Comparative Gene Finding

B. Majoros

Duke
UNIVERSITY

What is Comparative Gene Finding?

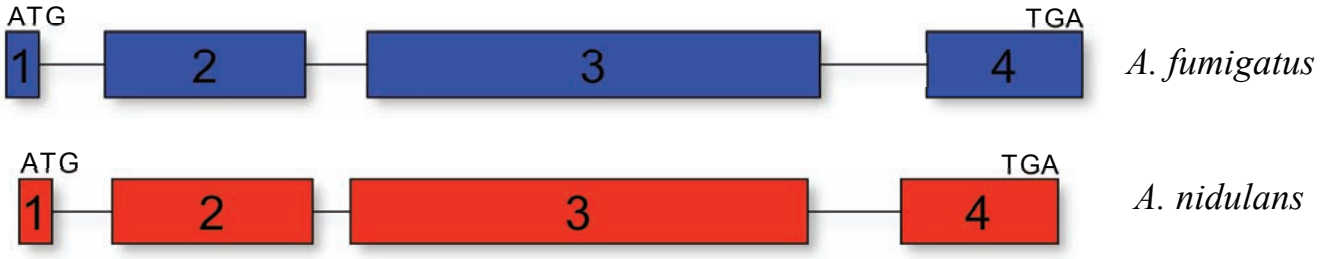
Problem: Predict genes in a *target genome* S based on the contents of S and also based on the contents of one or more *informant genomes* $I^{(1)}$... $I^{(n)}$:

```
S:   AAGGGAAGACAGGTGAGGGTCAAGCCCCAGCAAGTGCACCCAG----------ACACC
I1:  AAGGGAAGACAGGTGAGGGTCAAGCCCCAGCAAGTGCACCCAG----------ACACC
I2:  AAGGGAAGACATTTACGAGTCAAGCCACAGAAAGAGCCCTGAG----------GTGCC
I3:  AAAGGAGGACATGTGAGGGCCAAACTACTGAAGGTTCAACCAGG----------ATGCT
I4:  AAGGGGAGACAGGGGAGGGTCACACCATGGCAGAGG--CCAAG----------ACAGC
I5:  AAAGGAAACAATGGGAAGGTTA-TCAACTCCAAGTATGCCCAAGATCAAGGGAACCCCTT
I6:  AAAGGAAACCACTGGGAGGTTA-GAAATCACAGGTGCACCCAAGATCAAGGAA--CCCCT
```

Rationale: *Natural selection should operate more strongly on protein-coding DNA than on the non-functional "junk DNA" between genes. Intervals of strongly conserved DNA should therefore be more likely to contain "functional" elements.*

Duke UNIVERSITY

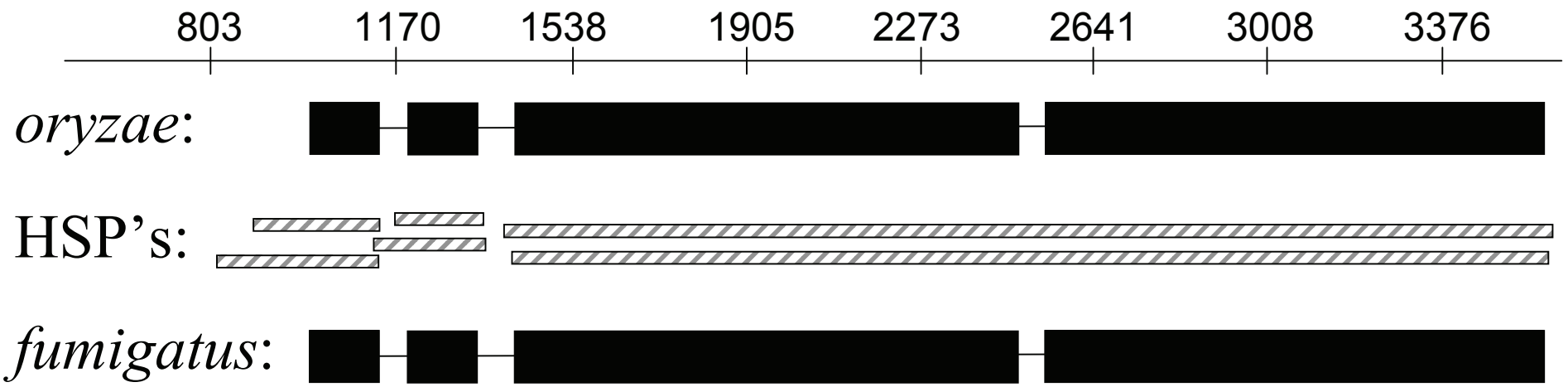
How Does Conservation Help?



feature	amino acid alignment score	<,>	nucleotide alignment score
exon 1	100%	>	71%
intron 1	14%	<	51%
exon 2	98%	>	85%
intron 2	29%	<	49%
exon 3	97%	>	82%
intron 3	9%	<	49%
exon 4	96%	>	83%

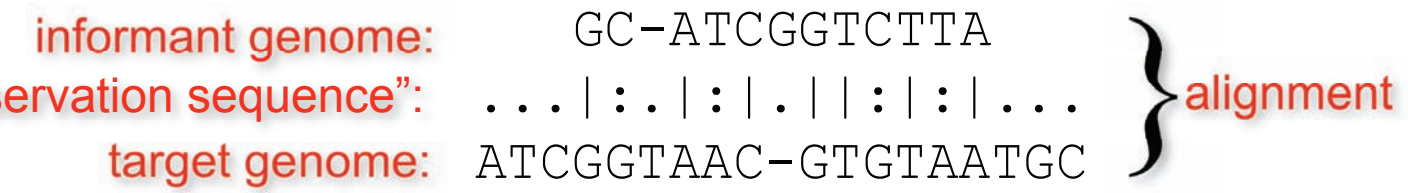
The Utility of Amino Acid Alignments

Amino acid alignments, such as those provided by the *PROmer* program (part of the popular *MUMmer* package) can be extremely informative in identifying conserved coding regions of genes:



Although the HSP boundaries (shown in several frames) generally do not coincide precisely with the edges of coding segments, they can be highly informative when combined with other information such as the scores from signal sensors and feature length distributions.

The TWINSCAN Approach



$$\phi^* = \arg\max_{\phi} P(\phi \mid S, C)$$

$$= \arg\max_{\phi} \frac{P(\phi, S, C)}{P(S, C)}$$

$$= \arg\max_{\phi} P(\phi, S, C)$$

$$= \arg\max_{\phi} P(\phi) P(S, C \mid \phi)$$

$$= \arg\max_{\phi} P(\phi) P(S \mid \phi) P(C \mid S, \phi)$$

$$\approx \arg\max_{\phi} P(\phi) P(S \mid \phi) P(C \mid \phi)$$

Find the parse ϕ which is most probable, given the target sequence S and the "*conservation sequence*" C (which encodes information about the informant).

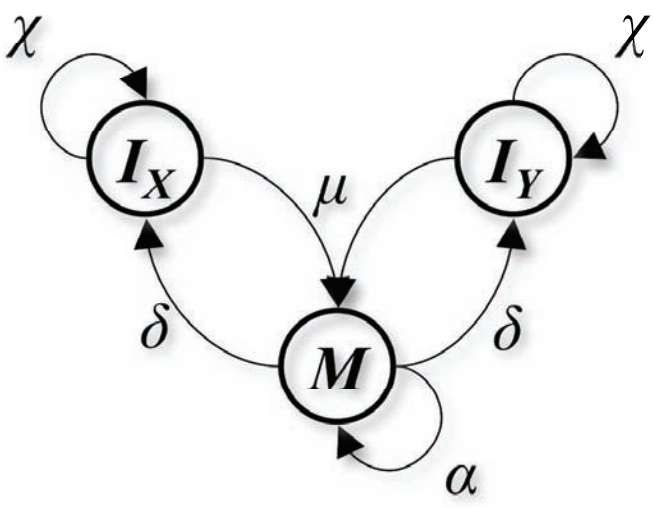
The $P(C|\phi)$ term is decomposed into a state-specific function that assesses whether the patterns of conservation in any given interval best match a coding state or a non-coding state. These functions are defined by 5th-order Markov chains on the alphabet of *matches* (|), *mismatches* (:), and *unaligned* positions (.)

(Korf *et al.*, 2001)

Duke UNIVERSITY

Pair HMM's (PHMM's)

A *Pair HMM* is an HMM which has *two output channels* rather than one; each state can emit a symbol into one or the other (or both) channels.



An example PHMM.

I_X : emit a symbol into output channel X

I_Y : emit a symbol into output channel Y

M : emit a symbol into both X and Y

I_X can be called an *insertion state*

I_Y can be called a *deletion state*

M can be called a *match state*

This is a simple PHMM used for *pairwise alignment*; more general PHMM's can have many more states, but those states can all be classified as *insertion states*, *deletion states*, or *match states*.

