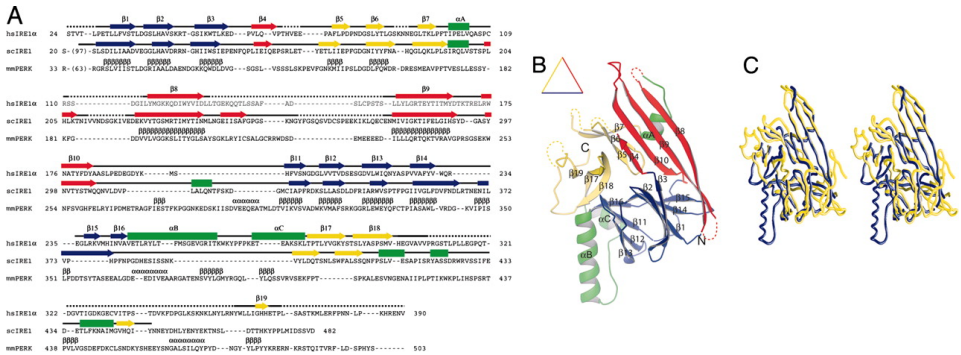
Definition

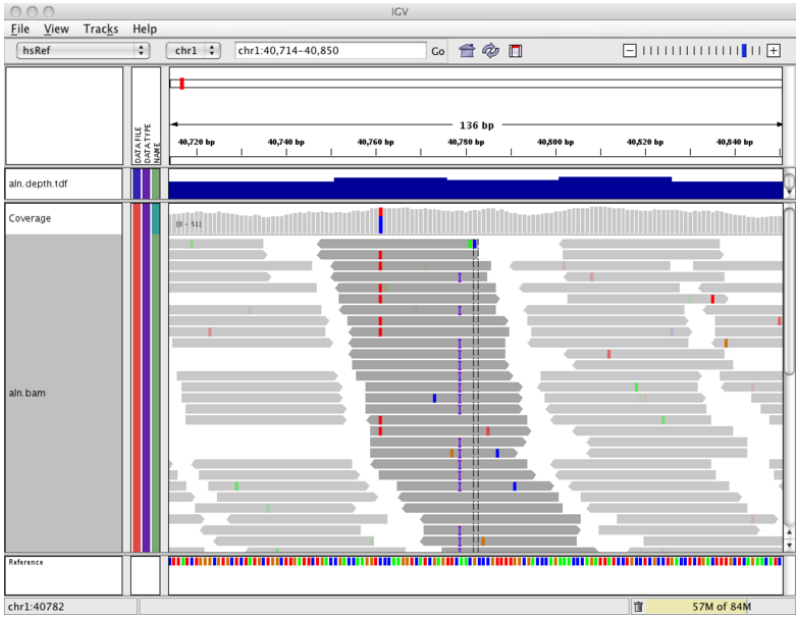
Given N sequences x^1, x^2,…, x^N:

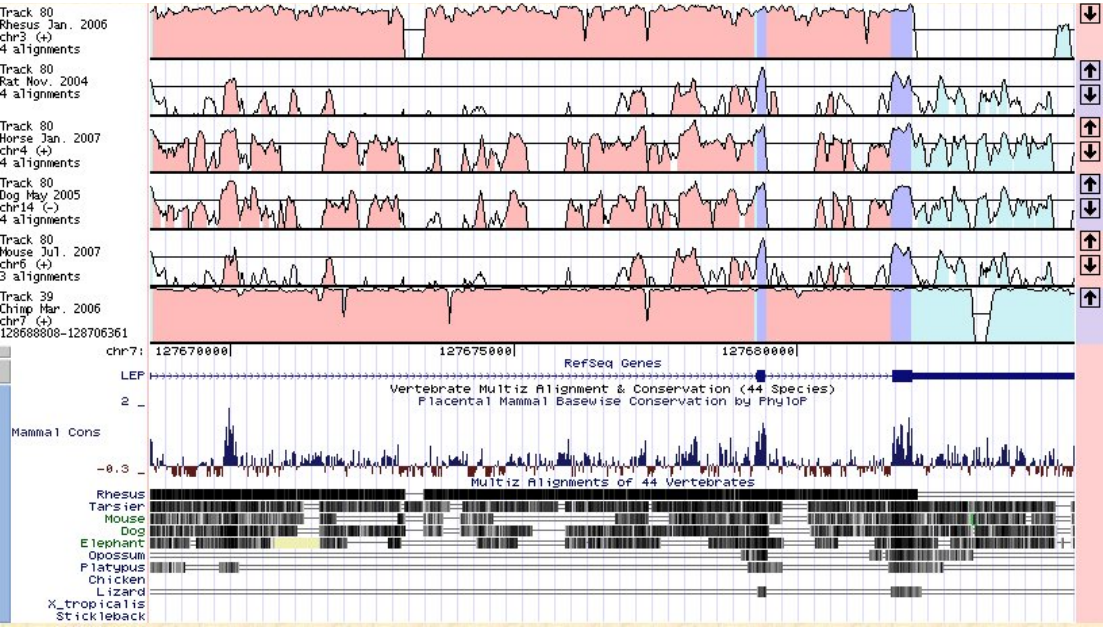
- Insert gaps (-) in each sequence x^i, such that
  - All sequences have the same length L
  - Score of the global map is maximum



