

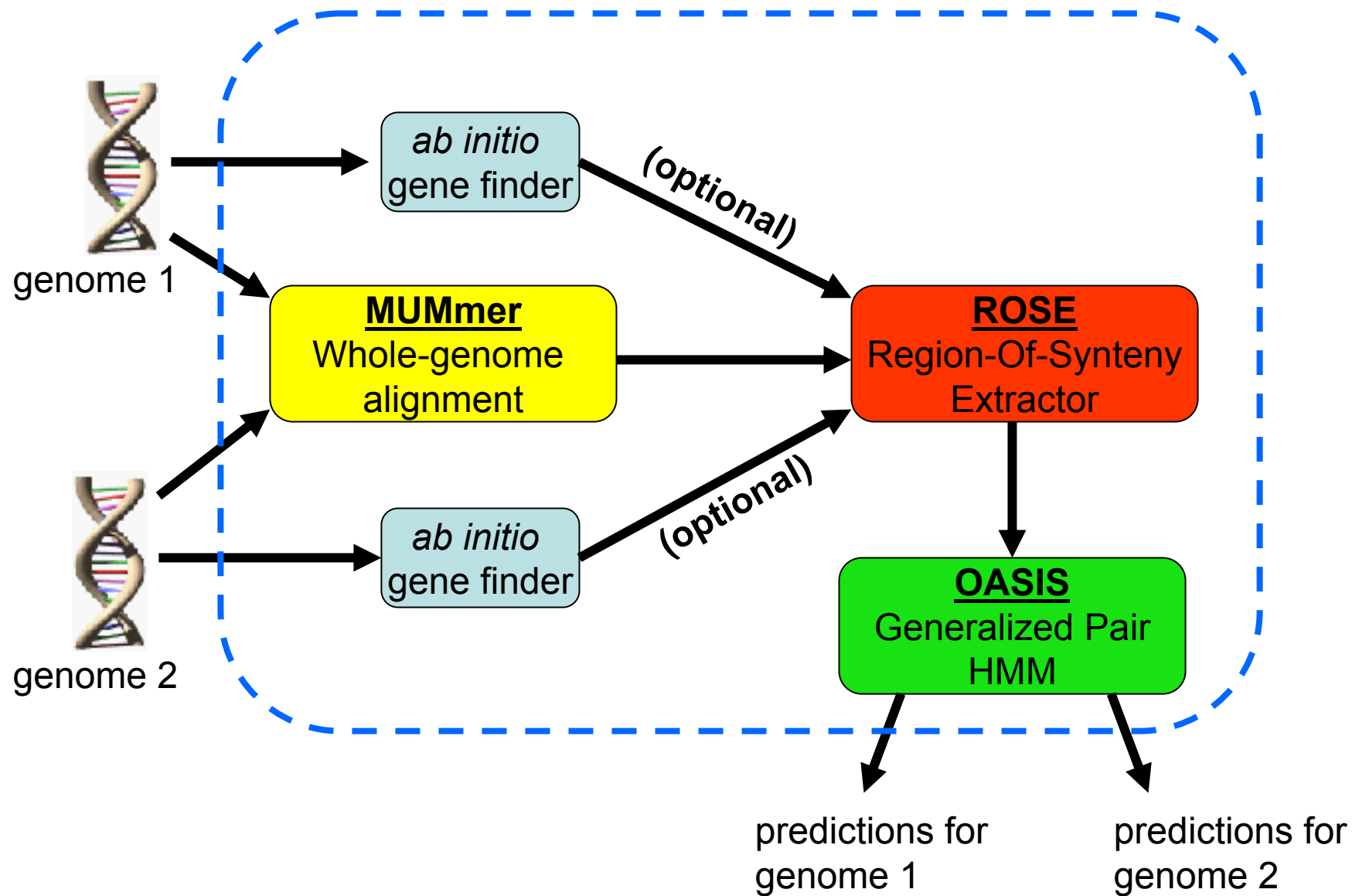
# Efficient Implementation of a Generalized Pair HMM for Comparative Gene Finding