Duke UNIVERSITY

Pair HMM's (PHMM's)

Formally, we define a PHMM for DNA sequences as a 7-tuple:

$$M=(q^0,Q_M,Q_I,Q_D,\alpha,P_t,P_e),$$

for state set $Q=Q_M\cup Q_I\cup Q_D\cup\{q^0\}$, DNA alphabet α, transition distribution $P_t:Q\times Q\mapsto\mathbb{R}$, silent initial/final state q^0, and emission distribution $P_e:Q\times\alpha_+\times\alpha_+\mapsto\mathbb{R}$, for the augmented alphabet $\alpha_+=\{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T},-\}$. As before, all emission probabilities in q^0 are 0. All the state sets Q_M, Q_I, Q_D, and $\{q^0\}$ are disjoint. States in Q_M are referred to as *match states*, and are subject to:

$$P_e(q\in Q_M,s\in\alpha_+,-)=P_e(q\in Q_M,-,s\in\alpha_+)=0,$$

though it should be clear that these so-called "match" states can emit non-matching pairs of symbols as well as matching pairs. States in Q_I are known as *insertion states*, and satisfy:

$$P_e(q\in Q_I,s\in\alpha_+,s\in\alpha)=P_e(q\in Q_I,-,-)=0,$$

while states in Q_D are known as *deletion states*, and satisfy

$$P_e(q\in Q_D,s\in\alpha,s\in\alpha_+)=P_e(q\in Q_D,-,-)=0,$$

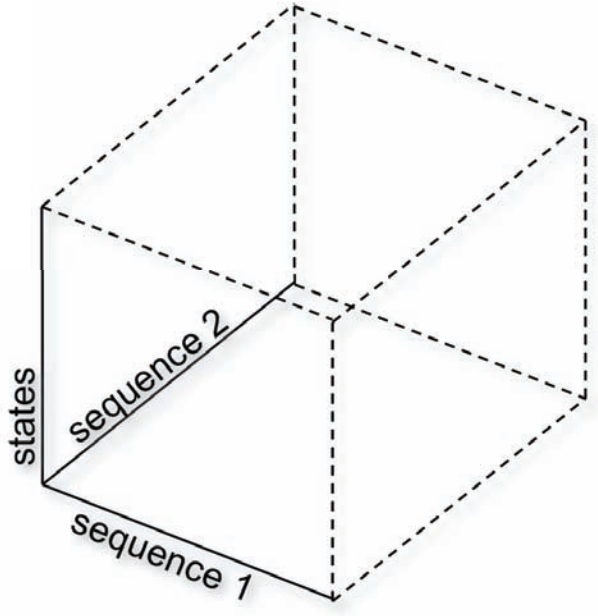
so that insertion states can emit only pairs in $\alpha\times\{-\}$ while deletion states emit only pairs in $\{-\}\times\alpha$.

Decoding for PHMM's

$$\phi^* = \underset{\phi=\{y_0,\ldots,y_{m-1}\}}{\arg\max}\ P_t(q^0 \mid y_{m-2}) \prod_{i=1}^{m-2} P_e(a_{i,1}, a_{i,2} \mid y_i) P_t(y_i \mid y_{i-1})$$

$$V_{i,j,k} = \begin{cases} \max_h\ V_{i-1,j-1,h} P_t(q_k \mid q_h) P_e(s_{i-1,1}, s_{j-1,2} \mid q_k) & \text{for } q_k \in Q_M \\ \max_h\ V_{i-1,j,h} P_t(q_k \mid q_h) P_e(s_{i-1,1}, - \mid q_k) & \text{for } q_k \in Q_I \\ \max_h\ V_{i,j-1,h} P_t(q_k \mid q_h) P_e(-, s_{j-1,2} \mid q_k) & \text{for } q_k \in Q_D \\ 0 & \text{for } q_k = q^0 \end{cases}$$

$$T_{i,j,k} = \begin{cases} \underset{(i-1,j-1,h)}{\arg\max}\ V_{i-1,j-1,h} P_t(q_k \mid q_h) P_e(s_{i-1,1}, s_{j-1,2} \mid q_k) & \text{for } q_k \in Q_M \\ \underset{(i-1,j,h)}{\arg\max}\ V_{i-1,j,h} P_t(q_k \mid q_h) P_e(s_{i-1,1}, - \mid q_k) & \text{for } q_k \in Q_I \\ \underset{(i,j-1,h)}{\arg\max}\ V_{i,j-1,h} P_t(q_k \mid q_h) P_e(-, s_{j-1,2} \mid q_k) & \text{for } q_k \in Q_D \end{cases}$$



$$V_{0,0,k} = \begin{cases} 1 & \text{for } q_k = q^0 \\ 0 & \text{otherwise} \end{cases}$$

$$V_{i>0,0,k} = \begin{cases} \max_h\ V_{i-1,0,h} P_t(q_k \mid q_h) P_e(s_{i-1,1}, - \mid q_k) & \text{for } q_k \in Q_I \\ 0 & \text{otherwise} \end{cases}$$

$$V_{0,j>0,k} = \begin{cases} \max_h\ V_{0,j-1,h} P_t(q_k \mid q_h) P_e(-, s_{j-1,2} \mid q_k) & \text{for } q_k \in Q_D \\ 0 & \text{otherwise} \end{cases}$$

$$T_{i>0,0,k} = \begin{cases} \underset{(i-1,0,h)}{\arg\max}\ V_{i-1,0,h} P_t(q_k \mid q_h) P_e(s_{i-1,1}, - \mid q_k) & \text{for } q_k \in Q_I \\ NIL & \text{otherwise} \end{cases}$$

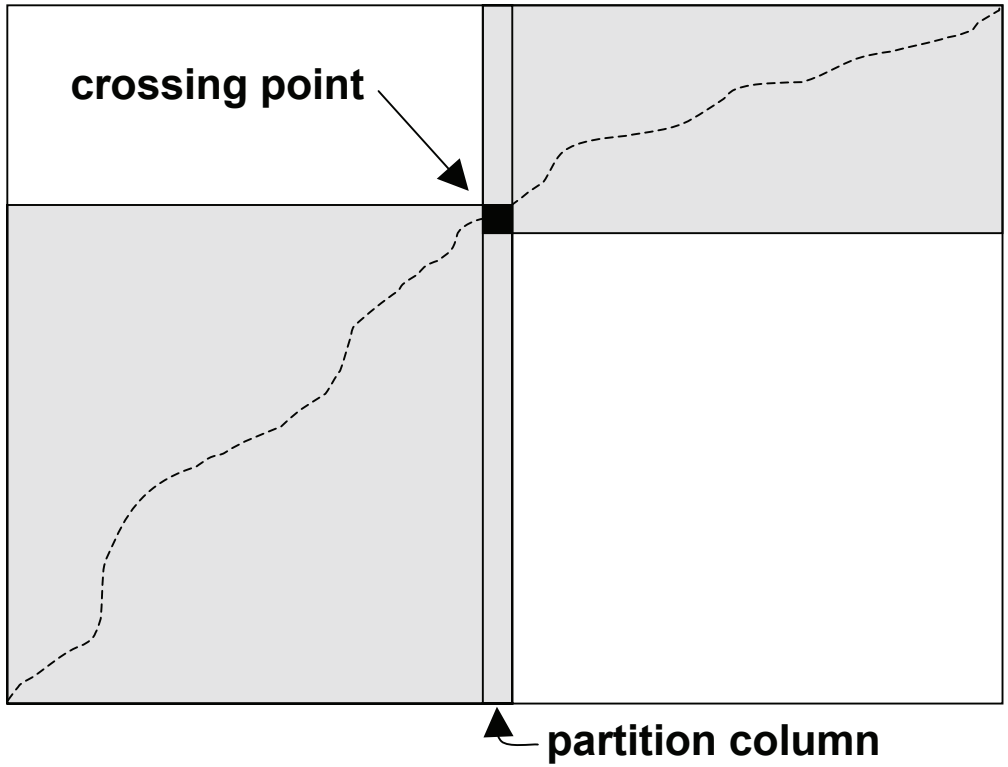
$$T_{0,j>0,k} = \begin{cases} \underset{(0,j-1,h)}{\arg\max}\ V_{0,j-1,h} P_t(q_k \mid q_h) P_e(-, s_{j-1,2} \mid q_k) & \text{for } q_k \in Q_D \\ NIL & \text{otherwise} \end{cases}$$

$$T_{0,0,k} = NIL$$

Duke UNIVERSITY

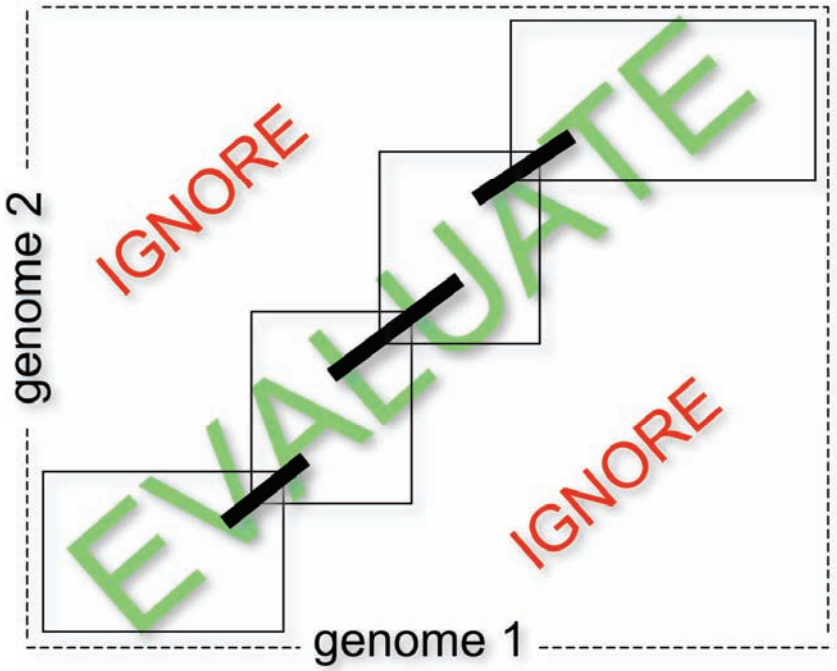
The Hirschberg Algorithm

The *Hirschberg algorithm* (Hirschberg, 1975) reduces the space requirements of a standard alignment algorithm from $\mathbb{O}(n^2)$ to $\mathbb{O}(n)$ while leaving the time complexity $\mathbb{O}(n^2)$, via a recursive procedure in which a decoding pass is made over the two halves of the matrix to determine the *crossing point* of the optimal path through the *partition column*. The matrix is then partitioned in half at the crossing point. The two remaining subproblems (gray areas in the figure) are then recursively solved in the same manner. The extension of this algorithm for use in PHMM decoding obviously requires that the partition column be generalized to a column-like volume in the 3D decoding matrix.