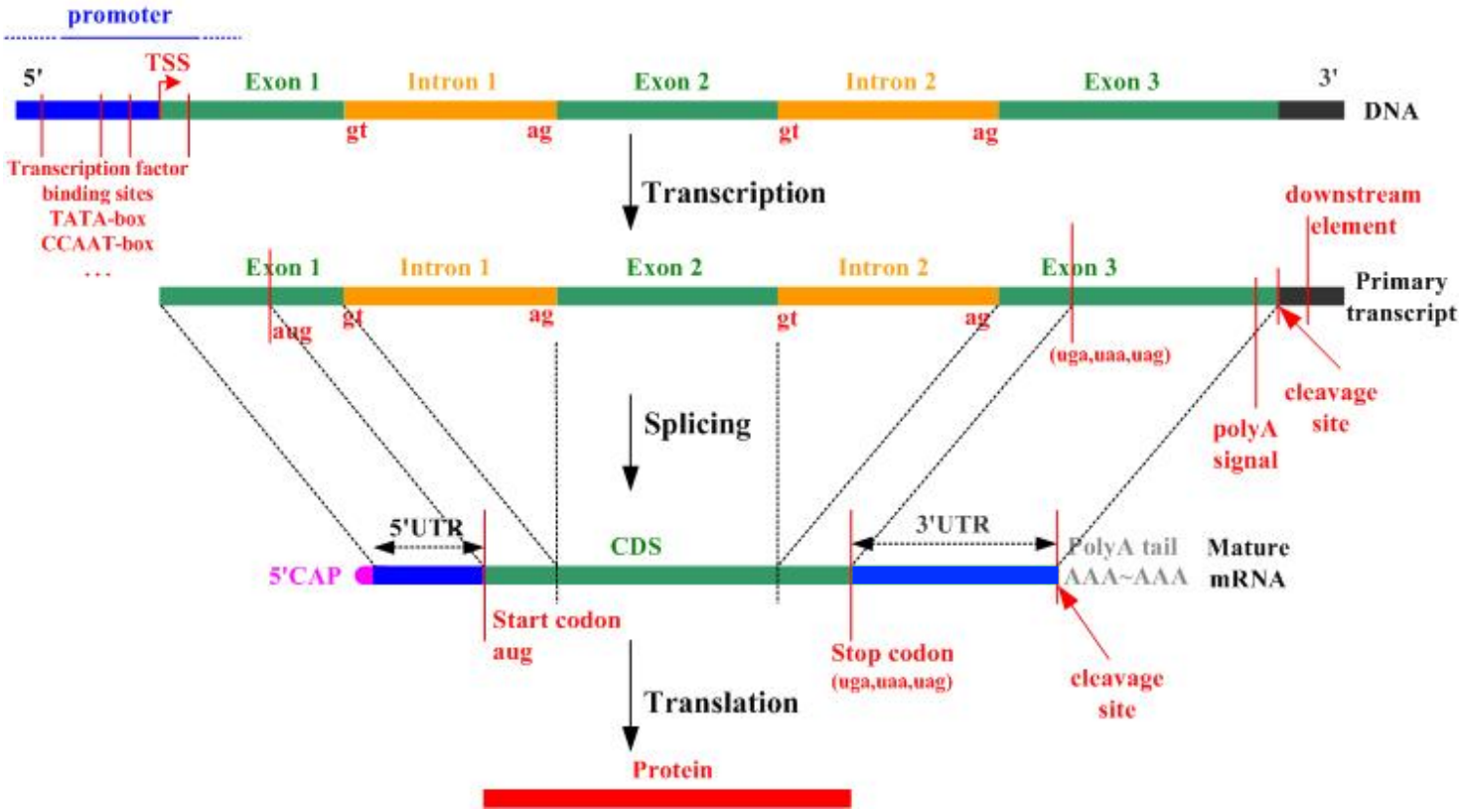
Applications

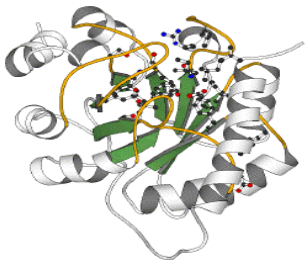






Gene structure





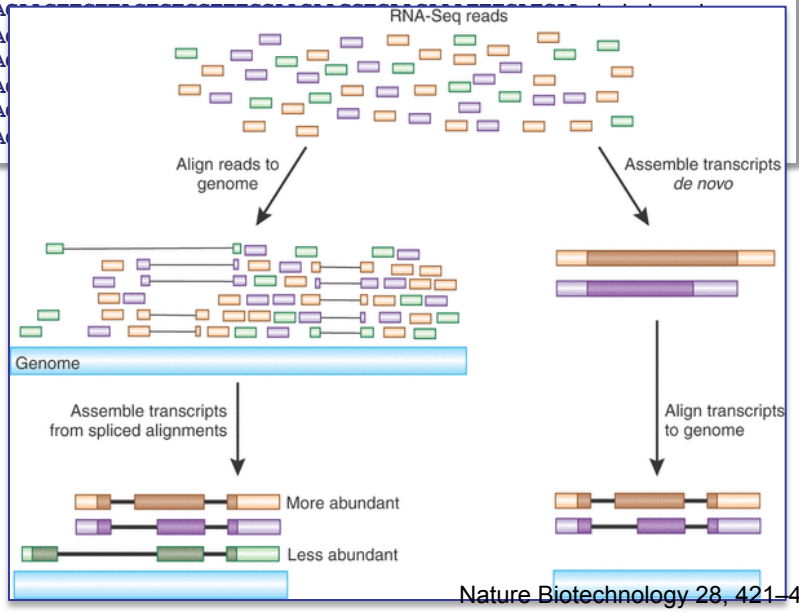
exon = protein-coding
intron = non-coding

Codon:
A triplet of nucleotides that is converted to one amino acid

Gene Finding

- Classes of Gene predictors
  - *Ab initio*: Only look at the genomic DNA of target genome
  - *De novo*: Target genome + aligned informant genome(s)

```
Human       tttcttagACTTTAAAGCTGTCAAGCCGTGTTCTAGATAAAATAAGTATTGGACAACTTGTTAGTCTCCTTTCCAACAACCTGAACAAATTTGATGAAgtatgtaccta
Macaque     tttcttagACTTTAAAGCTGTCAAGCCGTGTTCTAGATAAAATAAGTATTGGACAACTTGTTAGTCTCCTTTCCAACAACCTGAACAAATTTGATGAAgtatgtaccta
Mouse       ttgcttagACTTTAAAGTTGTCAAGCCGCGTTCTTGATAAAATAAGTATTGGACAACTTGTTAGTCTTCTTTCCAACAACCTGAACAAATTTGATGAAgtatgta-cca
Rat         ttgcttagACTTTAAAGTTGTCAAGCCGTGTTCTTGATAAAATAAGTATTGGACAACTTATTAGTCTTCTTTCCAACAACCTGAACAAATTTGATGAAgtatgtaccca
Rabbit      t--attagACTTTAAAGCTGTCAAGCCGTGTTCTAGATAAAATAAGTATTGGGCAACTTATTAGTCTCCTTTCCAACAACCTGAACAAATTTGATGAAgtatgtaccta
Dog         t-cattagACTTTAAAGCTGTCAAGCCGTGTTCTGGATAAAATAAGTATTGGACAACTTGTTAGTCTCCTTTCCAACAACCTGAACAAATTCGATGAAgtatgtaccta
Cow         t-cattagACTTTGAAGCTATCAAGCCGTGTTCTGGATAAAATAAGTATTGGA
Armadillo   gca--tagACCTTAAAACTGTCAAGCCGTGTTTTAGATAAAATAAGTATTGGA
Elephant    gct-ttagACTTTAAAACTGTCCAGCCGTGTTCTTGATAAAATAAGTATTGGA
Tenrec      tc-cttagACTTTAAAACTTTCGAGCCGGGTTCTAGATAAAATAAGTATTGGA
Opossum     ---tttagACCTTAAAACTGTCAAGCCGTGTTCTAGATAAAATAAGCACTGGA
Chicken     ----ttagACCTTAAAACTGTCAAGCAAAGTTCTAGATAAAATAAGTACTGGA
```