*B. Majoros*

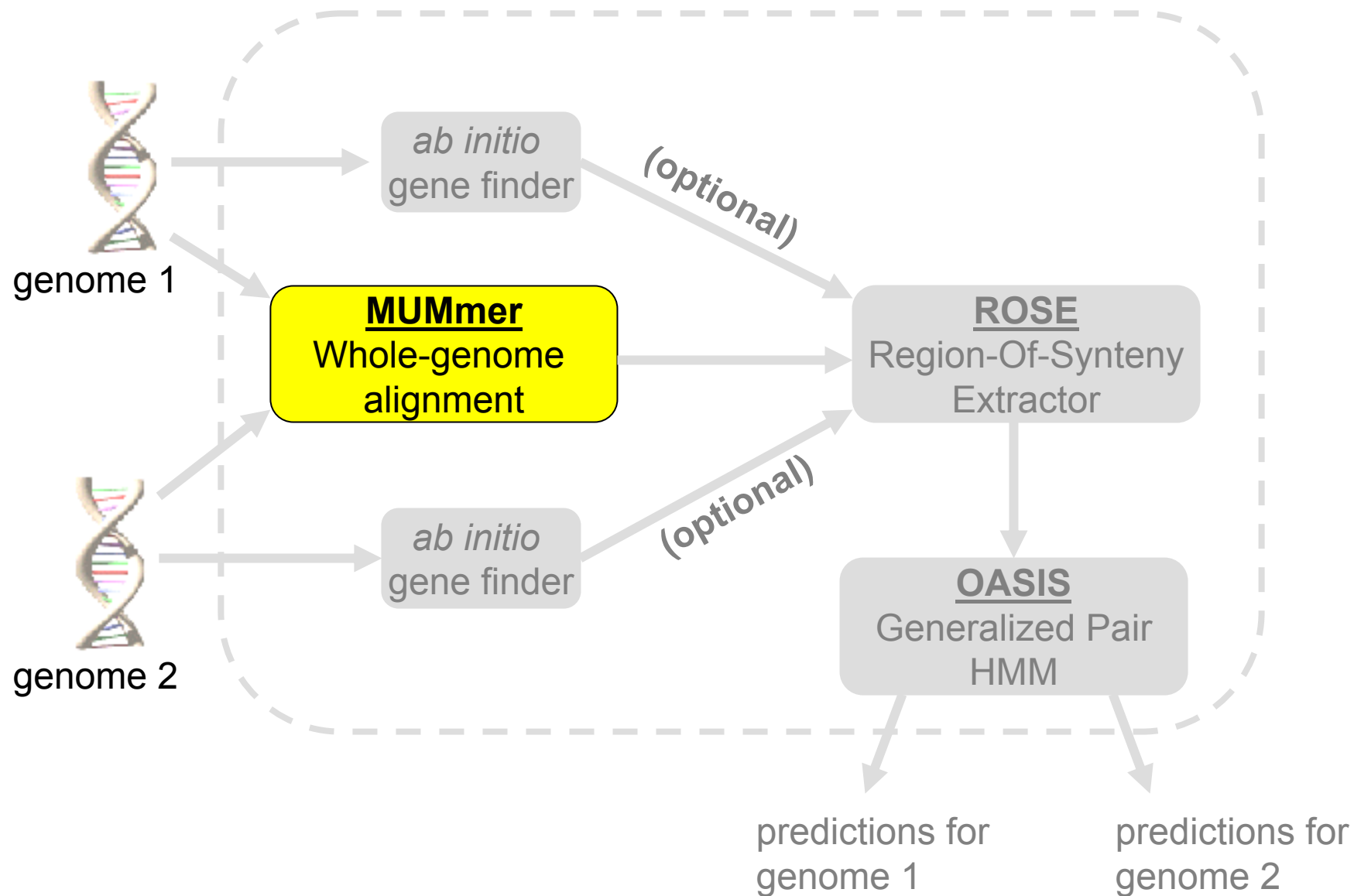
*M. Pertea*

*S.L. Salzberg*

# TWAIN



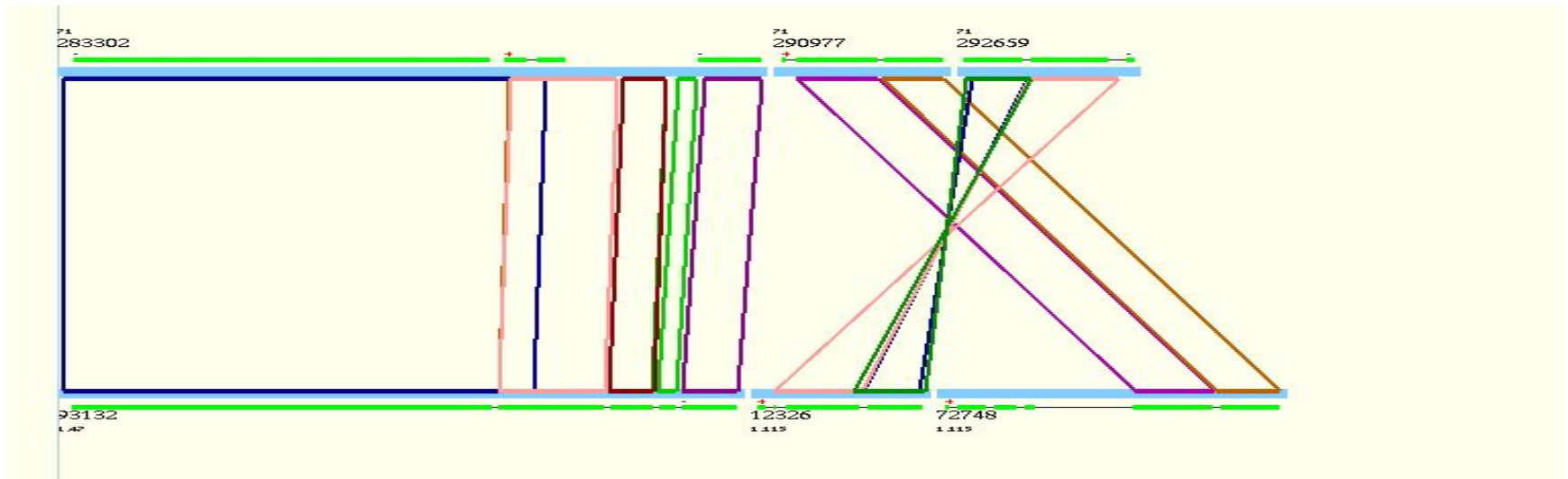
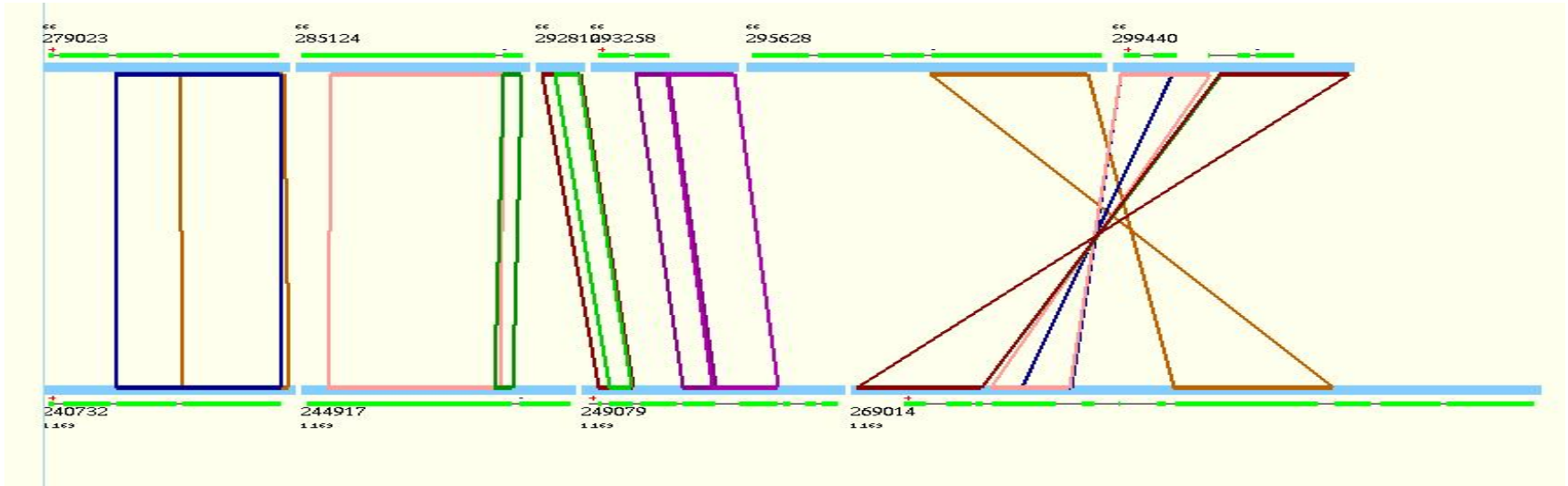
# TWAIN



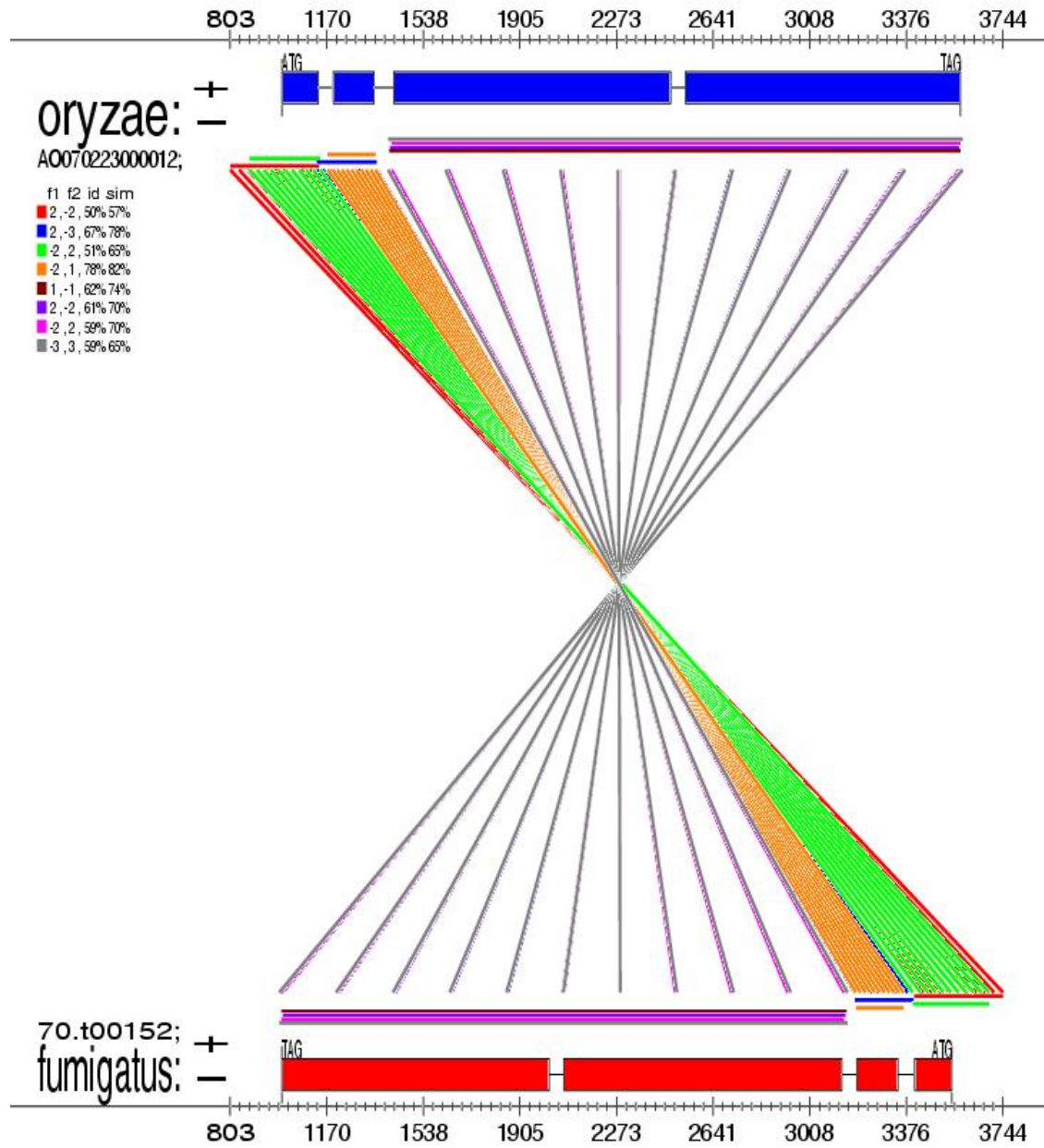
# What is MUMmer?

- Uses suffix trees to efficiently align whole genomes
- Can align in nucleotide space (NUCmer) or amino acid space (PROmer)
- Described in: Kurtz,S., Phillippy,A., Delcher,A.L., Smoot,M., Shumway,M., Antonescu,C. and Salzberg,S.L. (2004) Versatile and open software for comparing large genomes. *Genome Biol.*, **5**, R12.