Pruning the Search Space for PHMM's

Evaluating the full dynamic programming matrix can be impractical for long sequences; *pruning* (or *banding*) is a common alternative.



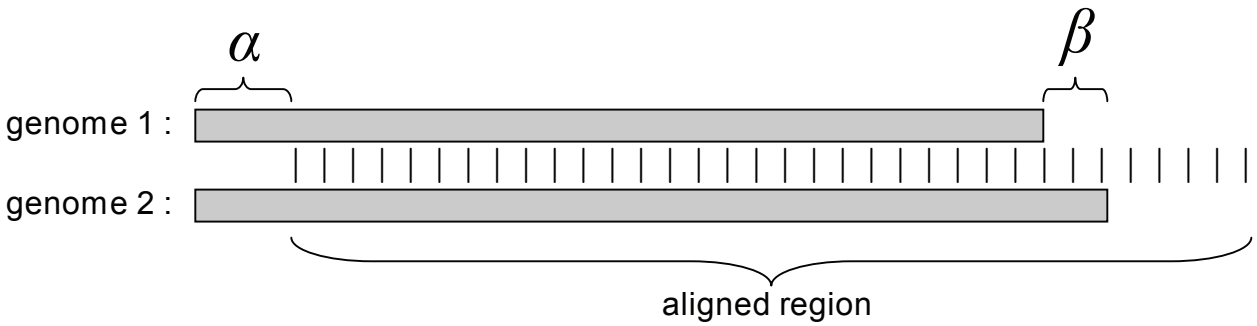
- Find significantly conserved regions (thick bars) using BLAST
- Force the DP algorithm to select a path which passes through these regions
- Allow more flexibility in the regions not aligned
- Do not evaluate regions of the matrix far from the conserved regions

Duke UNIVERSITY

Prediction on a Pairwise Alignment

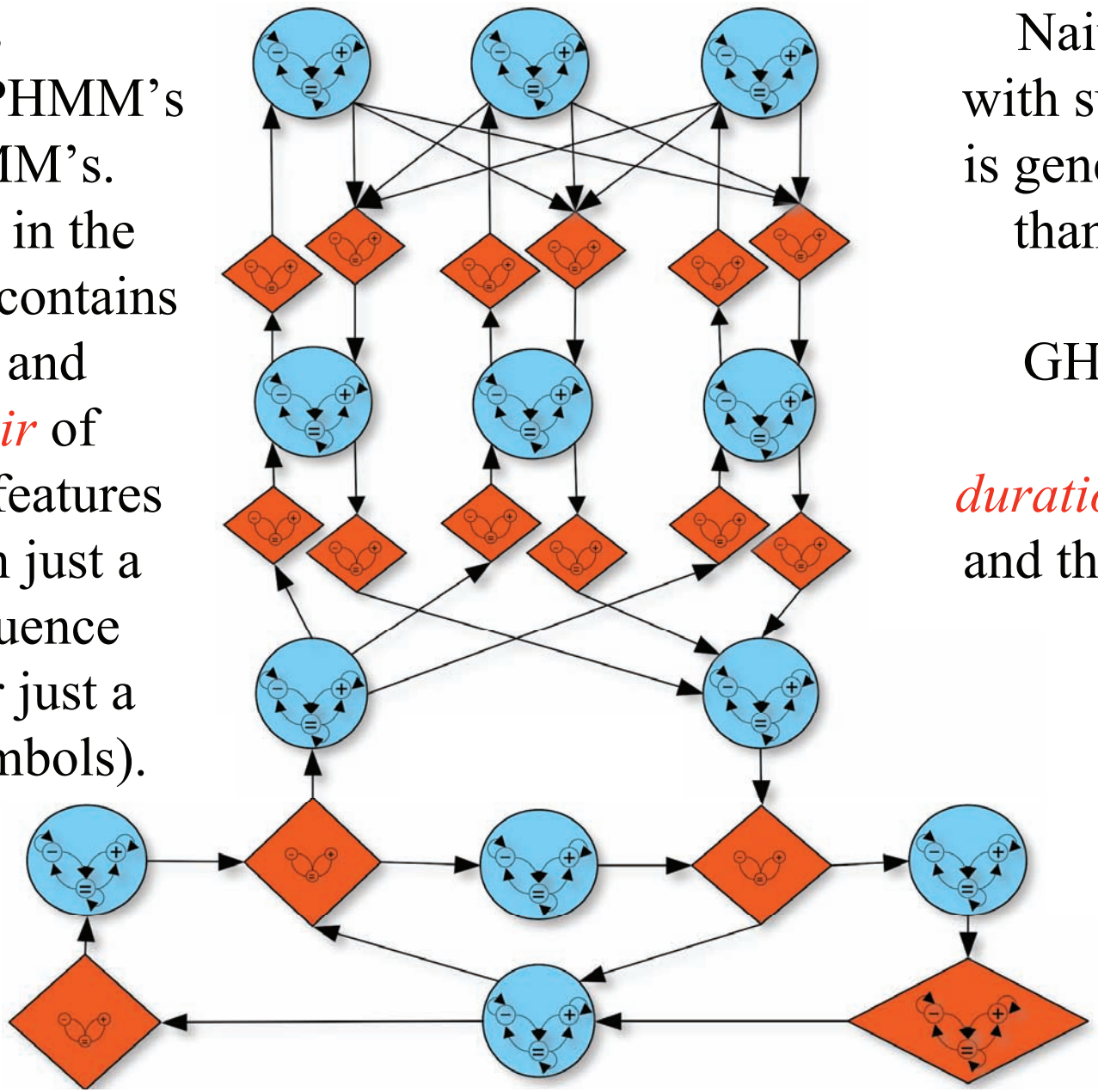
Given an alignment between two genomes, we can perform gene prediction on one of the genomes using existing single-genome techniques, but meanwhile adjust the scoring of putative features based on how well they are conserved in the informant genome. This is much faster than a PHMM since *the alignment is pre-computed*.

To account for evolutionary divergence in feature lengths as well as the imperfect nature of pre-computed alignments, we can allow some "*fuzziness*" at the ends of paired features in the two genomes:



Pair GHMM's (PGHMM's)

GPHMMs combine PHMM's with GHMM's. Each state in the GPHMM contains a PHMM, and emits a *pair* of sequence features rather than just a single sequence feature (or just a pair of symbols).



Naive decoding with such a model is generally worse than for either a PHMM or a GHMM, due to the *explicit duration modeling* and the size of the DP matrix.

Duke UNIVERSITY

Recall: GHMM Decoding

Finding the optimal parse, $\phi_{\max}$:

$$\phi_{\max} = \arg\max_{\phi} P(\phi \mid S) = \arg\max_{\phi} \frac{P(\phi, S)}{P(S)}$$

$$= \arg\max_{\phi} P(\phi, S) = \arg\max_{\phi} P(S \mid \phi) P(\phi)$$

$$= \arg\max_{\phi} \prod_{i=1}^{n-1} P_e(S_i \mid q_i, d_i) P_t(q_i \mid q_{i-1}) P_d(d_i \mid q_i)$$

=emission =transition =duration

Decoding with a GPHMM

$$\phi^* = P_t(q^0 \mid q_{n-2}) \underset{\phi}{\arg\max} \prod_{i=1}^{n-2} P_e(S_{i,1}, S_{i,2} \mid q_i, d_{i,1}, d_{i,2})$$

$$\underbrace{P_t(q_i \mid q_{i-1})}_{\text{=transition}} P_d(d_{i,1}, d_{i,2} \mid q_i)$$

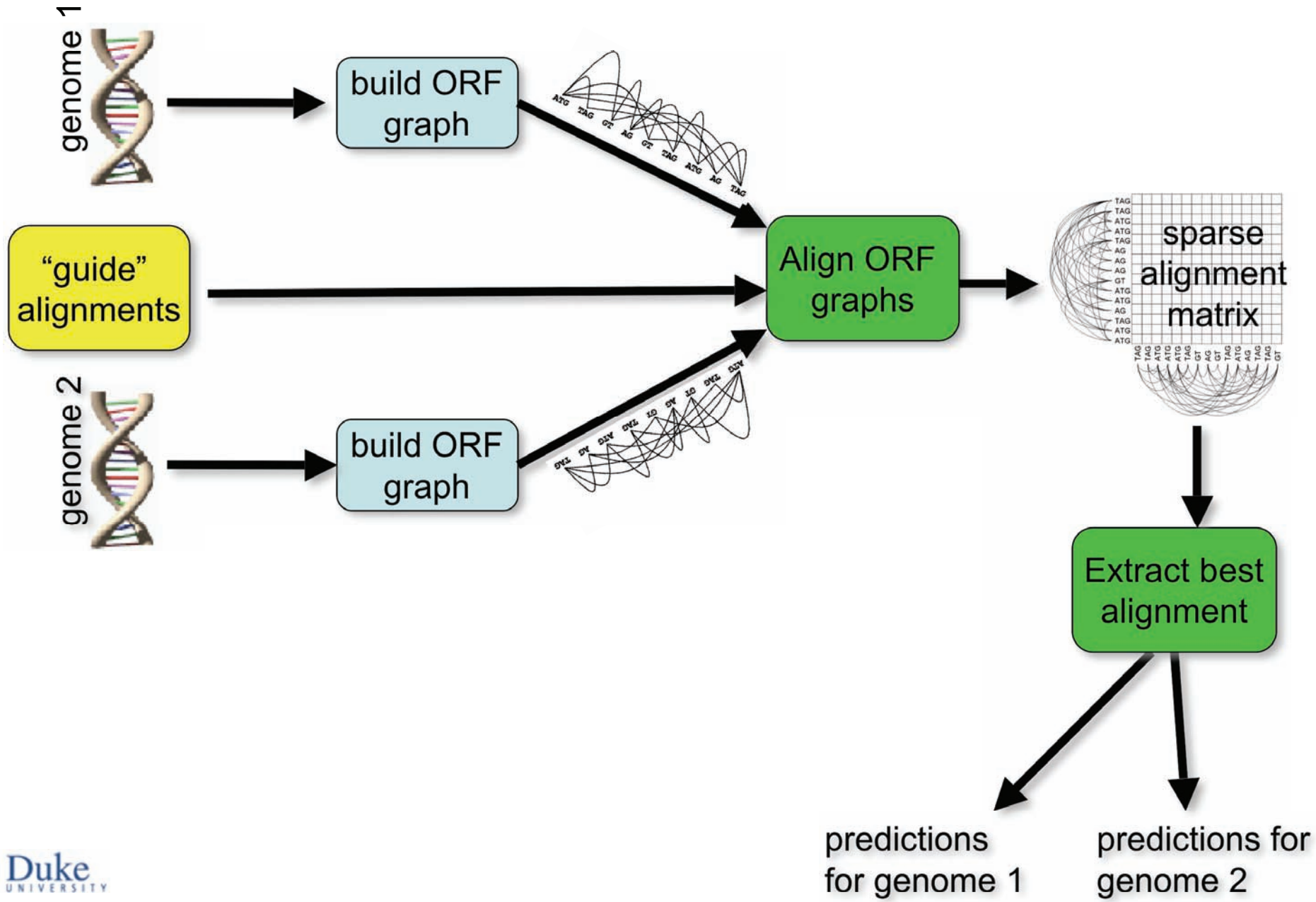
$$P_e(S_{i,1}, S_{i,2} \mid q_i, d_{i,1}, d_{i,2}) \approx \underbrace{P_e(S_{i,1} \mid q_i, d_{i,1})}_{\text{=single-genome emission score}} \underbrace{P_e(S_{i,2} \mid S_{i,1}, q_i, d_{i,2})}_{\text{≈alignment score}}$$