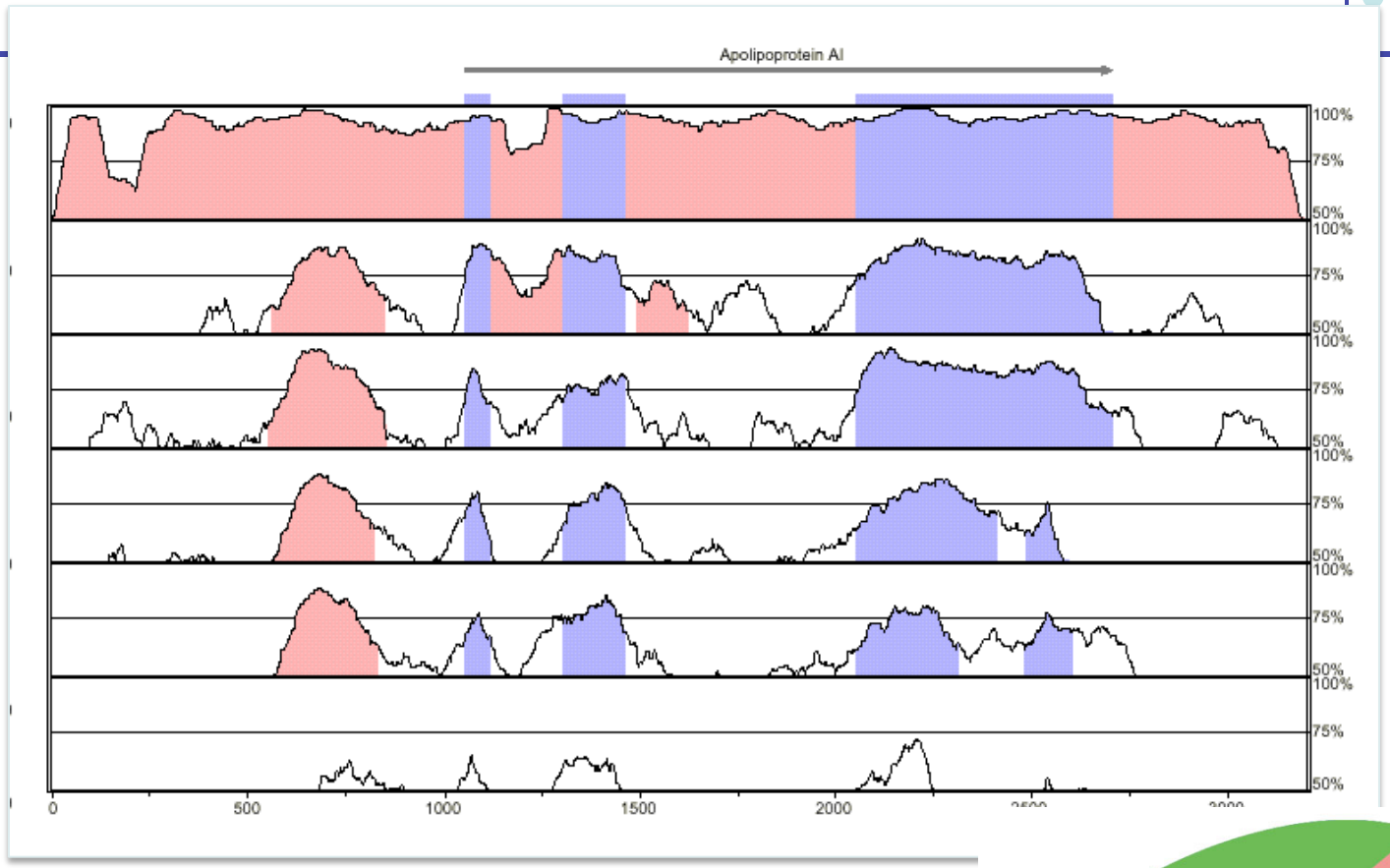
- RNA-seq based approaches

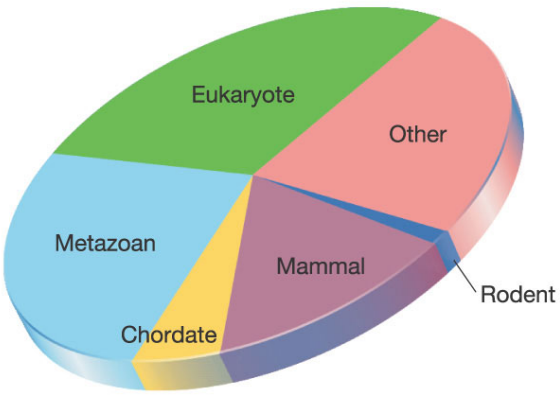
Nature Biotechnology 28, 421–423 (2010)

Using Comparative Information

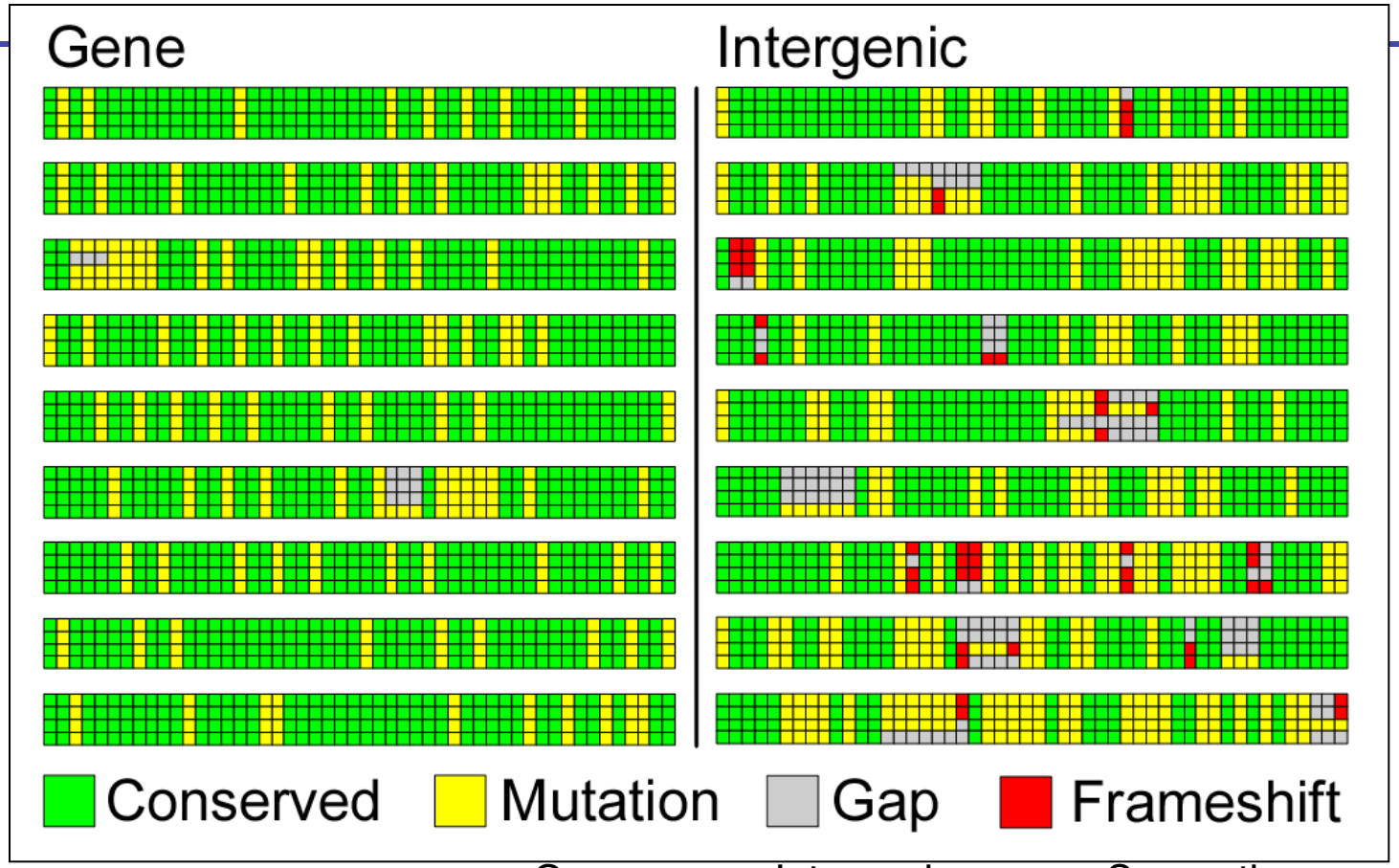
```
Alignment 1
 Seq1: human
 Seq2: macaque
 Reg id: 75
 Reg length: 100
 Plot min: 50
 Regions: 7

Alignment 2
 Seq1: human
 Seq2: pig
 Reg id: 75
 Reg length: 100
 Plot min: 50
 Regions: 6

Alignment 3
 Seq1: human
 Seq2: rabbit
 Reg id: 75
 Reg length: 100
 Plot min: 50
 Regions: 4

Alignment 4
 Seq1: human
 Seq2: mouse
 Reg id: 75
 Reg length: 100
 Plot min: 50
 Regions: 5

Alignment 5
 Seq1: human
 Seq2: rat
 Reg id: 75
 Reg length: 100
 Plot min: 50
 Regions: 5

Alignment 6
 Seq1: human
 Seq2: chicken
 Reg id: 75
 Reg length: 100
 Plot min: 50
 Regions: 0

Resolution: 4
Window size: 100
Start: 1
```

Exon
UTR
CNS



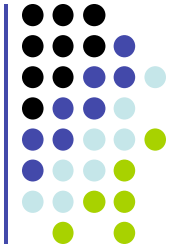


Patterns of Conservation



	Genes	Intergenic		Separation
Mutations	30%	58%	→	2-fold
Gaps	1.3%	14%	→	10-fold
Frameshifts	0.14%	10.2%	→	75-fold

Scoring Function: Sum Of Pairs



Definition: Induced pairwise alignment

A pairwise alignment induced by the multiple alignment

Example:

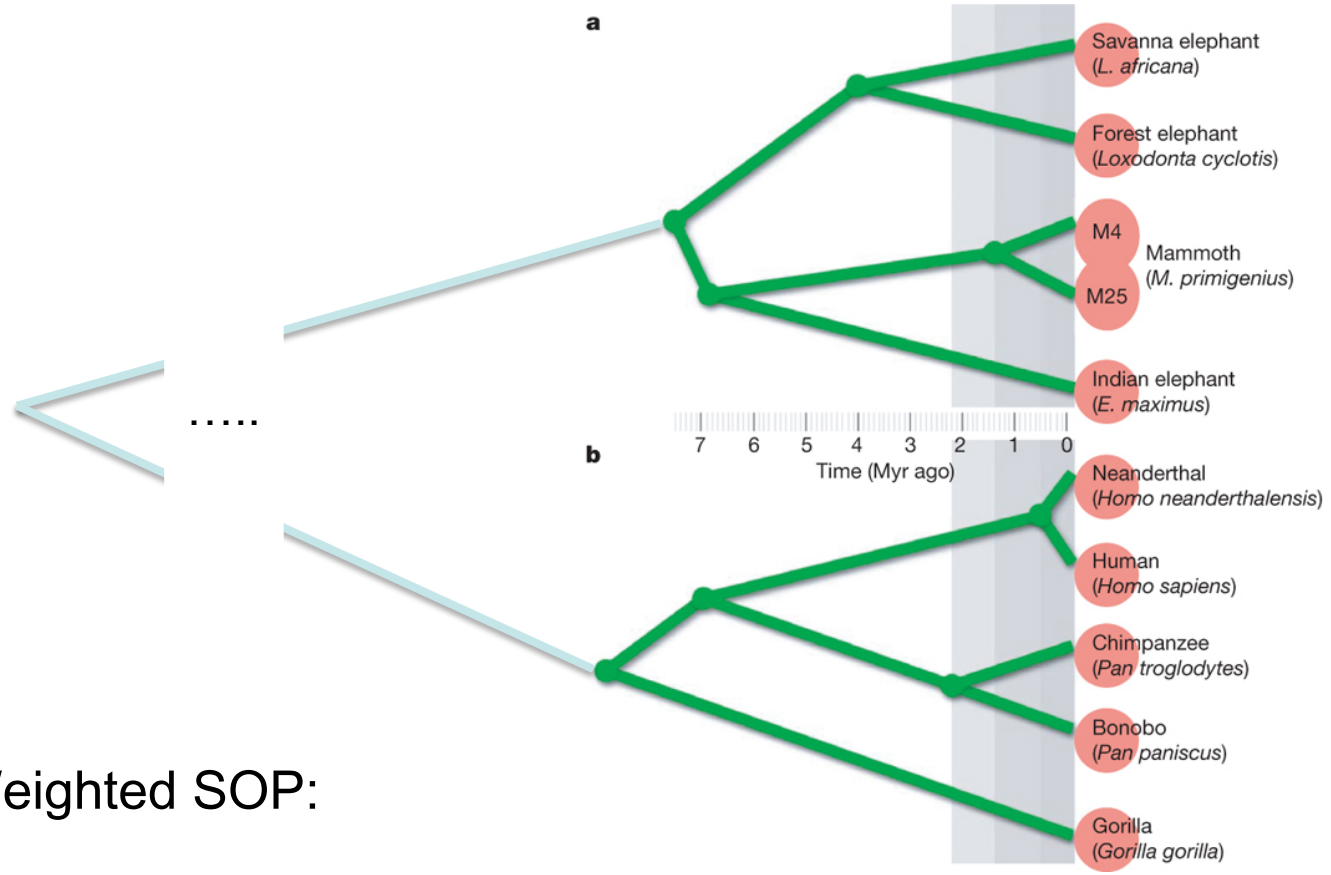
```
x:  AC-GCGG-C
y:  AC-GC-GAG
z:  GCCGC-GAG
```

Induces:

```
x: ACGCGG-C;   x: AC-GCGG-C;   y: AC-GCGAG
y: ACGC-GAC;   z: GCCGC-GAG;   z: GCCGCGAG
```

Sum Of Pairs (cont'd)

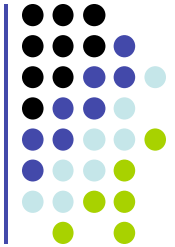
- Heuristic way to incorporate evolution tree:



- Weighted SOP:

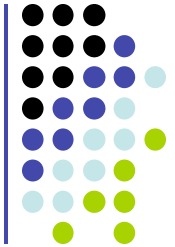
$$S(m) = \Sigma_{k<l} w_{kl} s(m^k, m^l)$$