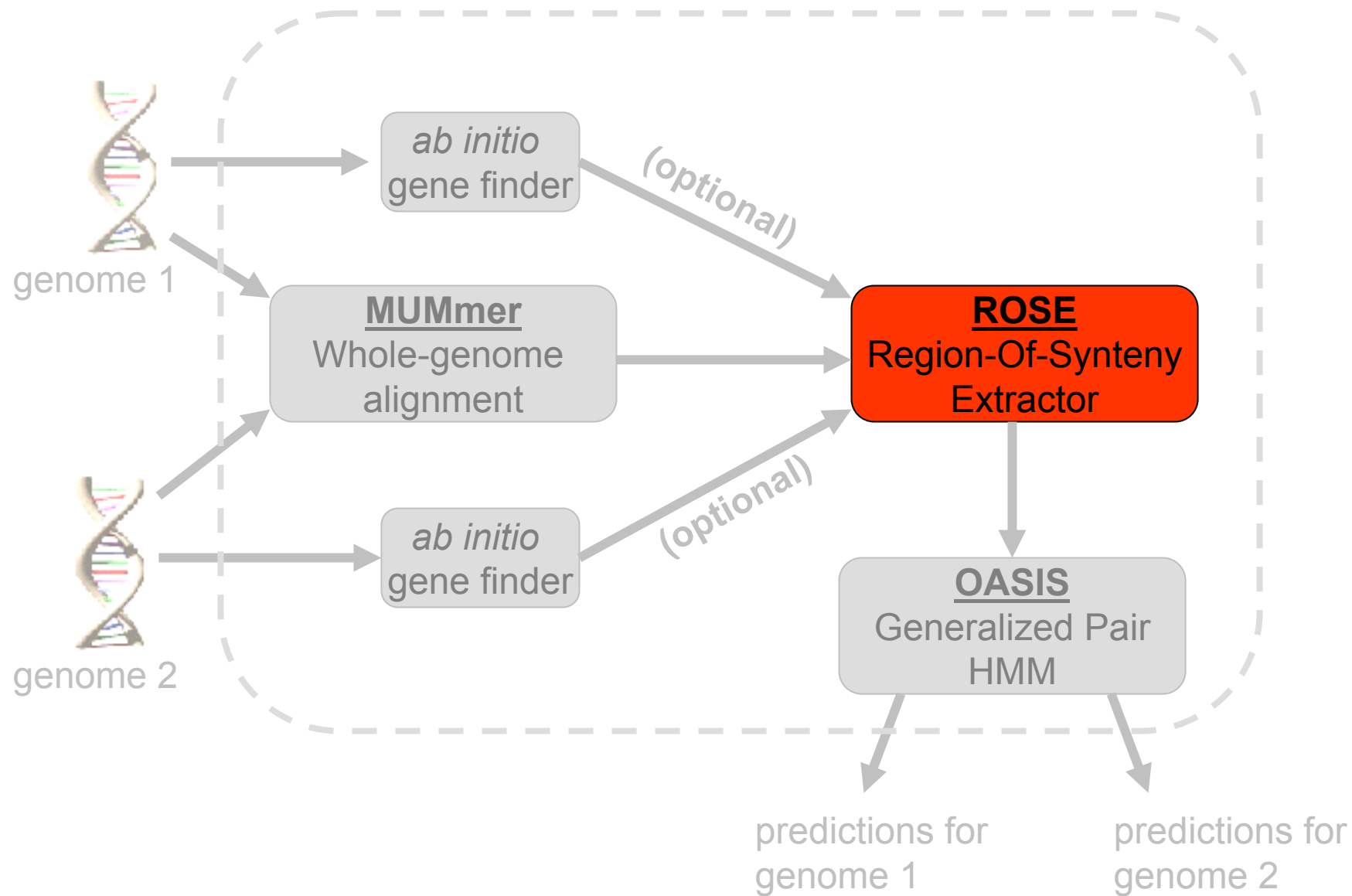
# Sample MUMmer Output



# Sample MUMmer Output



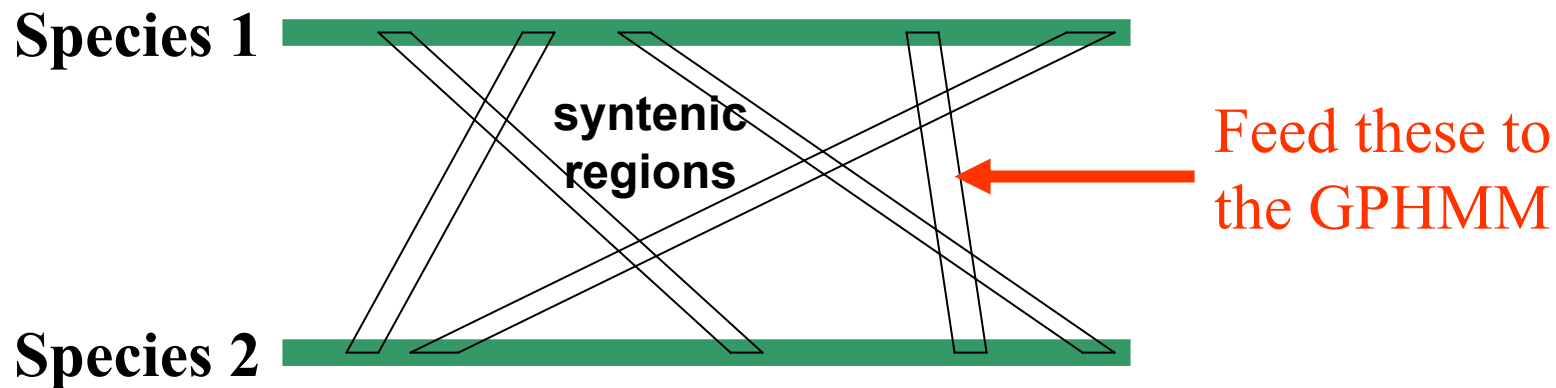
# TWAIN



# ROSE

## “Region-Of-Synteny Extractor”

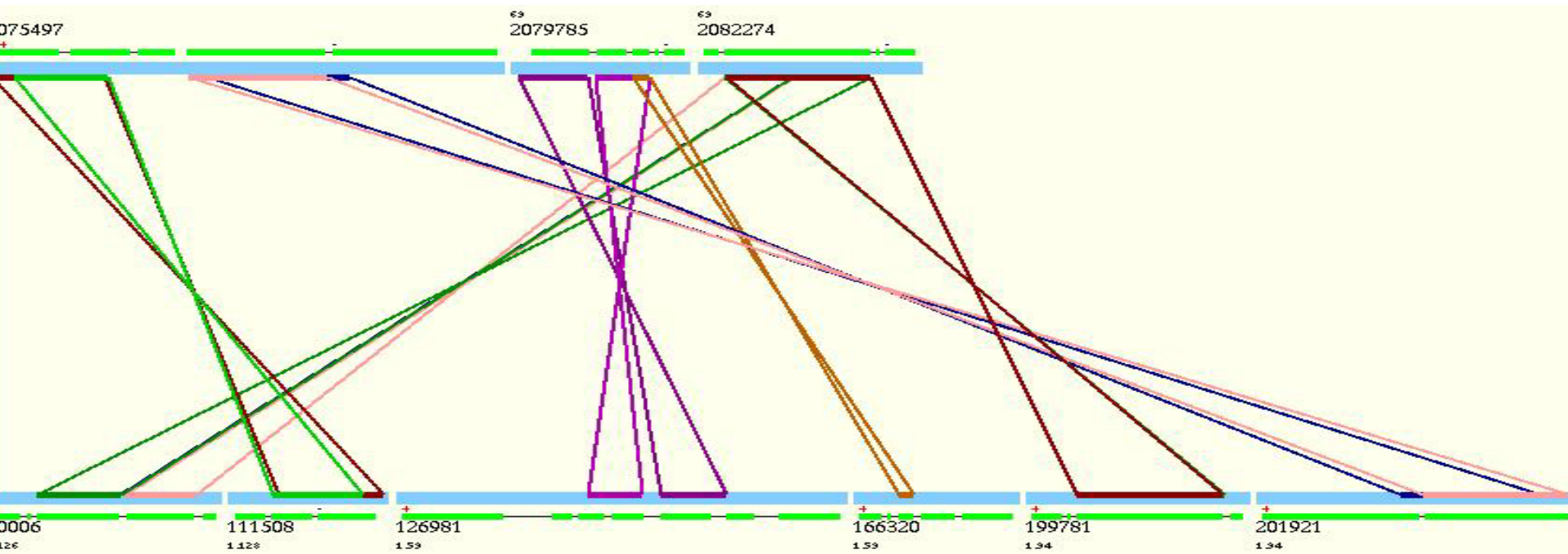
- Sole purpose is to identify (and preprocess) likely orthologous regions so they can be fed to OASIS