$$P_d(d_{i,1}, d_{i,2} \mid q_i) = P(d_{i,1} \mid q_i) P(d_{i,2} \mid d_{i,1}, q_i)$$

$$\approx \underbrace{P(d_{i,1} \mid q_i)}_{\text{=single-genome duration score}} \underbrace{P(\Delta_d \mid q_i)}_{\text{ignore (implicit in alignment score)}}, \quad \Delta_d = d_{i,2} - d_{i,1}$$

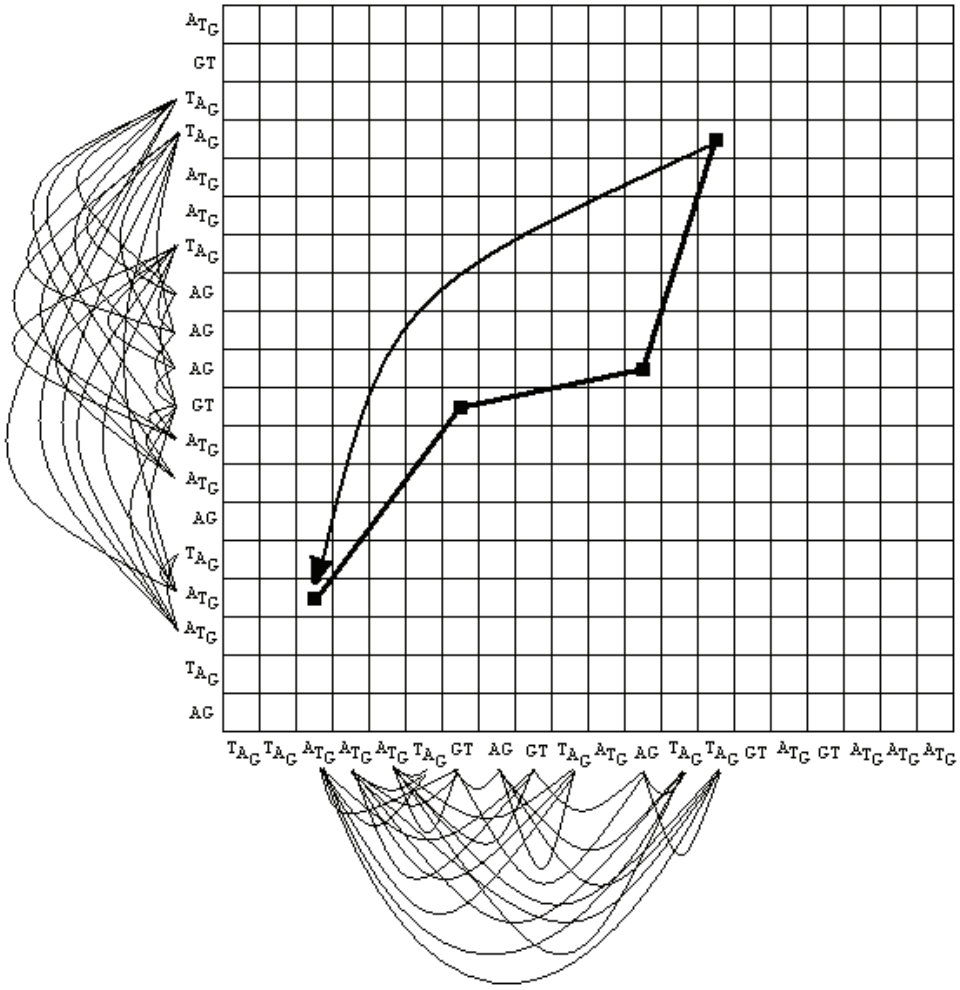
Duke UNIVERSITY

Practical GPHMM Decoding



Aligning ORF Graphs

The two ORF graphs can be aligned using a global alignment algorithm. The optimal alignment corresponds to the chosen pair of orthologous gene predictions.



The alignment is constrained by the topologies of the two ORF graphs:

1.Only like signals can align,

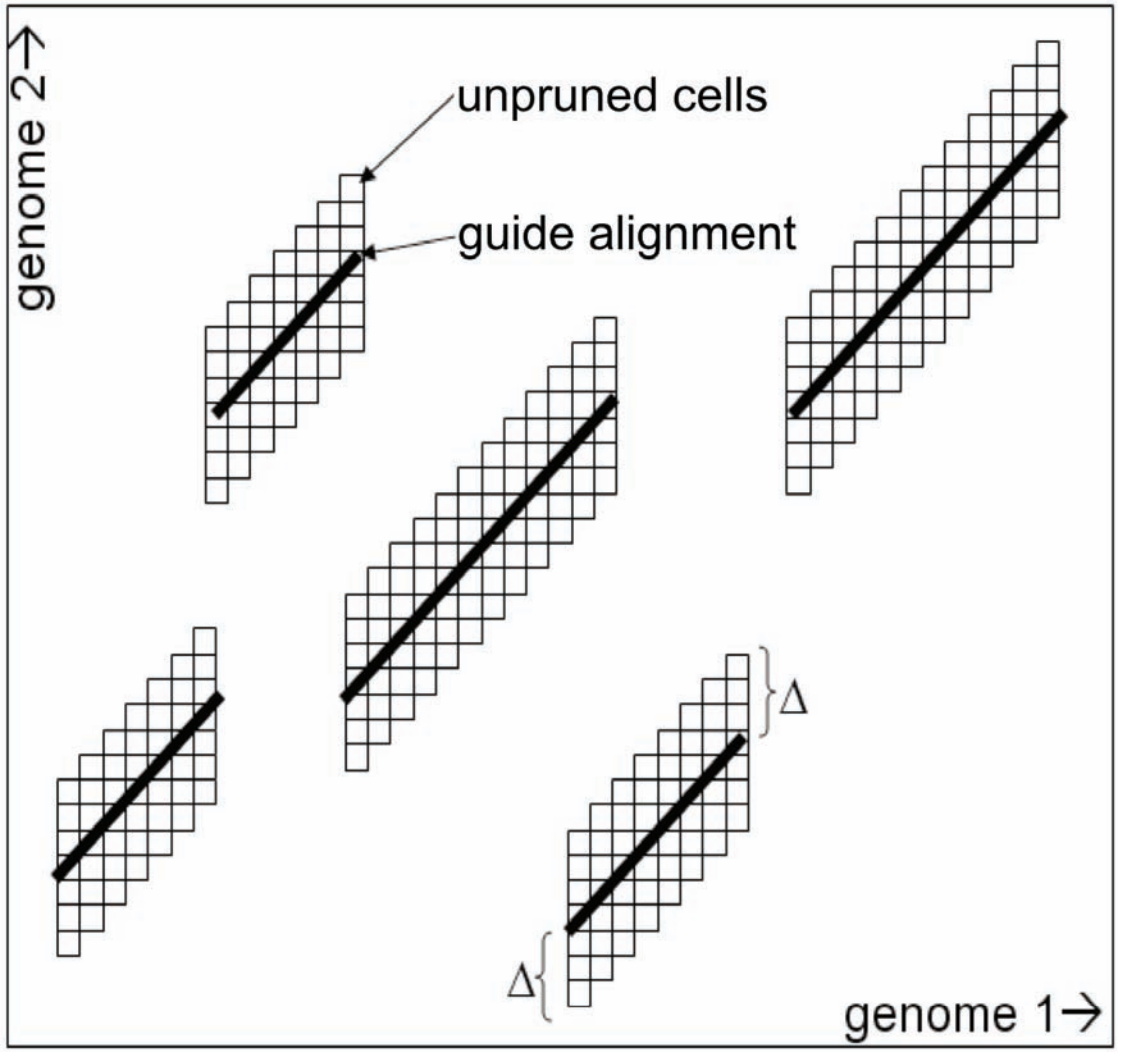
2.Two signals can align only if they have neighbors which also align,

3.Standard phase constraints apply.