A Profile Representation



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

A  0  1  0  0  0  0  1  0  0 .8  0  0  0  0
C .6  0  0  0  1  0  0 .4  1  0 .6 .2  0  0
G  0  0  1 .2  0  0  0  0  0 .2  0  0 .4  1
T .2  0  0  0  0  1  0 .6  0  0  0  0 .2  0
- .2  0  0 .8  0  0  0  0  0  0 .4 .8 .4  0
```

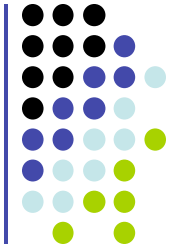
- Given a multiple alignment M = $m_1 \ldots m_n$
  - Replace each column m_i with profile entry p_i
    - Frequency of each letter in Σ
    - # gaps
    - Optional: # gap openings, extensions, closings
  - Can think of this as a “likelihood” of each letter in each position



Multiple Sequence Alignments

Algorithms

Multidimensional DP



Generalization of Needleman-Wunsh:

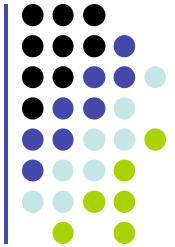
$$S(m) = \sum_i S(m_i)$$

(sum of column scores)

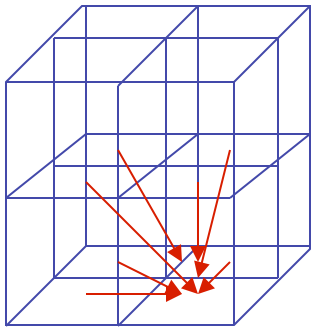
$F(i_1,i_2,\ldots,i_N)$: Optimal alignment up to $(i_1, \ldots, i_N)$

$F(i_1,i_2,\ldots,i_N) = \max_{\text{(all neighbors of cube)}}(F(\text{nbr})+S(\text{nbr}))$

Multidimensional DP

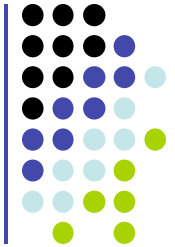


- Example: in 3D (three sequences):
- 7 neighbors/cell



$$
\begin{aligned}
F(i,j,k) = \max\{ & F(i-1, j-1, k-1) + S(x_i, x_j, x_k), \\
& F(i-1, j-1, k\ \ ) + S(x_i, x_j, -), \\
& F(i-1, j\ \ , k-1) + S(x_i, -, x_k), \\
& F(i-1, j\ \ , k\ \ ) + S(x_i, -, -), \\
& F(i\ \ , j-1, k-1) + S(-, x_j, x_k), \\
& F(i\ \ , j-1, k\ \ ) + S(-, x_j, -), \\
& F(i\ \ , j\ \ , k-1) + S(-, -, x_k) \}
\end{aligned}
$$

Multidimensional DP



Running Time:

1. Size of matrix: L^N;

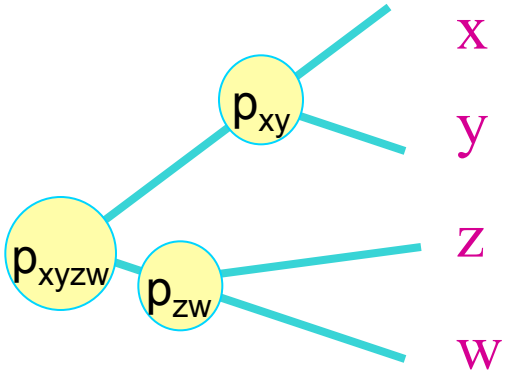
   Where L = length of each sequence

   N = number of sequences

2. Neighbors/cell: $2^N - 1$

Therefore…… $O(2^N L^N)$

Progressive Alignment

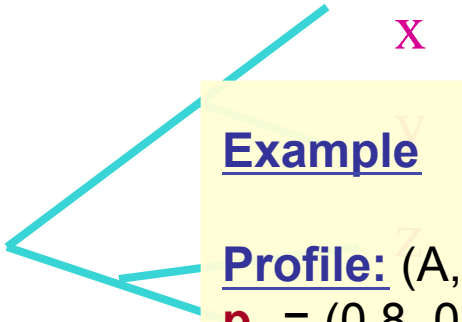


- When evolutionary tree is known:
  - Align closest first, in the order of the tree
  - In each step, align two sequences x, y, or profiles p_x, p_y, to generate a new alignment with associated profile p_{result}

Weighted version:

- Tree edges have weights, proportional to the divergence in that edge
- New profile is a weighted average of two old profiles

Progressive Alignment



- When evolutionary tree is known:
  - Align closest first, in the order of the tree
  - In each step, align two sequence profiles and generate new alignment with associated profile p_{result}

  Weighted version:
  - Tree edges have weights, proportional to the divergence in that edge
  - New profile is a weighted average of two old profiles

Example

Profile: (A, C, G, T, -)
p_x = (0.8, 0.2, 0, 0, 0)
p_y = (0.6, 0, 0, 0, 0.4)

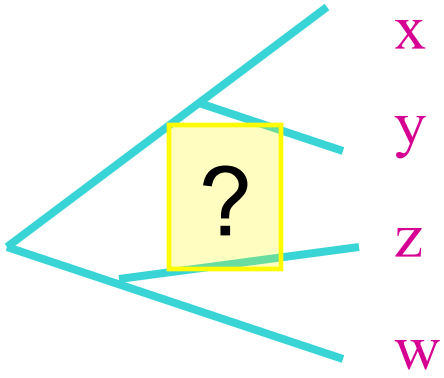
$s(p_x, p_y)$ = 0.8*0.6*s(A, A) + 0.2*0.6*s(C, A)
+ 0.8*0.4*s(A, -) + 0.2*0.4*s(C, -)

Result: p_{xy} = (0.7, 0.1, 0, 0, 0.2)

$s(p_x, -)$ = 0.8*1.0*s(A, -) + 0.2*1.0*s(C, -)

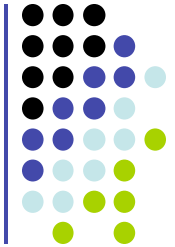
Result: p_{x-} = (0.4, 0.1, 0, 0, 0.5)

Progressive Alignment



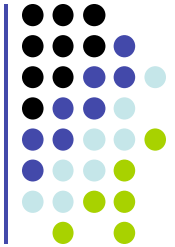
- When evolutionary tree is unknown:
  - Perform all pairwise alignments
  - Define distance matrix D, where D(x, y) is a measure of evolutionary distance, based on pairwise alignment
  - Construct a tree *(UPGMA / Neighbor Joining / Other methods)*
  - Align on the tree

Heuristics to improve alignments



- Iterative refinement schemes
- A*-based search
- Consistency
- Simulated Annealing
- …

Iterative Refinement



One problem of progressive alignment:

- Initial alignments are “frozen” even when new evidence comes

Example:

```
x:    GAAGTT
y:    GAC-TT
```

Frozen!

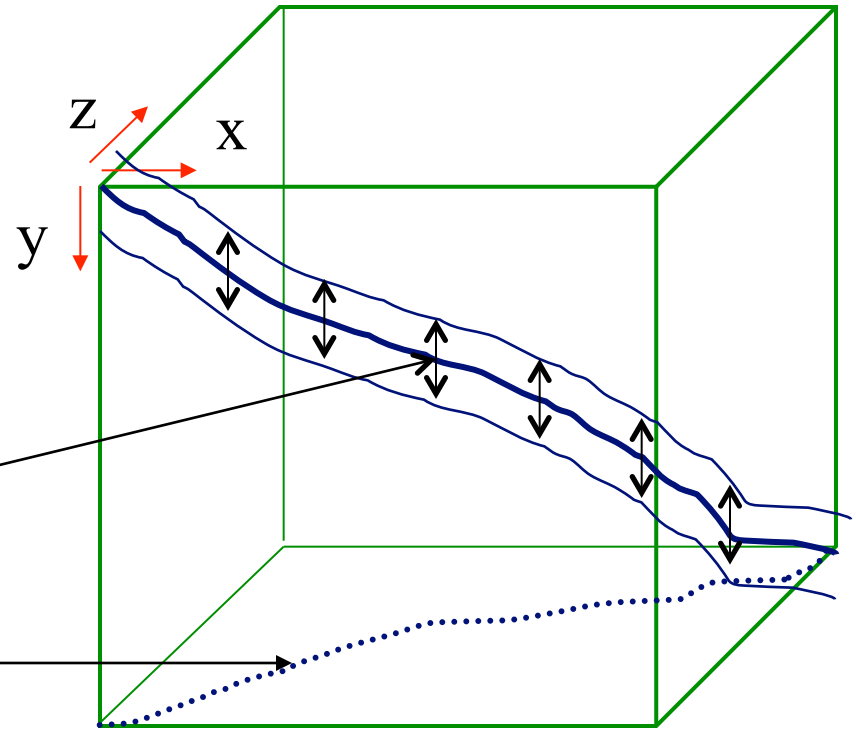
```
z:    GAACTG
w:    GTACTG
```

Now clear correct y = GA-CTT

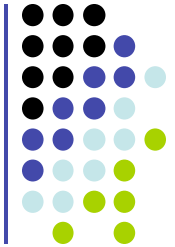
Iterative Refinement

Algorithm (Barton-Stenberg):

1. For j = 1 to N,
   Remove x^j, and realign to $x^1 \ldots x^{j-1} x^{j+1} \ldots x^N$
2. Repeat 4 until convergence



Iterative Refinement



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

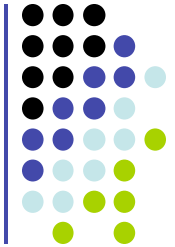
```
x:    GAAGTTA
y:    GAC-TTA
z:    GAACTGA
w:    GTACTGA
```

After realigning y:

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

+ 3 matches

Iterative Refinement

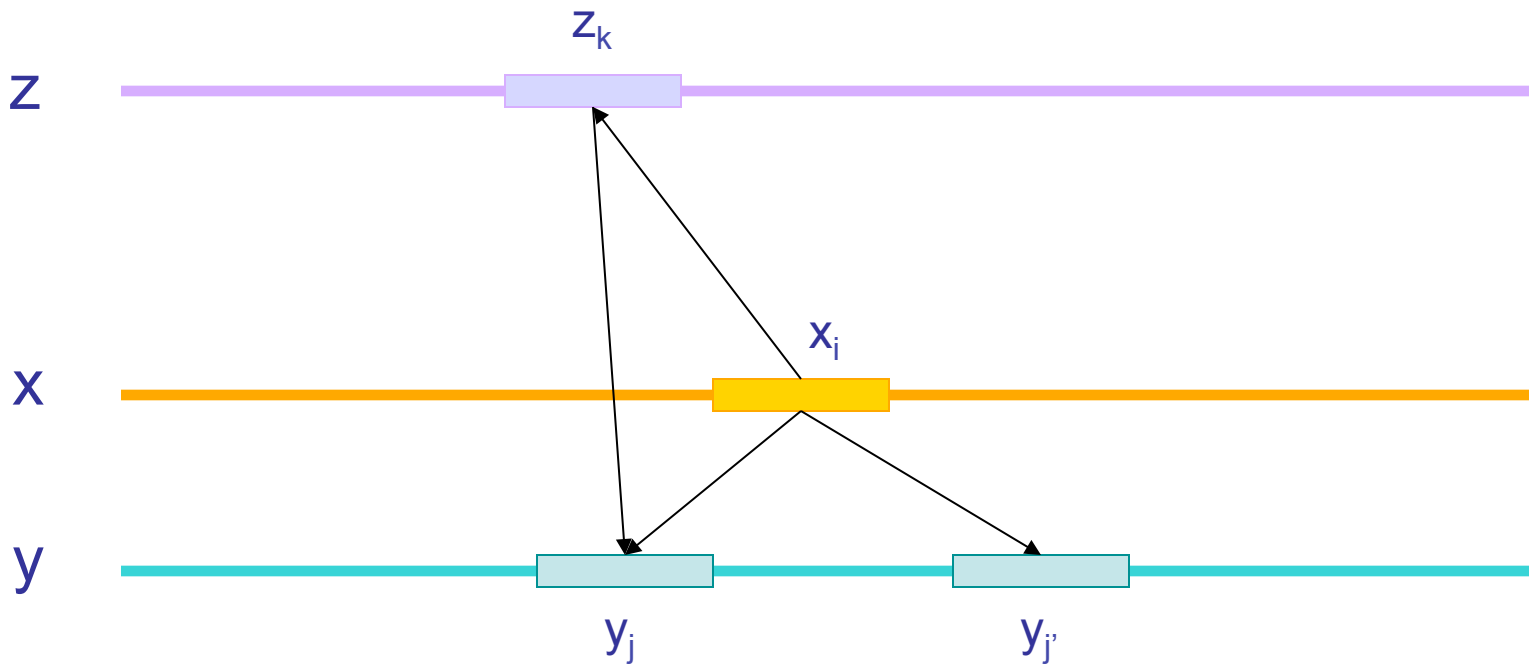


Example not handled well:

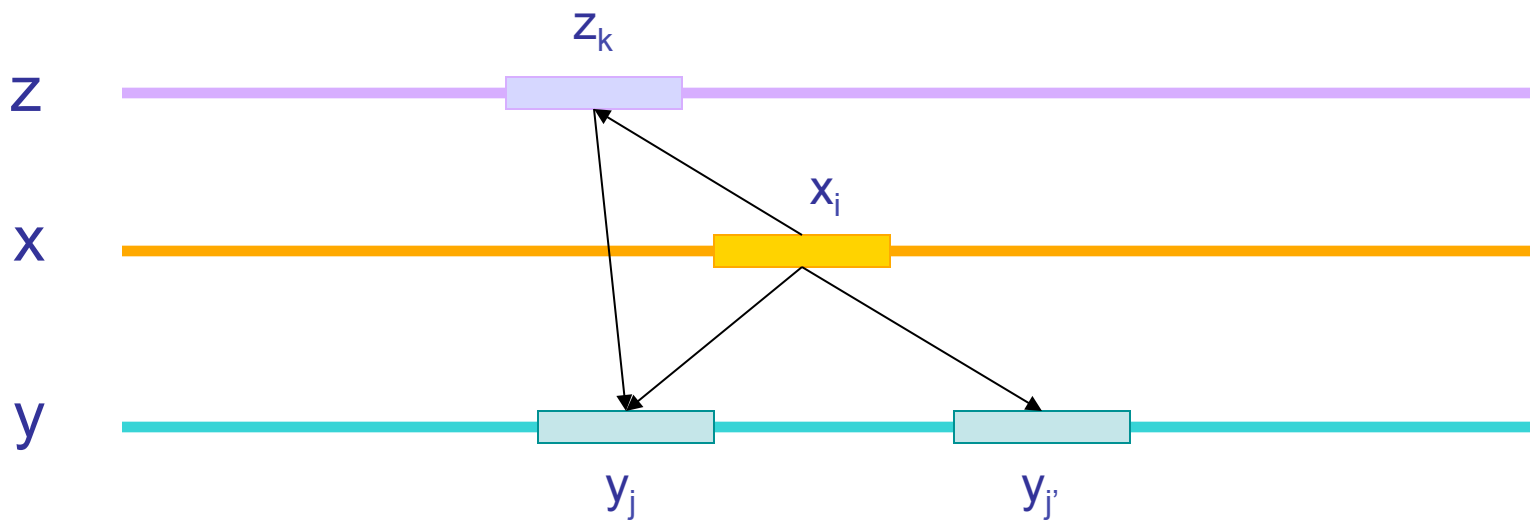
```
x:    GAAGTTA
y1:   GAC-TTA
y2:   GAC-TTA
y3:   GAC-TTA