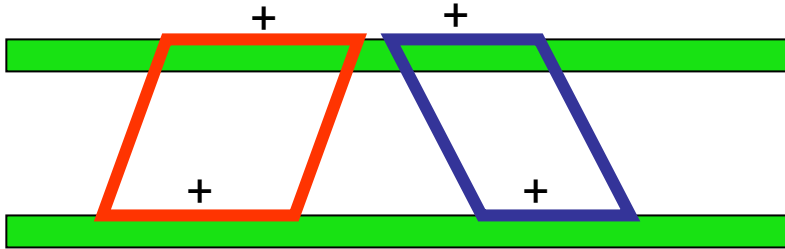


# How ROSE Works

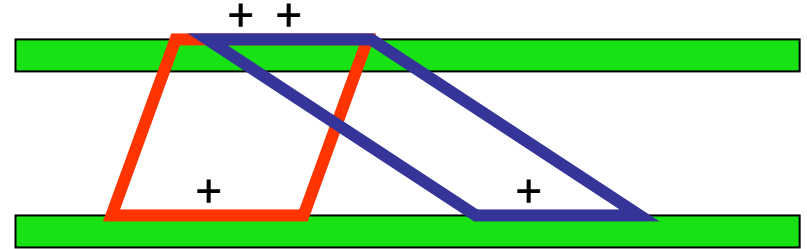
- ROSE performs maximal-length clustering of consistent MUMmer hits
- If *ab initio* predictions are available, ROSE uses them to extend the boundaries of clusters to avoid interrupting probable genes



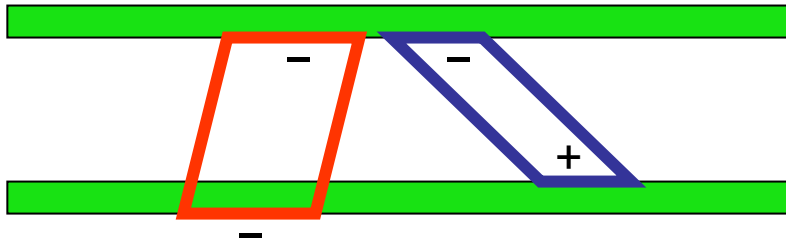
# Three Types of Inconsistency



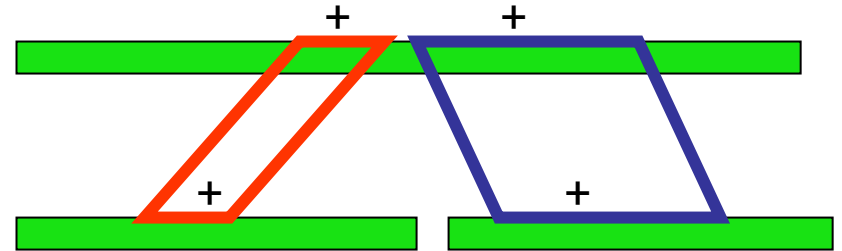
Consistent



Inconsistent – mapping to different locations



Inconsistent – mapping to inconsistent strands



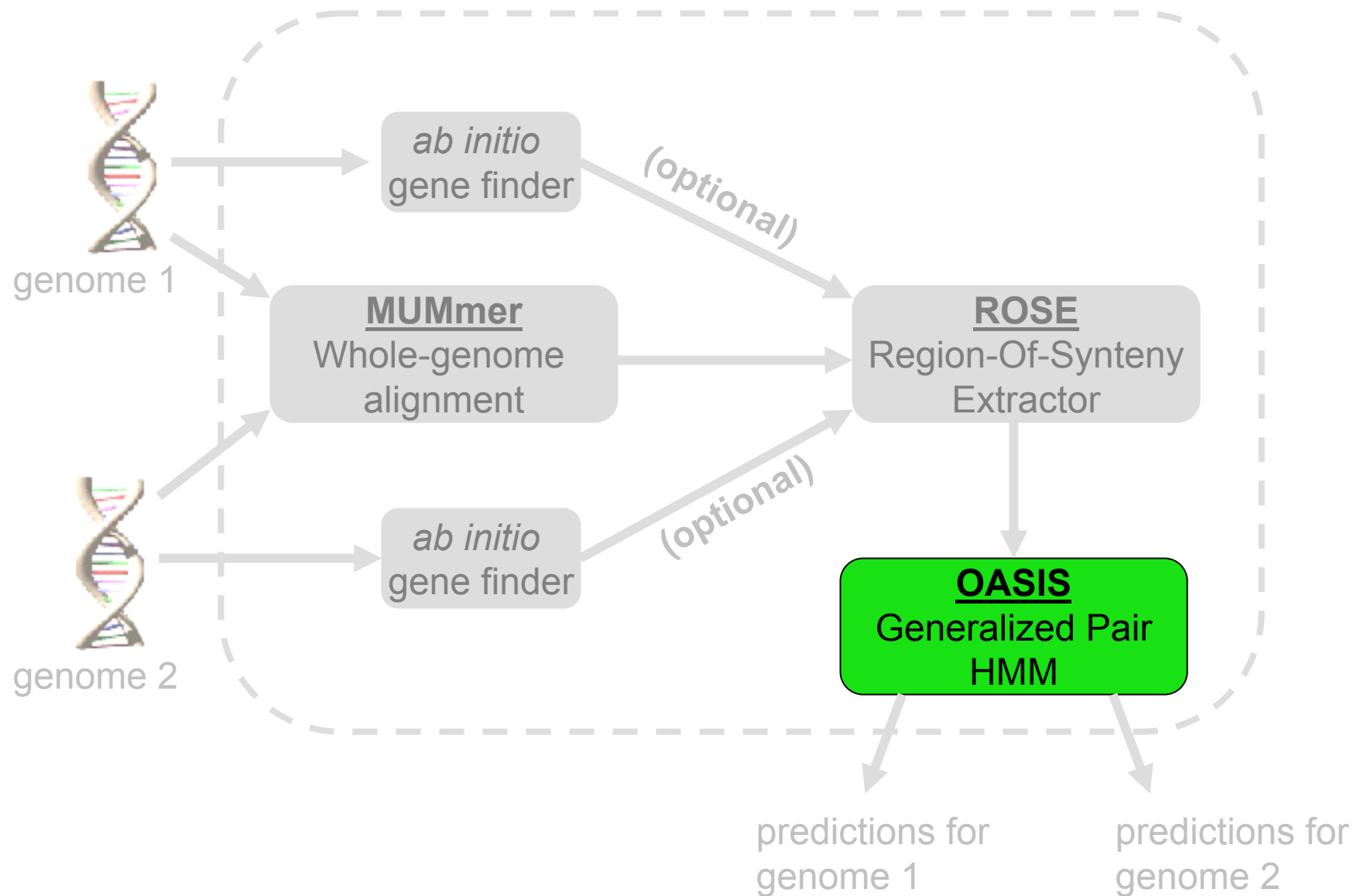
Inconsistent – mapping to different contigs

# Resolving the Inconsistencies

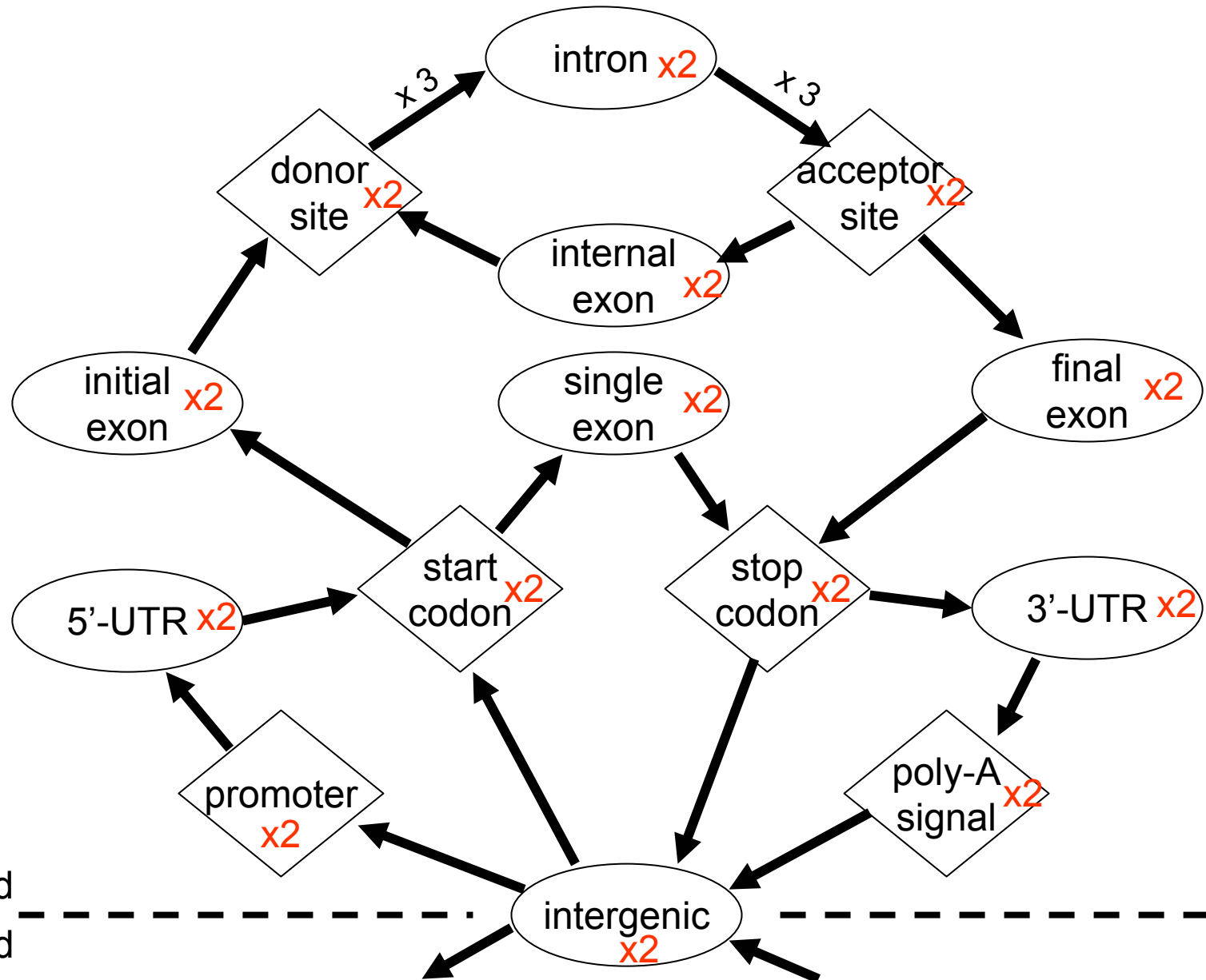
When ROSE determines that two MUMmer hits are not consistent, it does the following:

- 1) It refuses to cluster them together
- 2) If the difference in alignment scores for the two regions is not great, ROSE supplies both regions to OASIS via separate OASIS runs (even if the regions overlap)
- 3) If the difference in alignment scores is large, ROSE discards the MUMmer hit with the lower score

# TWAIN



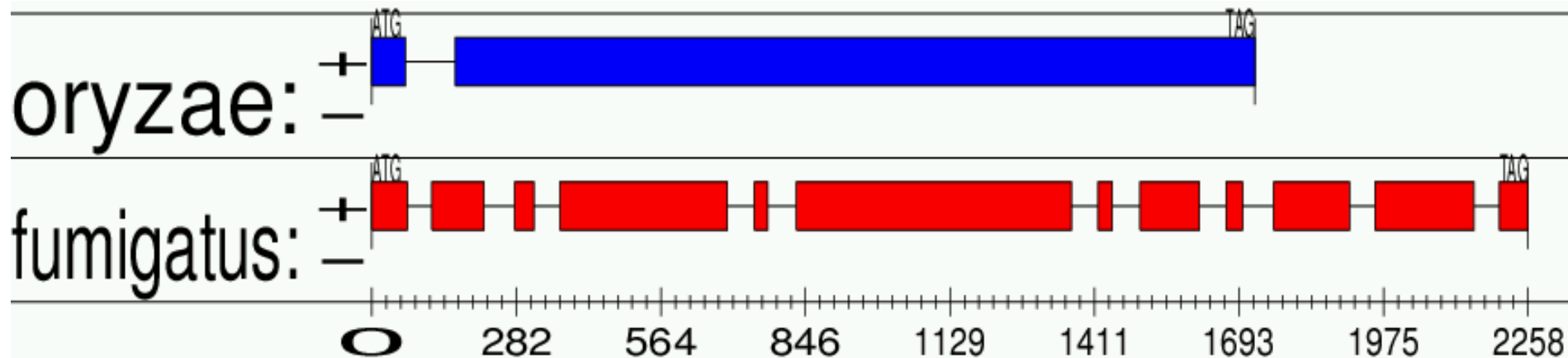
# Model Topology



# Conservation of Exon Structure

Our formulation of GPHMM operation assumes that orthologues have equal numbers of exons.

This assumption does not hold in many cases. Many *Aspergillus* orthologs have unequal numbers of exons:




This is a shortcoming that we hope to address in the near future.

# Background: GHMM Decoding

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

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

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


$$= \arg \max_{\phi} \prod_{i=1}^{n-1} \underbrace{P_e(S_i | q_i, d_i)}_{\text{emission}} \underbrace{P_t(q_i | q_{i-1})}_{\text{transition}} \underbrace{P_d(d_i | q_i)}_{\text{duration}}$$

# Generalizing to Pairs of Features

The emission probability  $P_e(S_i|q_i, d_i)$  can be replaced by a paired emission probability,  $\psi_e(S_{i,1}, S_{i,2}|q_i, d_{i,1}, d_{i,2})$ . The duration probability can likewise be replaced by a paired duration probability,  $\psi_d(d_{i,1}, d_{i,2}|q_i)$ :

$$\operatorname{argmax}_{\phi} \prod_{q_i \in \phi} \underbrace{\psi_e(S_{i,1}, S_{i,2} | q_i, d_{i,1}, d_{i,2})}_{\text{paired emission}} \cdot \underbrace{P_t(q_i | q_{i-1})}_{\text{transition}} \cdot \underbrace{\psi_d(d_{i,1}, d_{i,2} | q_i)}_{\text{paired duration}}$$

where:

$$\underbrace{\psi_e(S_{i,1}, S_{i,2} | q_i, d_{i,1}, d_{i,2})}_{\text{Markov chain}} = \underbrace{P_e(S_{i,1} | q_i, d_{i,1})}_{\text{Markov chain}} \cdot \underbrace{P_{\text{cond}}(S_{i,2} | S_{i,1}, q_i, d_{i,2})}_{\text{alignment}}$$

(if we ignore any dependence of  $S_{i,1}$  on  $d_{i,2}$  and of  $S_{i,2}$  on  $d_{i,1}$ ).

$P_e(S_{i,1}|q_i, d_{i,1})$  can be evaluated using the standard GHMM methods.

$\psi_d(d_{i,1}, d_{i,2}|q_i)$  can be estimated using an empirical distribution of  $|d_{i,1} - d_{i,2}|$  values, or more roughly using  $P_d(d_{i,1})$  or  $P_d(d_{i,2})$ .

What remains is to evaluate  $P_{\text{cond}}(S_{i,2}|S_{i,1}, q_i, d_{i,2})...$



# Efficiently Estimating $P_{\text{cond}}$

$P_{\text{cond}}(S_{i,2}|S_{i,1},q_i,d_{i,2})$  can be (very roughly) estimated from the approximate alignment score,  $P_{\text{ident}}$ :

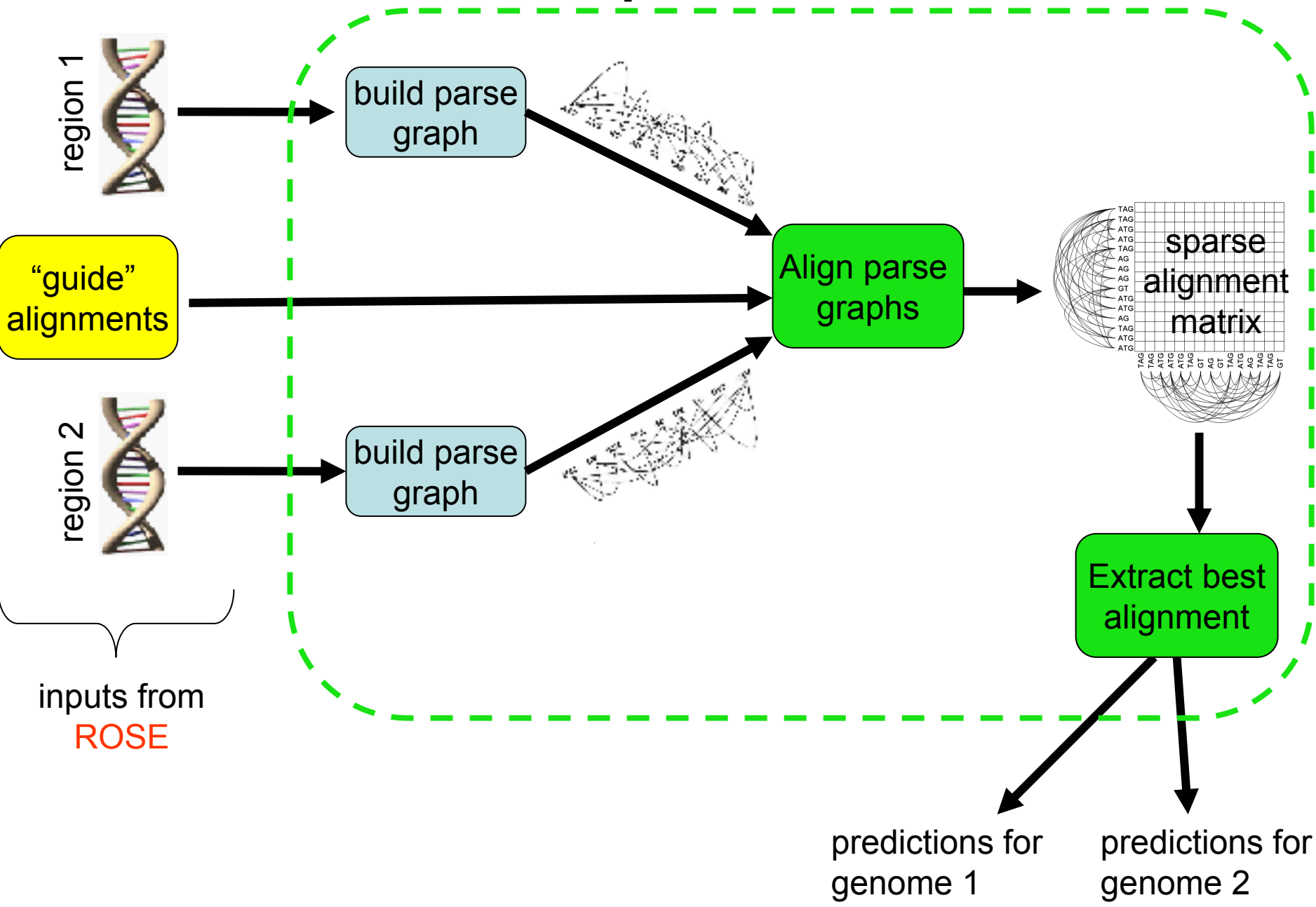
$$P_e(S_{i,2} | S_{i,1}, q_i, d_{i,2}) = (P_{\text{match}})^{P_{\text{ident}}L} (P_{\text{mismatch}})^{(1-P_{\text{ident}})L}$$