Duke UNIVERSITY

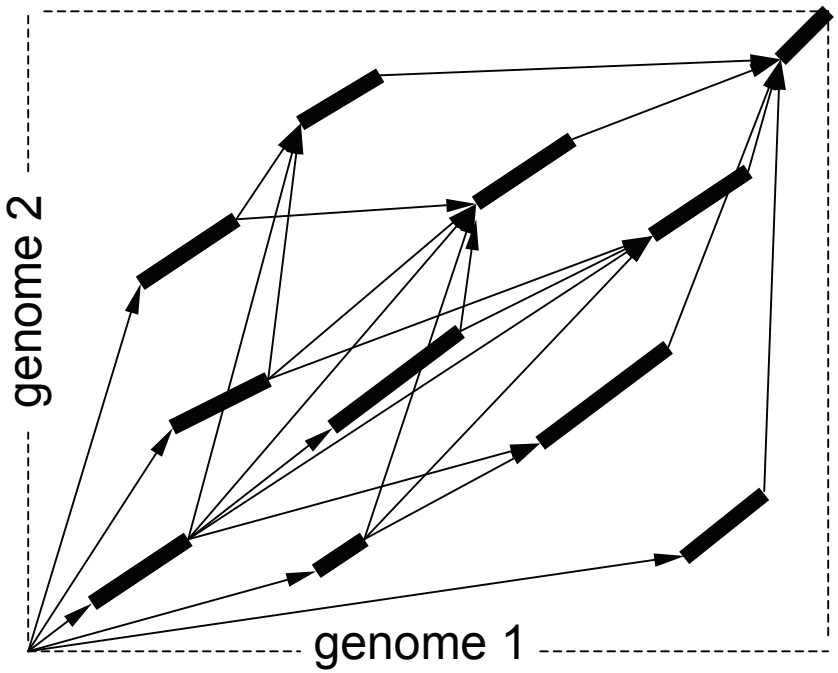
Using a Sparse Alignment Matrix

Superimposing *guide alignments* (precomputed using *BLAST* or *MUMMER*) onto the DP matrix allows us to prune the matrix:



HSP Graphs

Guide alignments may be combined into a *partial ordering* such that the end of one HSP falls strictly *before* (in both axes of the DP matrix) the beginning coordinates of its successor in the partial ordering:



Such a graph is similar to a *Steiner graph*, and may be used to search for an optimal tiling across the DP matrix.

Computing Approximate Alignment Scores

Alignment scores based on the guide alignments have to account for missing information -- i.e., the *guide alignments generally do not form a complete tiling across the matrix*, so that portions of the optimal DP path do not overlap any guide alignment.

Rapid evaluation of approximate alignment scores can be achieved by precomputing *prefix-sum arrays* for the guide alignments; numbers of matches and alignment lengths for portions of a guide alignment can then be computed in constant time via simple subtraction:

$$P_{ident} = \begin{cases} \dfrac{\mu_z - \mu_{y-1}}{\lambda_z - \lambda_{y-1} + \delta_A + \delta_B} & \text{if } \mu_z - \mu_{y-1} > 0 \\ 0 & \text{otherwise} \end{cases}$$

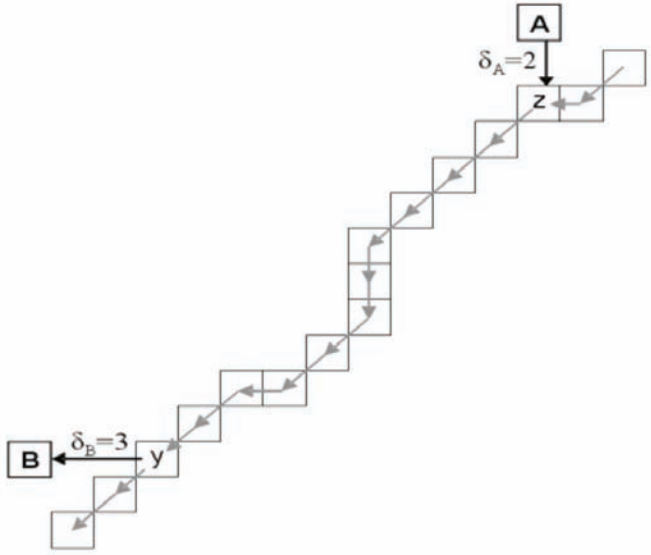


***Figure 9.15**: A crude method for computing approximate alignment scores. A guide alignment (MUMmer HSP) is shown extending through cells z and y. The OASIS cells A and B, between which an alignment and its score are desired, do not fall directly on the guide alignment, so the shortest path from each cell to the guide alignment is constructed. The alignment score between z and y is rapidly obtained using a prefix sum array (PSA), with additional indel penalties corresponding to distances δ_A and δ_B being incorporated into the approximate score.*

Duke UNIVERSITY

Accuracy: GPHMM versus GHMM

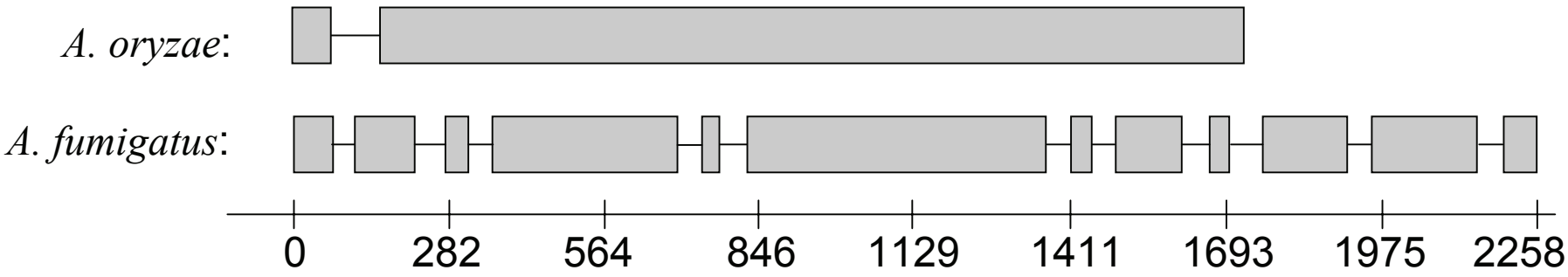
Data set: 147 high-confidence *Aspergillus fumigatus* × *A. nidulans* orthologs (493 exons, 564kb).

	nucleotide accuracy	exon sensitivity	exon specificity	exact genes
GHMM	99%	78%	73%	54%
GPHMM	99%	89%	85%	74%

(TWAIN: Majoros et al, 2004)

Gene Structure Evolution

Orthologous genes can sometimes differ radically in their intron-exon structure, due to *gene structure evolution*:

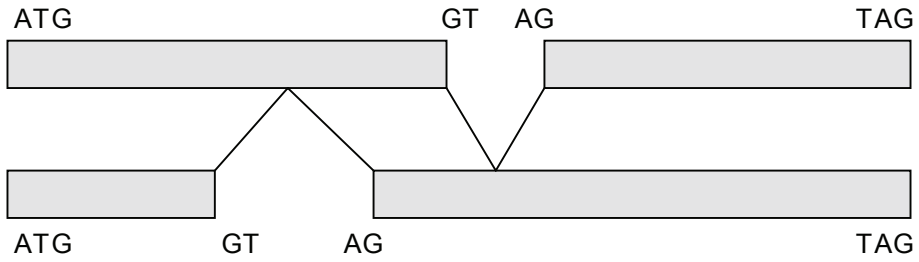


These two *Aspergillus* genes encode identical proteins!

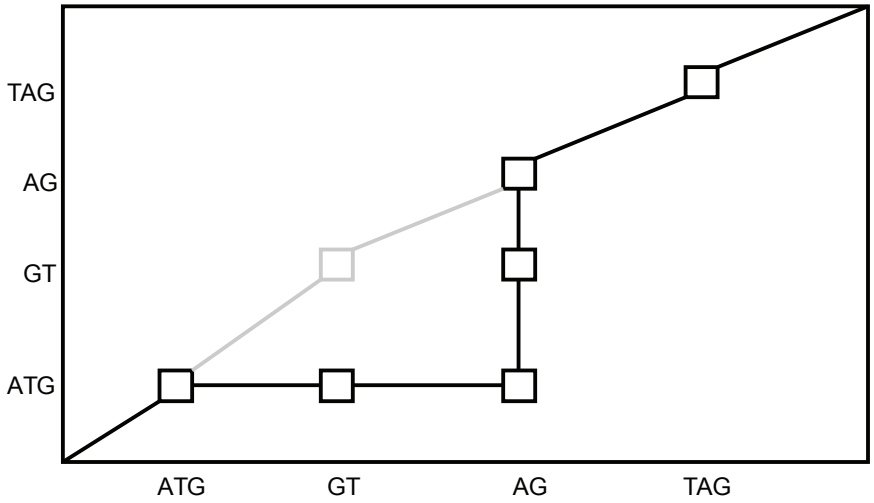
This appears to be more prevalent in some taxa than others; intron gain and loss in the mammals appears to be rare. Nevertheless, in other taxa it is more common, and must be addressed by gene prediction programs.