z:    GAACTGA
w:    GTACTGA
```

Realigning any single y_i changes nothing

Consistency



Consistency

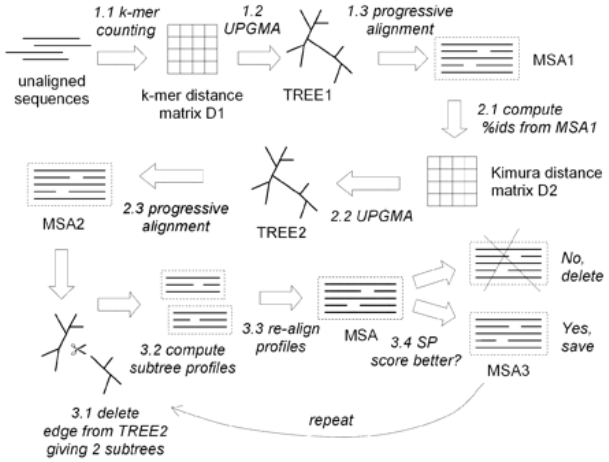


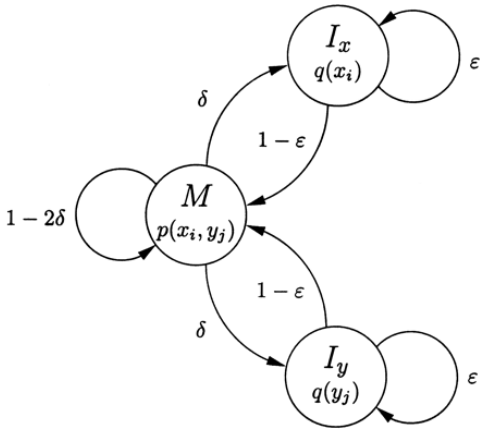
Basic method for applying consistency

- Compute all pairs of alignments xy, xz, yz, …
- When aligning x, y during progressive alignment,
  - For each (x_i, y_j), let $s(x_i, y_j)$ = function_of(x_i, y_j, a_{xz}, a_{yz})
  - Align x and y with DP using the modified s(.,.) function

Real-world protein aligners

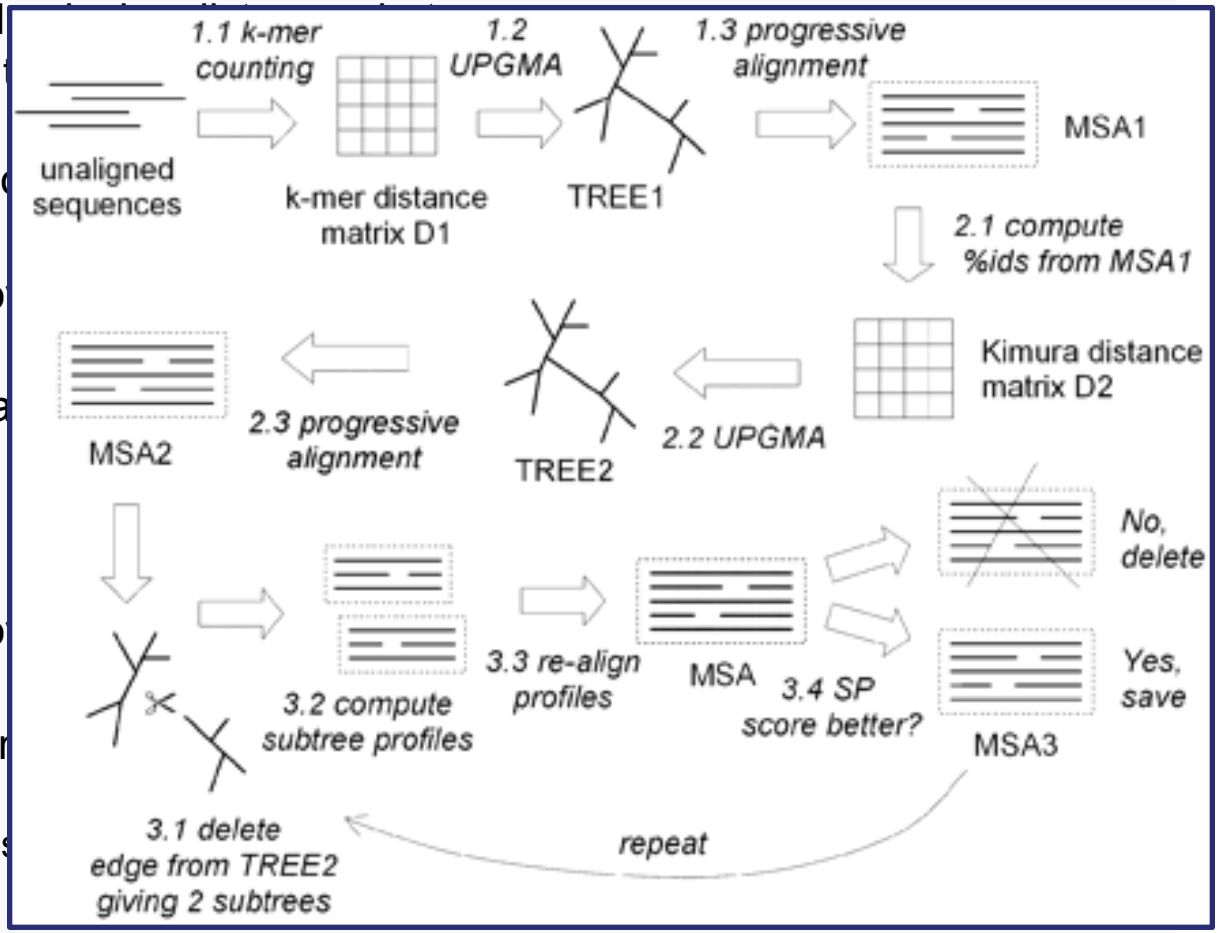
- MUSCLE
  - High throughput
  - One of the best in accuracy
- ProbCons
  - High accuracy
  - Reasonable speed



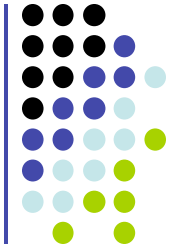


MUSCLE at a glance

1. Fast measurement of all
   - $D_{DRAFT}(x, y)$ defined in
2. Build tree T_{DRAFT} based
3. Progressive alignment o
4. Measure new Kimura-ba
5. Build tree T based on D
6. Progressive alignment o
7. Iterative refinement; for
   - *Tree Partitioning:* Split
   - If new alignment M' has



PROBCONS at a glance



1. Computation of all posterior matrices M_{xy} : $M_{xy}(i, j) = \text{Prob}(x_i \sim y_j)$, using a HMM

2. Re-estimation of posterior matrices M'_{xy} with ***probabilistic consistency***