where  $L$  is the alignment length,  $P_{\text{match}}$  is a parameter to the GPHMM (probability of a match), and  $P_{\text{mismatch}} = 1 - P_{\text{match}}$ .

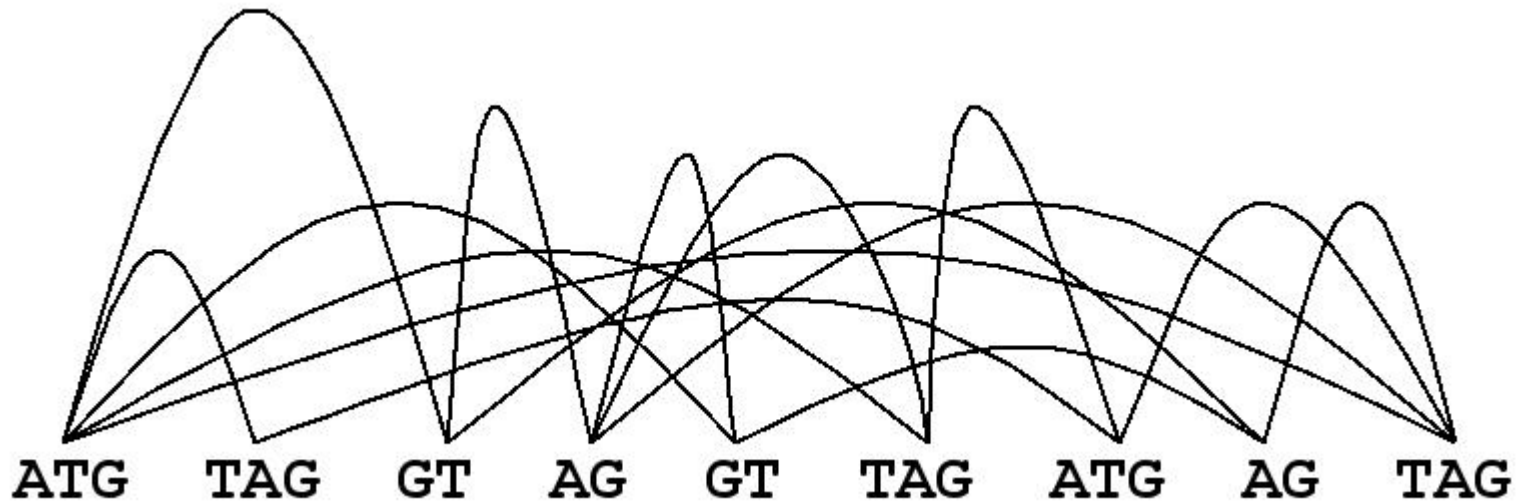
For putative noncoding features,  $P_{\text{ident}}$  is the *percent identity* estimated from the global NUCmer alignment.

For coding features,  $P_{\text{ident}}$  is the *percent similarity* (counting a BLOSUM score > 0 as a similar pair of residues) estimated from the nearest PROmer (amino acid) alignment.

# OASIS Implementation



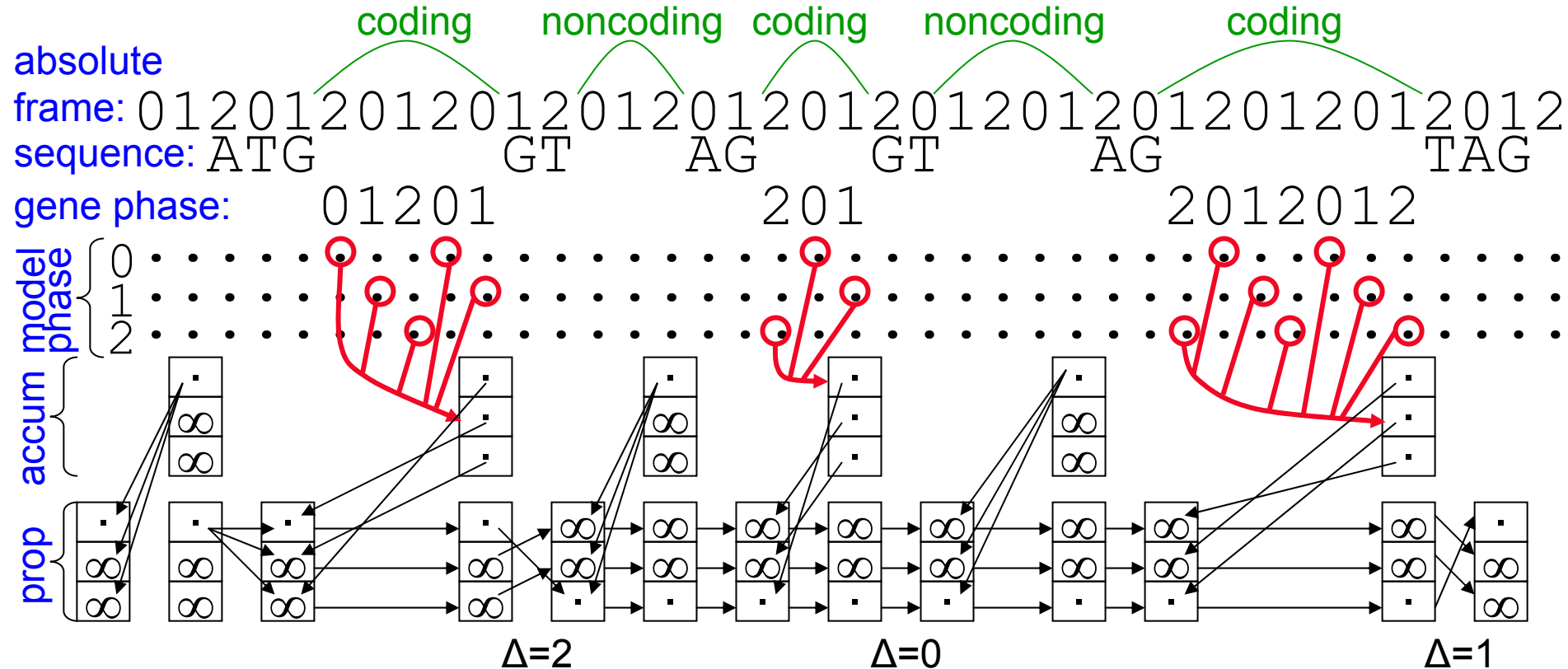
# What is a Parse Graph?



- represents all high-scoring ORFs
- each vertex is a signal and each edge is a feature such as an exon or intron
- a complete path through the graph from left to right gives a single gene prediction
- can be used to explore sub-optimal gene models
- when our GHMM's prediction is not exactly correct, the true gene model is often one of the top few sub-optimal parses.