Modeling Gene Structure Evolution

Orthologous genes do not always have the same *intron-exon structure*, even if they do have the same numbers of exons:



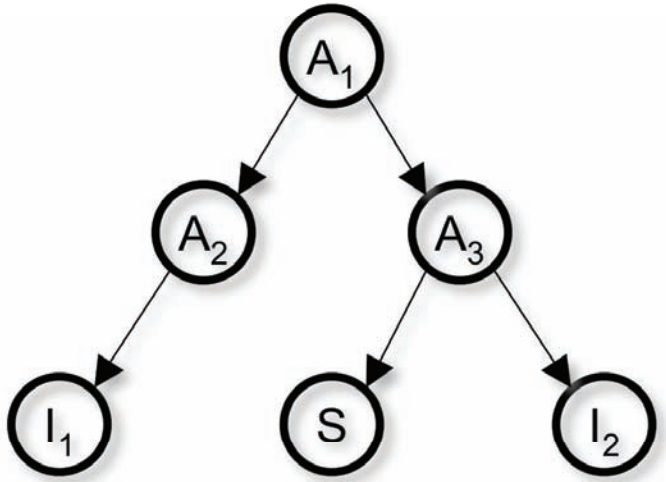
Changes in intron-exon structure are readily accommodated by the *alignment-of-ORF-graphs* representation, though doing so can severely limit the options for *pruning* of the alignment matrix, resulting in unacceptable run times:



Duke UNIVERSITY

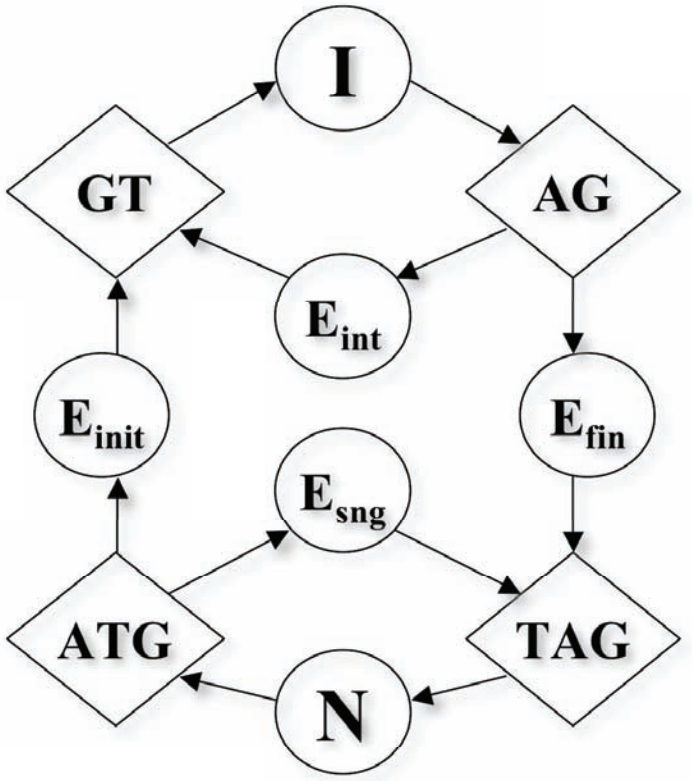
Phylogenomic HMM's (PhyloHMM's)

model of phylogeny



+

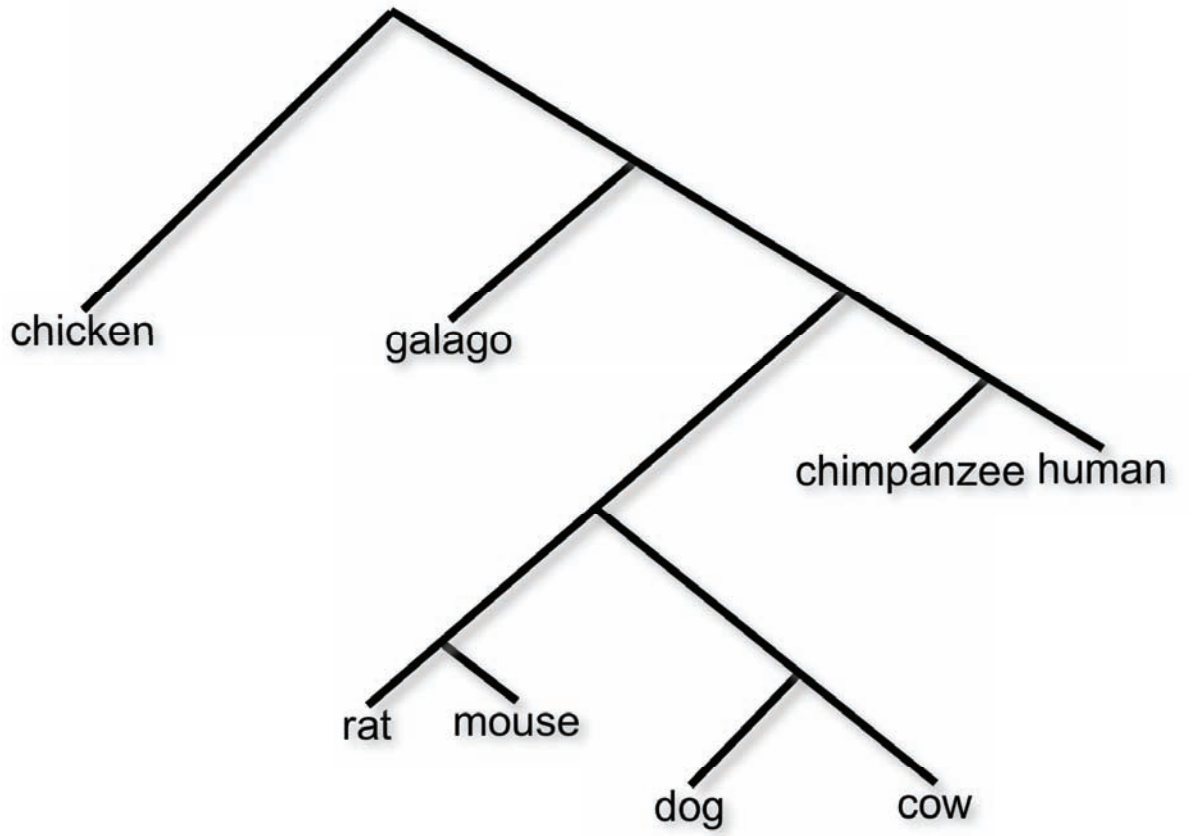
model of gene structure



= a model of gene structure informed by observed evolutionary divergence

Duke UNIVERSITY

Evolutionary Sequence Conservation



- Using multiple genomes increases effective sample size
- However, we have to control for the non-independence of informant genomes
- The "ideal evolutionary distance" for informant genomes usually is not known

```
human:   AAGGGAAGACAGGTGAGGGTCAAGCCCCAGCAAGTGCACCCAG----------ACACC
chimp:   AAGGGAAGACAGGTGAGGGTCAAGCCCCAGCAAGTGCACCCAG----------ACACC
cow:     AAGGGAAGACATTTACGAGTCAAGCCACAGAAAGAGCCCCTGAG----------GTGCC
dog:     AAAGGAGGACATGTGAGGGCCAAACTACTGAAGGTTCAACCAGG----------ATGCT
galago:  AAGGGGAGACAGGGGAGGGTCACACCATGGCAGAGG--CCAAG----------ACAGC
rat:     AAAGGAAACAATGGGAAGGTTA-TCAACTCCAAGTATGCCCAAGATCAAGGGAACCCCTT
mouse:   AAAGGAAACCACTGGGAGGTTA-GAAATCACAGGTGCACCCAAGATCAAGGAA--CCCCT
```

Decoding with a PhyloHMM

$$\phi^* = \arg\max_{\phi} P(\phi \mid S, I^{(1)}, \ldots, I^{(n)})$$

$$= \arg\max_{\phi} \frac{P(\phi, S, I^{(1)}, \ldots, I^{(n)})}{P(S, I^{(1)}, \ldots, I^{(n)})}$$

$$= \arg\max_{\phi} P(\phi, S, I^{(1)}, \ldots, I^{(n)})$$

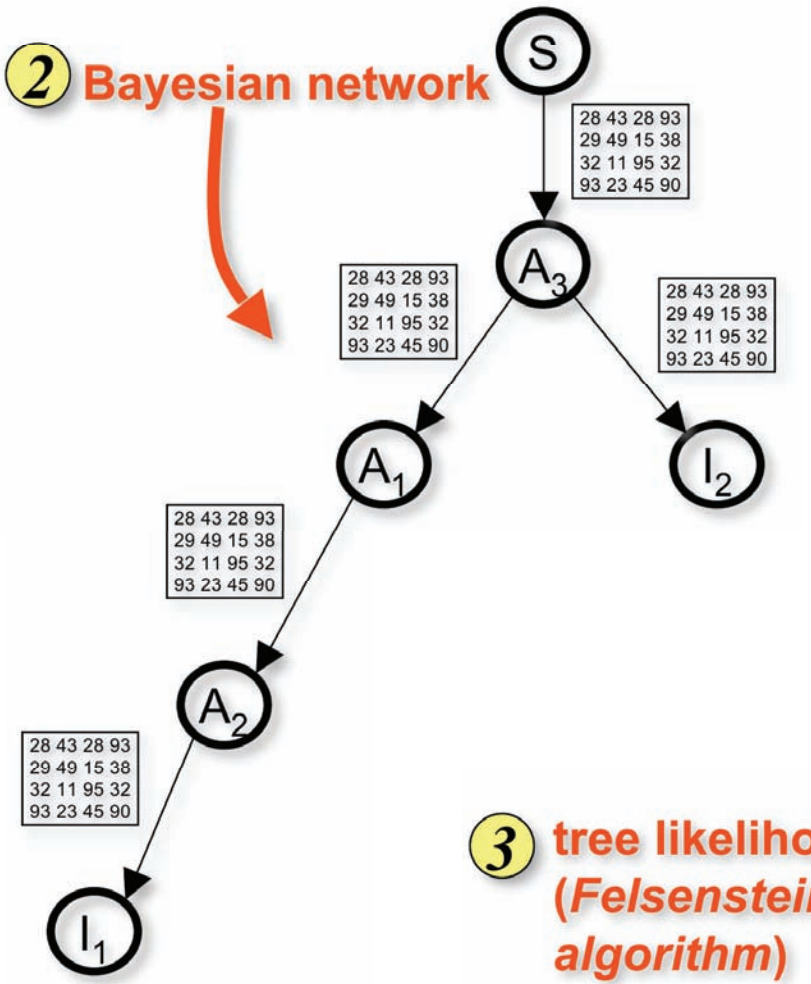
$$= \arg\max_{\phi} P(\phi) P(S, I^{(1)}, \ldots, I^{(n)} \mid \phi)$$

$$= \arg\max_{\phi} P(\phi) P(S \mid \phi) P(I^{(1)}, \ldots, I^{(n)} \mid S, \phi)$$