   - $M'_{xy}(i, j) = 1/N \sum_{\text{sequence } z} \sum_k M_{xz}(i, k) \times M_{yz}(j, k)$; $M'_{xy} = \text{Avg}_z(M_{xz}M_{zy})$

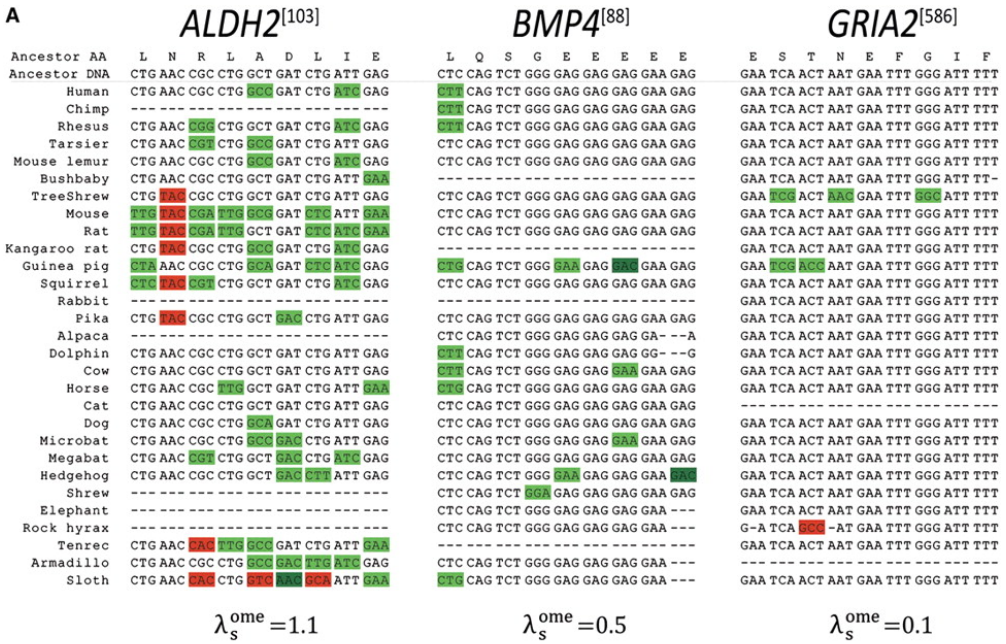
3. Compute for every pair x, y, the maximum expected accuracy alignment
   - A_{xy}: alignment that maximizes $\sum_{\text{aligned (i, j) in A}} M'_{xy}(i, j)$
   - Define $E(x, y) = \sum_{\text{aligned (i, j) in Axy}} M'_{xy}(i, j)$

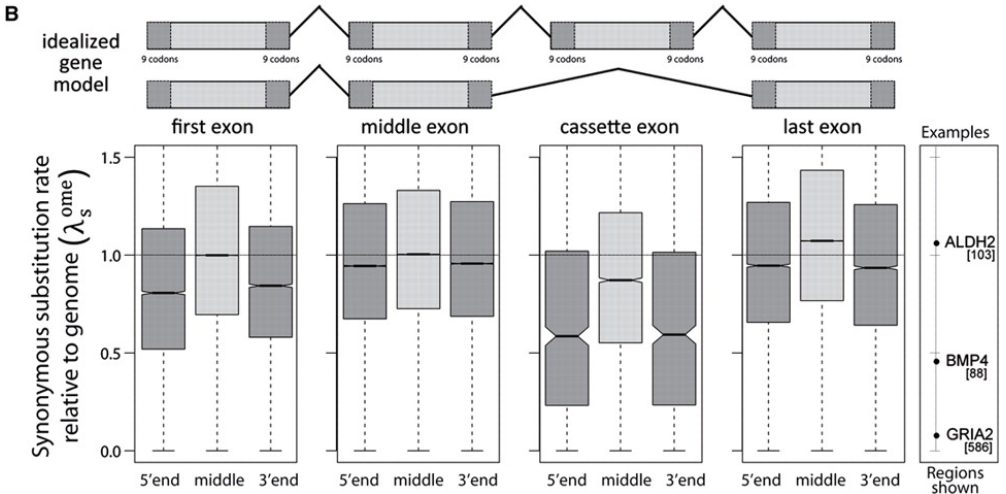
4. Build tree T with hierarchical clustering using similarity measure E(x, y)

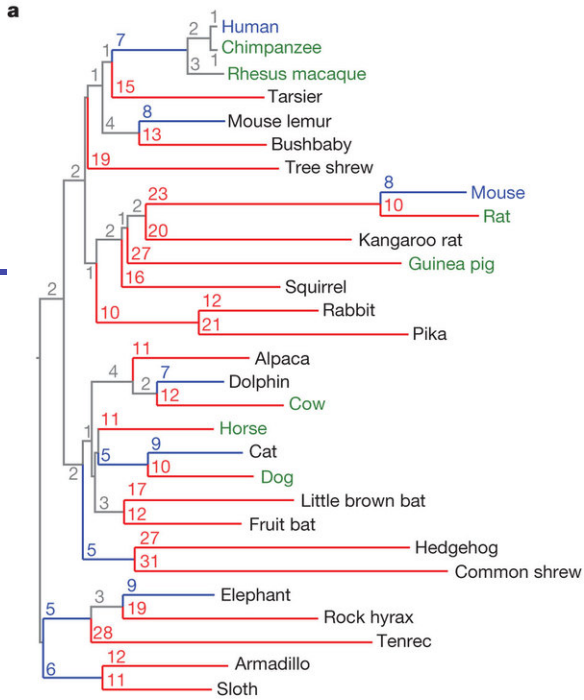
5. Progressive alignment on T to maximize E(.,.)

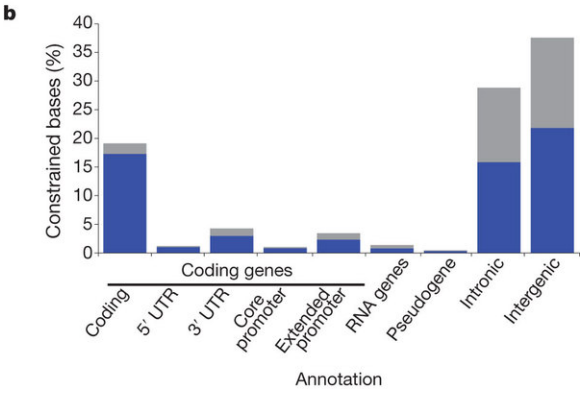
6. Iterative refinement; for many rounds, do:
   - *Randomized Partitioning:* Split sequences in M in two subsets by flipping a coin for each sequence and realign the two resulting profiles

Mammalian alignments





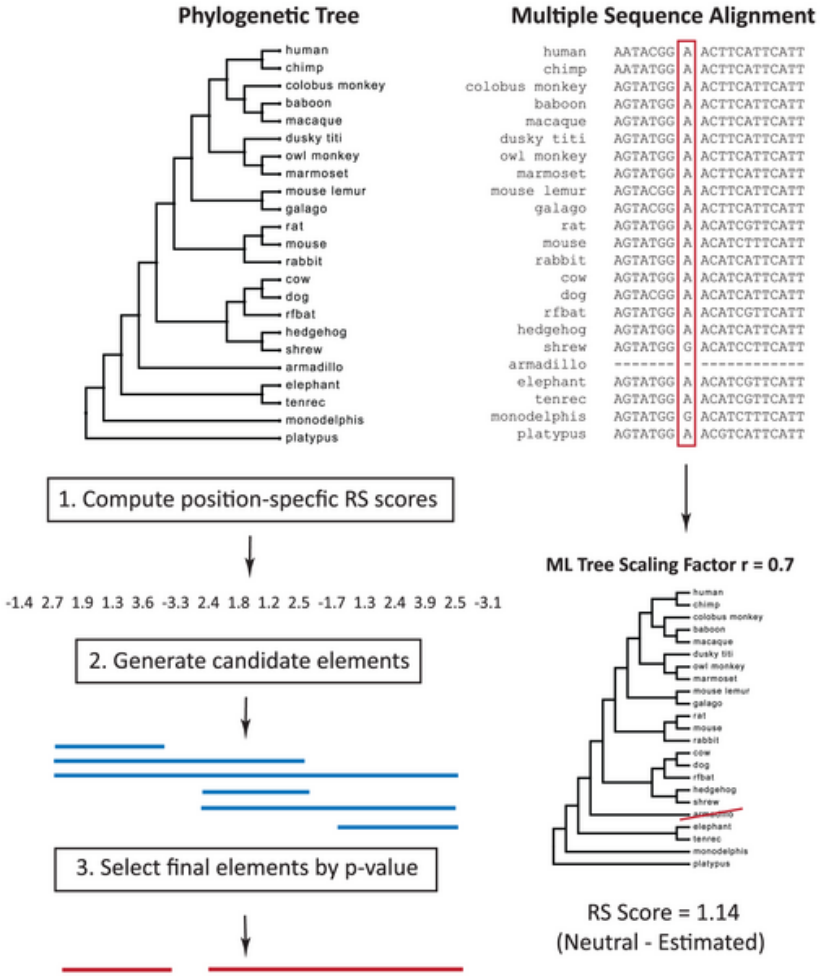


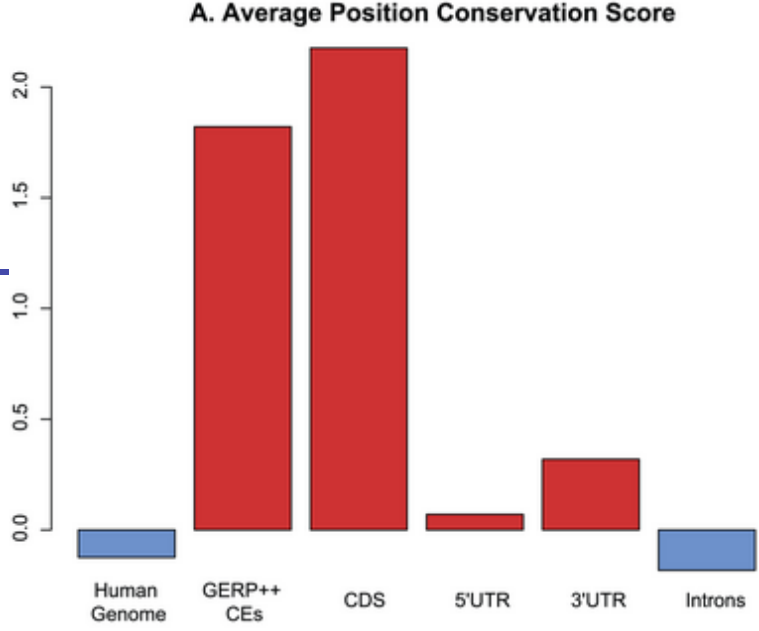


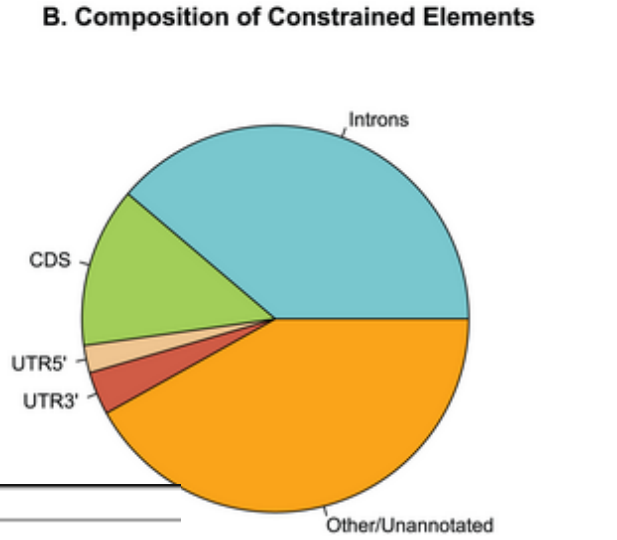
References

- Lindblad-Toh et al. Nature 478:476-482, 2011
- Lin et al. Genome Research 21:1916-1928, 2011

Genome Evolutionary Rate Profiling (GERP)







Annotation	% Coverage by CEs
Exons	84.6%
Introns	6.9%
UTR5'	23.7%
UTR3'	33.9%
ncRNA	10.1%

doi:10.1371/journal.pcbi.1001025.t001