# Building Parse Graphs

Our parse graphs are built using a special GHMM decoding algorithm that uses very little memory while also pruning away unpromising subgraphs:



# GHMM Memory Requirements

Our GHMM's memory requirements increase linearly, as do Genscan's, but with a much smaller constant factor:

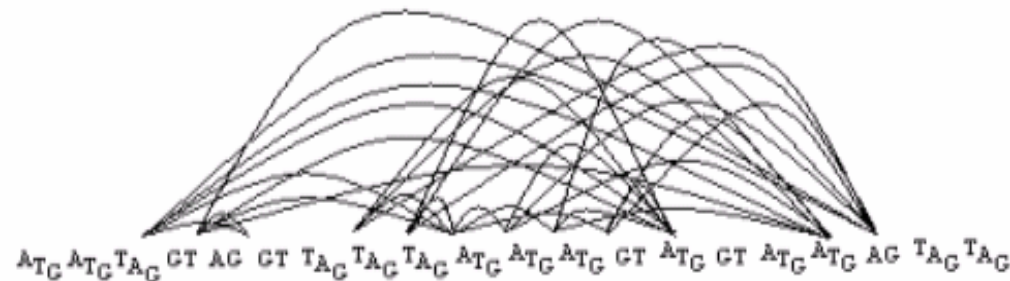
	Memory	Time
Genscan	445 Mb	2:57
Our GHMM	29 Mb	1:28

*Aspergillus* contig: 922,000 bases

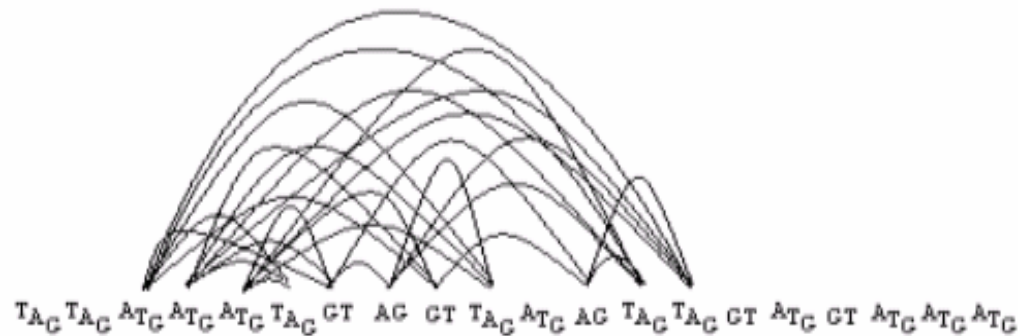
# Building Parse Graphs

Parse graphs for the two genomes are built independently.

Species 1:



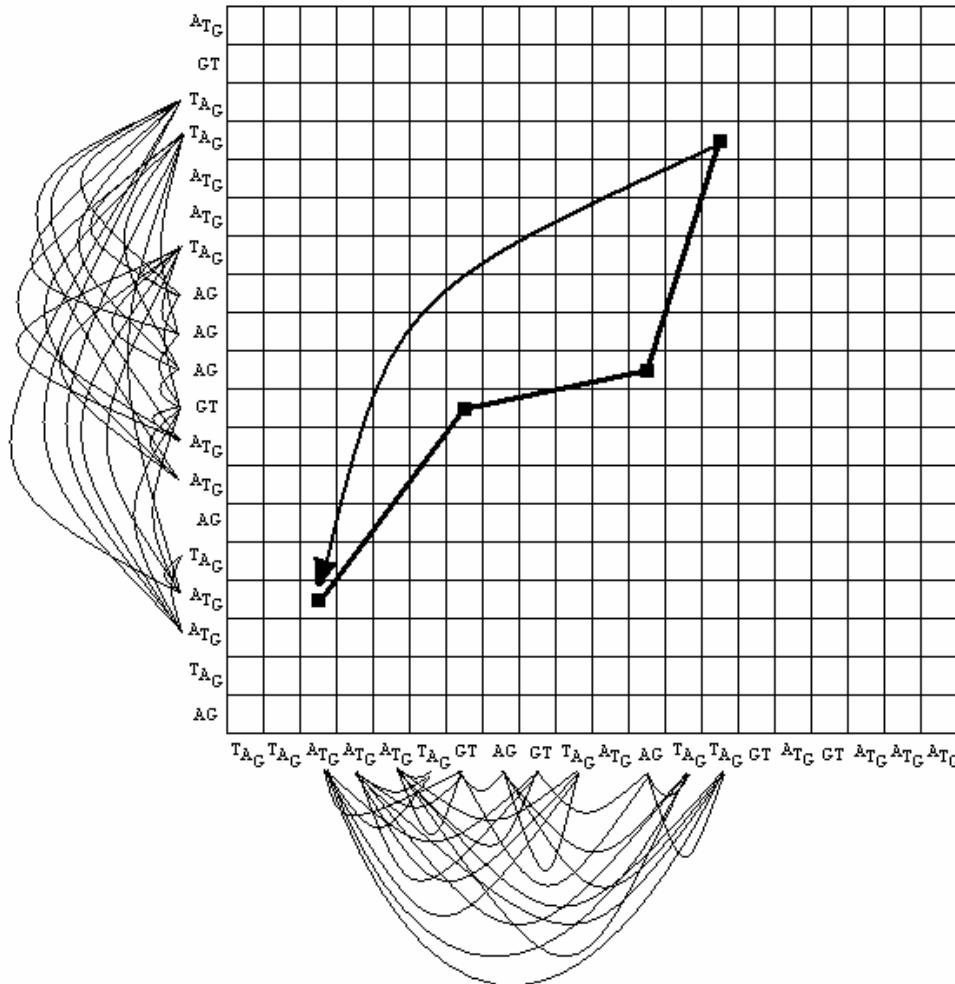
Species 2:



The graphs are weighted: edges are scored by Markov chains and vertices are scored by weight matrices.

# Aligning Parse Graphs

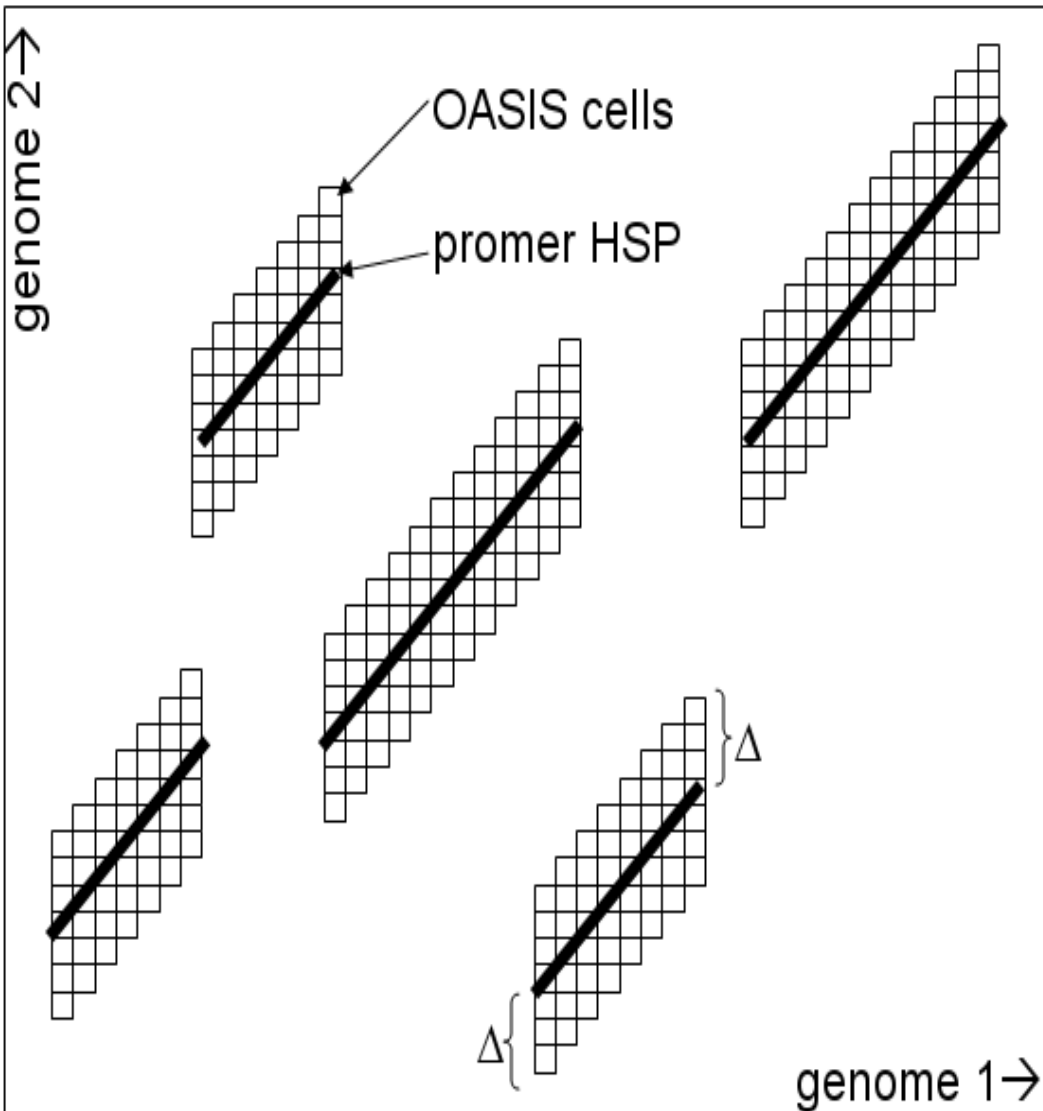
The two parse graphs are aligned using a global alignment algorithm. The optimal alignment corresponds to the chosen pair of syntenic gene predictions.