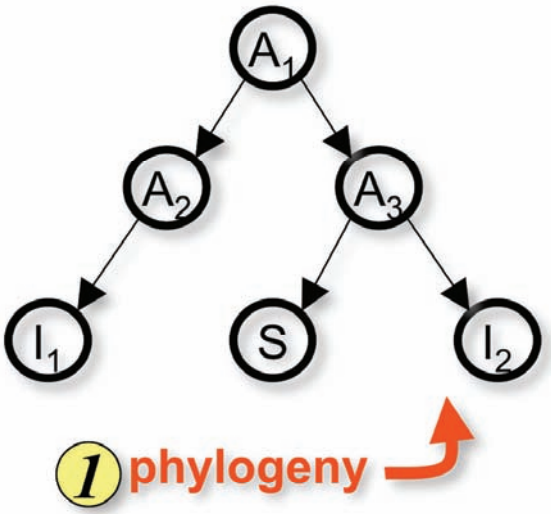
standard GHMM computation

tree likelihood (Felsenstein's algorithm)

Phylogenies as Bayesian Networks



re-root and attach rate matrices



$$P(I^{(1)},\ldots,I^{(n)} \mid S,\phi) =$$

3 tree likelihood (Felsenstein's algorithm)

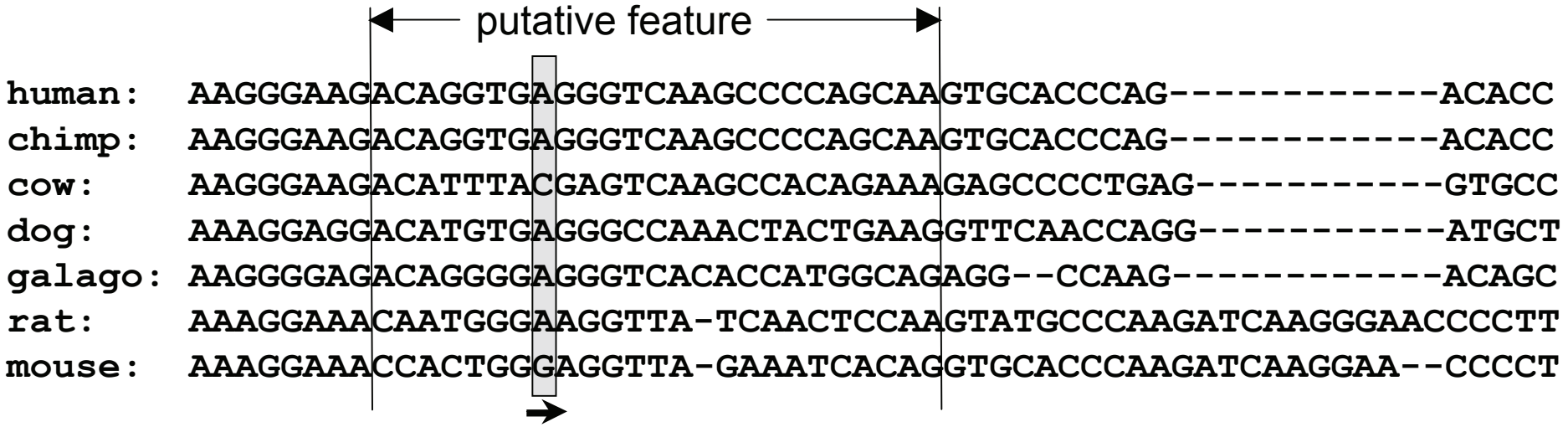
$$\sum_{\text{unobservables}} \left(\prod_{\substack{\text{nonroot} \\ v}} P(v \mid parent(v),\phi) \right)$$

$$= \sum_{A_1,A_2,A_3} P(I_1 \mid A_2)P(A_2 \mid A_1)P(A_1 \mid A_3)P(I_2 \mid A_3)P(A_3 \mid S)$$

Duke UNIVERSITY

Evaluating a Putative Feature

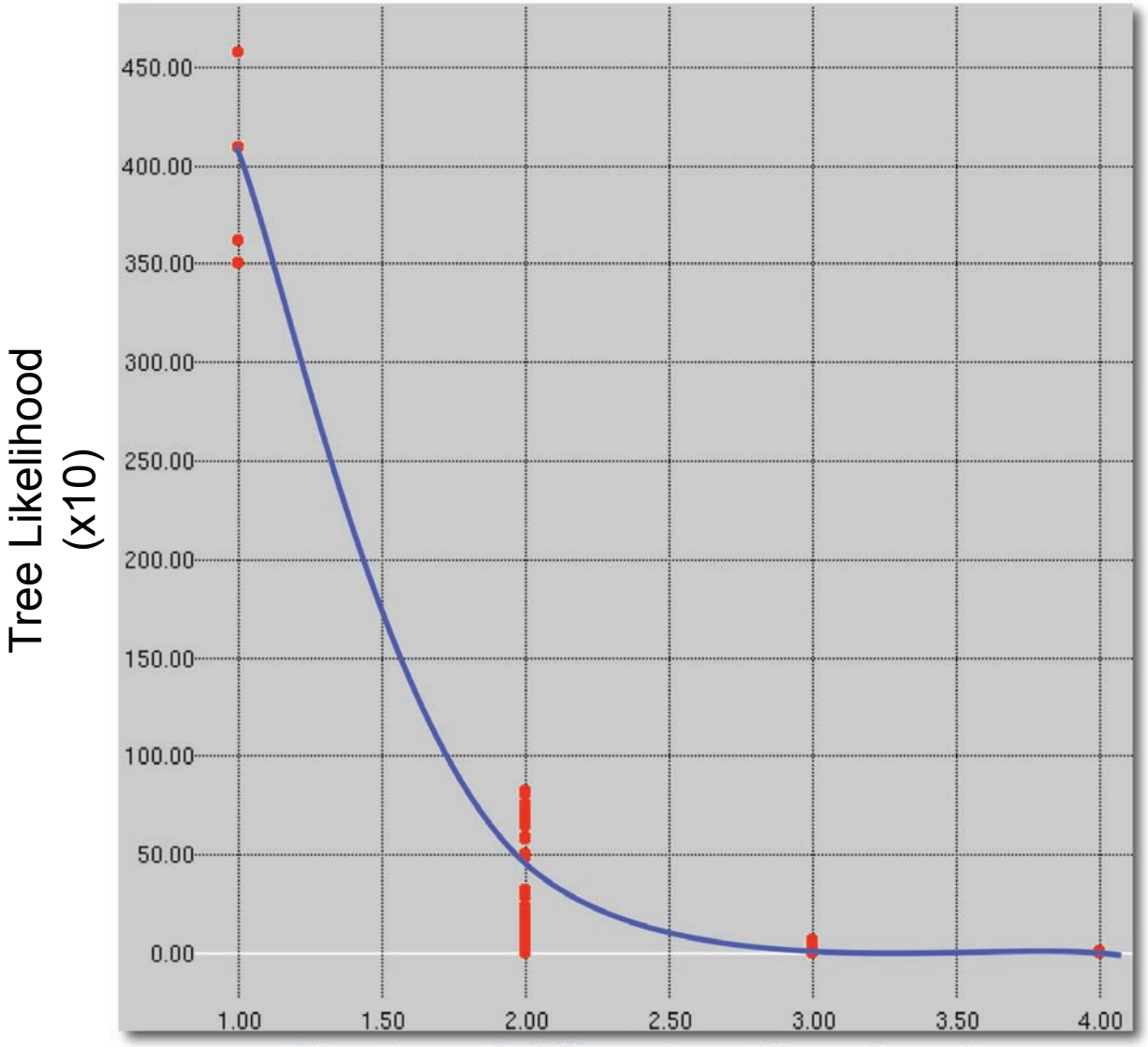
The tree likelihood for a single column can be combined across the columns of a putative feature (assuming independence) to evaluate the likelihood of that interval, *given the feature type*. Each feature type must therefore have its own *evolution model*.



Non-independence of columns can be accomodated by using *higher-order PhyloHMM's*, much like higher-order Markov chains, in which a *Markov assumption* is made regarding *conditional independence* of one column given some number of preceding columns. Higher-order models require more parameters, and more training effort.

Duke UNIVERSITY

Likelihood as a Function of Mutations



(five species aligned over 5000 columns)

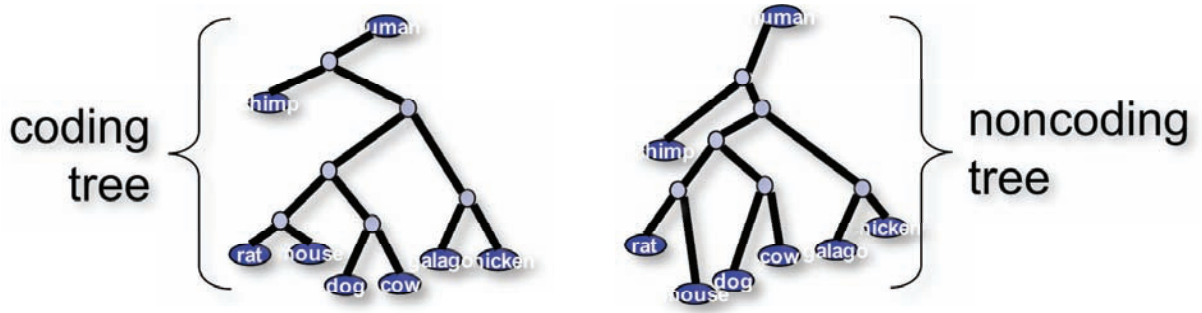
Duke UNIVERSITY

Modeling Evolutionary Change in Both Nucleotides and Amino Acids

Recall from the earlier slide:

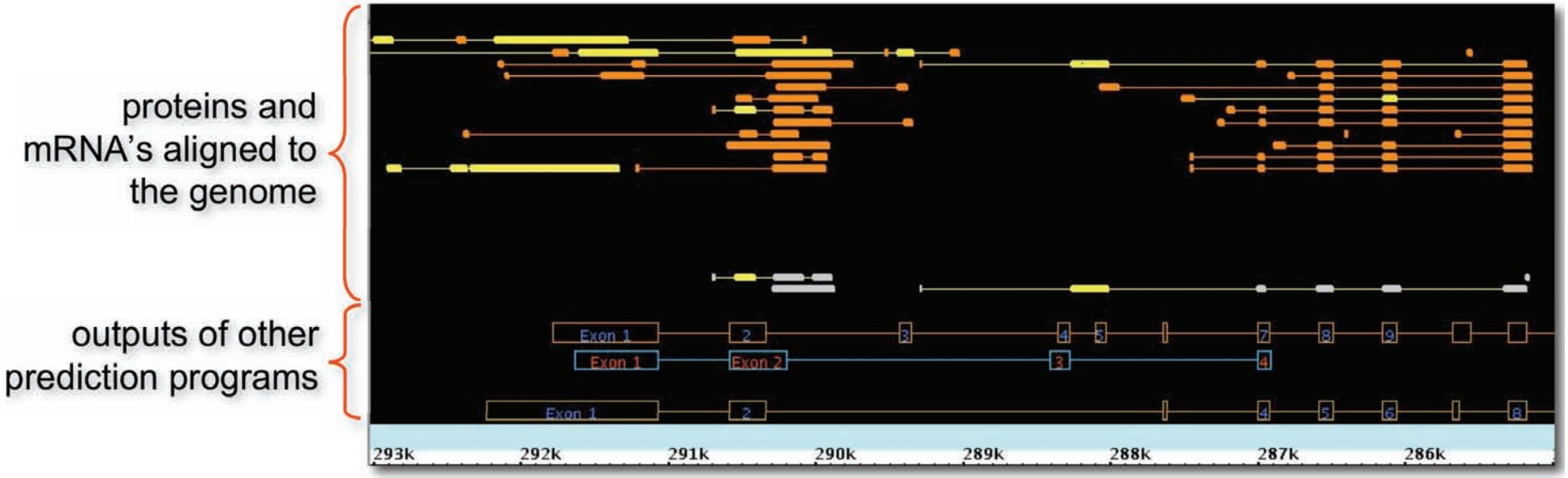
feature	amino acid alignment score	<,>	nucleotide alignment score
exon 1	100%	>	71%
intron 1	14%	<	51%
exon 2	98%	>	85%
intron 2	29%	<	49%
exon 3	97%	>	82%
intron 3	9%	<	49%
exon 4	96%	>	83%

Thus, we need to model <u>separately</u> the rate of evolutionary change in *coding regions* (ideally at the amino acid level) and *noncoding regions* (at the nucleotide level):



Duke UNIVERSITY

Expression-based Methods

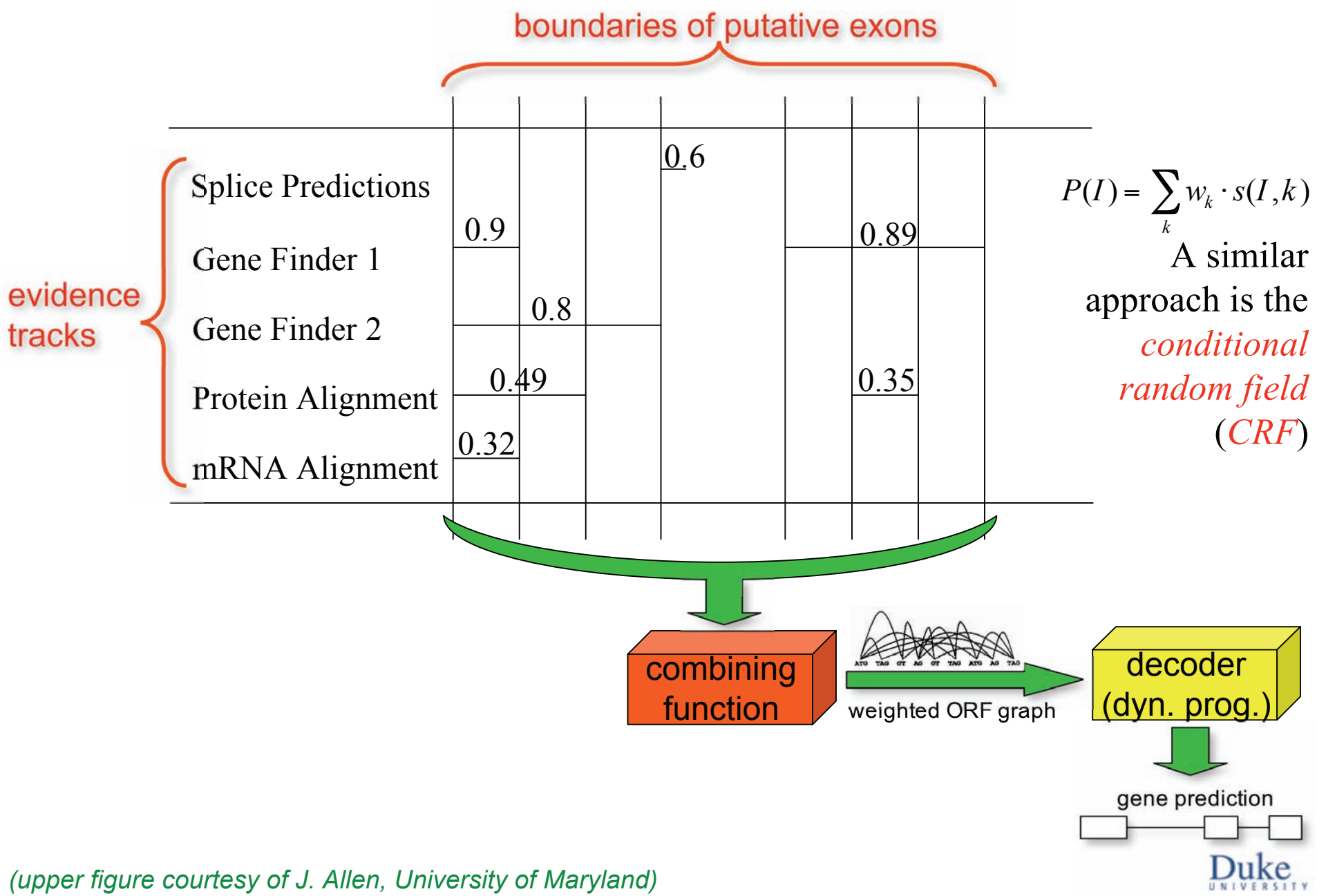


- Proteins and sequenced mRNA's can be aligned to the genome using dynamic programming algorithms.
- Various *ad hoc* methods have been explored for utilizing this information during gene prediction.
- Outputs from other gene finders can also be used as input to a "combiner" program.

(figure courtesy of J. Allen, University of Maryland)

Duke UNIVERSITY

Ad hoc "Combiner" Methods



$$P(I) = \sum_k w_k \cdot s(I,k)$$

A similar approach is the *conditional random field* (*CRF*)

(upper figure courtesy of J. Allen, University of Maryland)

Duke UNIVERSITY

CRF's : A Principled Alternative to Combiners

A *conditional random field* (*CRF*) is a 4-tuple F=(G,F,W,L) where:

- G=(V,E) is a graph describing a set of *conditional independence relations* E among variables V,
- F is a set of *scoring functions* over possible values for the variables in V,
- W is a set of *weights* for combining those functions, and
- L is a set of *labels*; e.g., {`exon`, `intron`}.

Given a CRF and a sequence S, the probability of a putative parse ϕ for the sequence is defined as:

$$P(\phi \mid S) = \frac{1}{Z(S)} e^{\sum_{i=0}^{L-1} \sum_{j \in J} \lambda_j f_j(y_{i-1} y_i, S, i)}$$

where Z (the *partition function*) is a summation over all possible parses. For (standard) decoding we need not evaluate Z at all:

$$\phi^* = \arg\max_{\phi} \sum_{i=0}^{L-1} \sum_{j \in J} \lambda_j f_j(y_{i-1} y_i, S, i)$$

It is interesting to note that the above formulation is similar to that of *simulated annealing*:

$$P(S_i) = \frac{e^{\frac{G_i}{kT}}}{\sum_j e^{\frac{G_j}{kT}}}$$

$k \approx 1.38065 \times 10^{-23}$ is the *Boltzmann constant*

Summary

- Comparative gene finding makes use of sequence *conservation patterns* between related taxa
- Conservation patterns reflect *selective pressures* and thus correlate with *functional importance* of sequence elements
- PHMM's and GPHMM's model the conservation patterns between *pairs* of taxa
- *PhyloHMM's* model conservation patterns between many taxa simultaneously
- *Combiner* programs incorporate conservation patterns as well as *expression data* and predictions of other programs
- *Conditional random fields* (*CRF's*) offer a principled way to combine disparate sources of information, and may in time come to dominate the comparative approaches to gene finding