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

1. Only like signals can align, and
2. Two signals can align only if they have neighbors which also align
3. Standard phase constraints apply

# Sparse Alignment Matrix



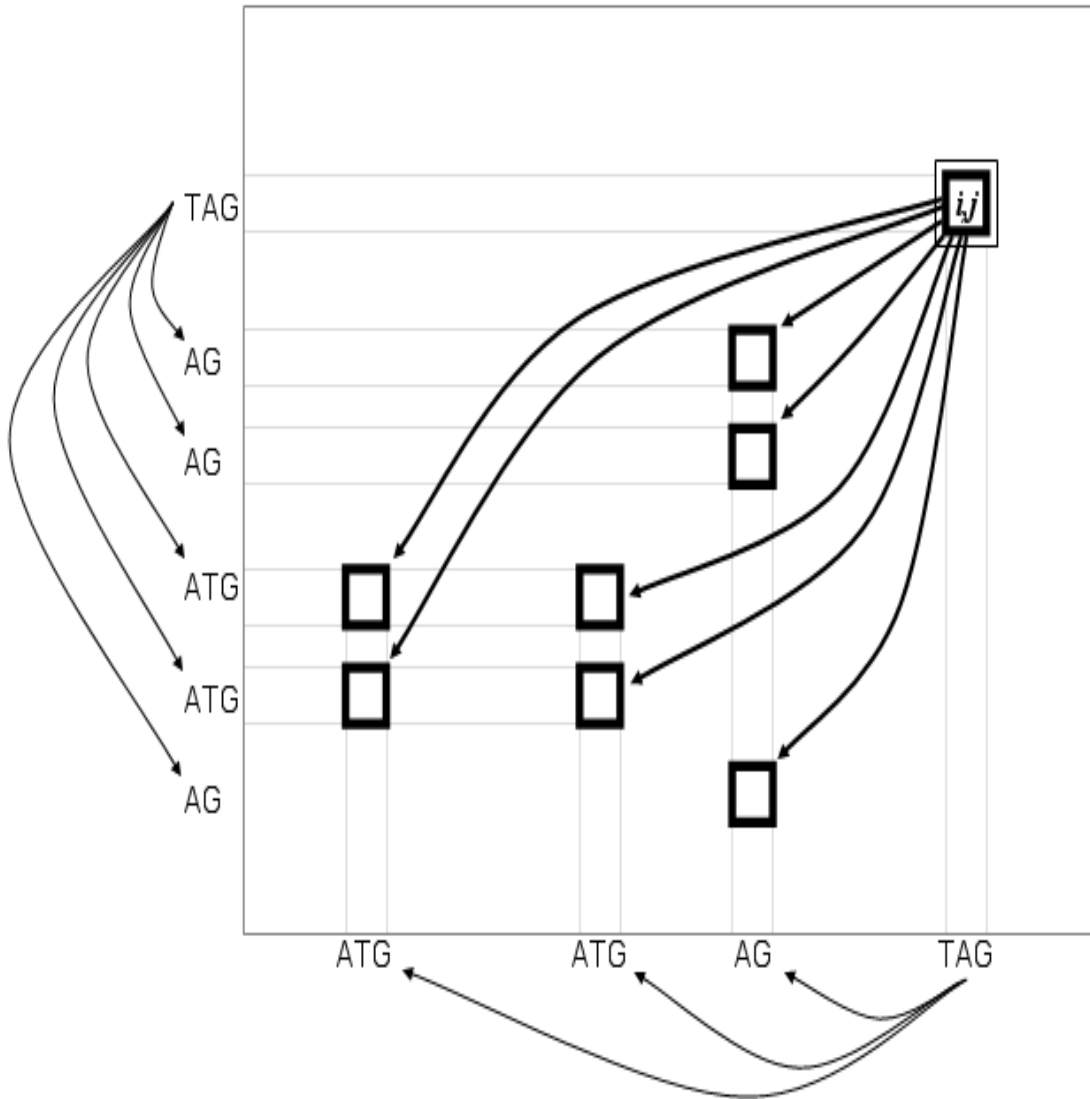
“Guide” alignments provided by MUMmer via ROSE dictate which portions of the alignment matrix to allocate. A user-specified  $\Delta$  influences the sparseness of the matrix.

Both coding (PROmer) and noncoding (NUCmer) guide alignments are used.

Coding HSPs are extended to include maximal ORFs.

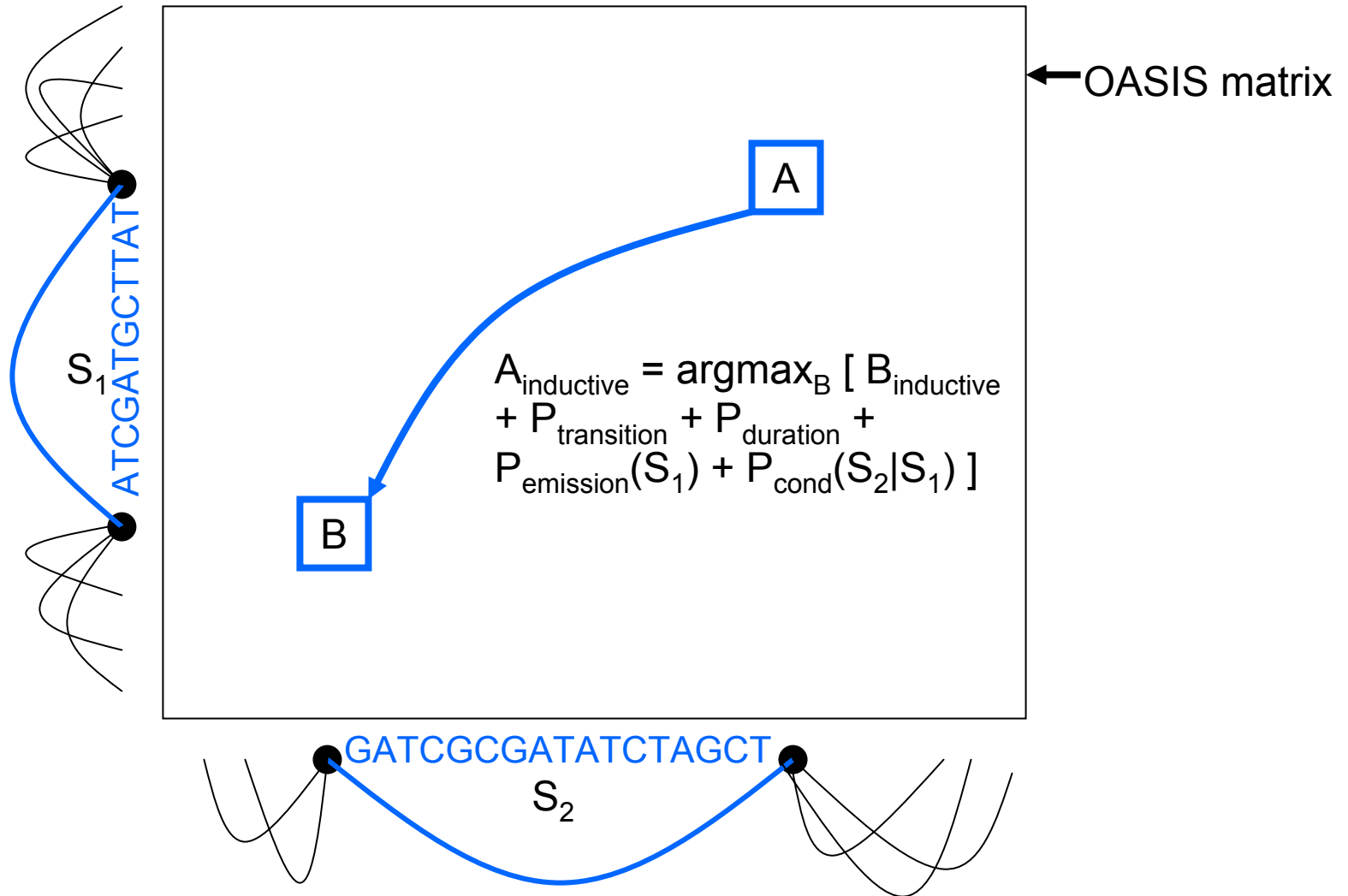


# The Alignment Trellis



An alignment “trellis” is formed by taking the cartesian product of edges in the parse graphs leading into corresponding vertices.

# Scoring the Trellis (in log space)

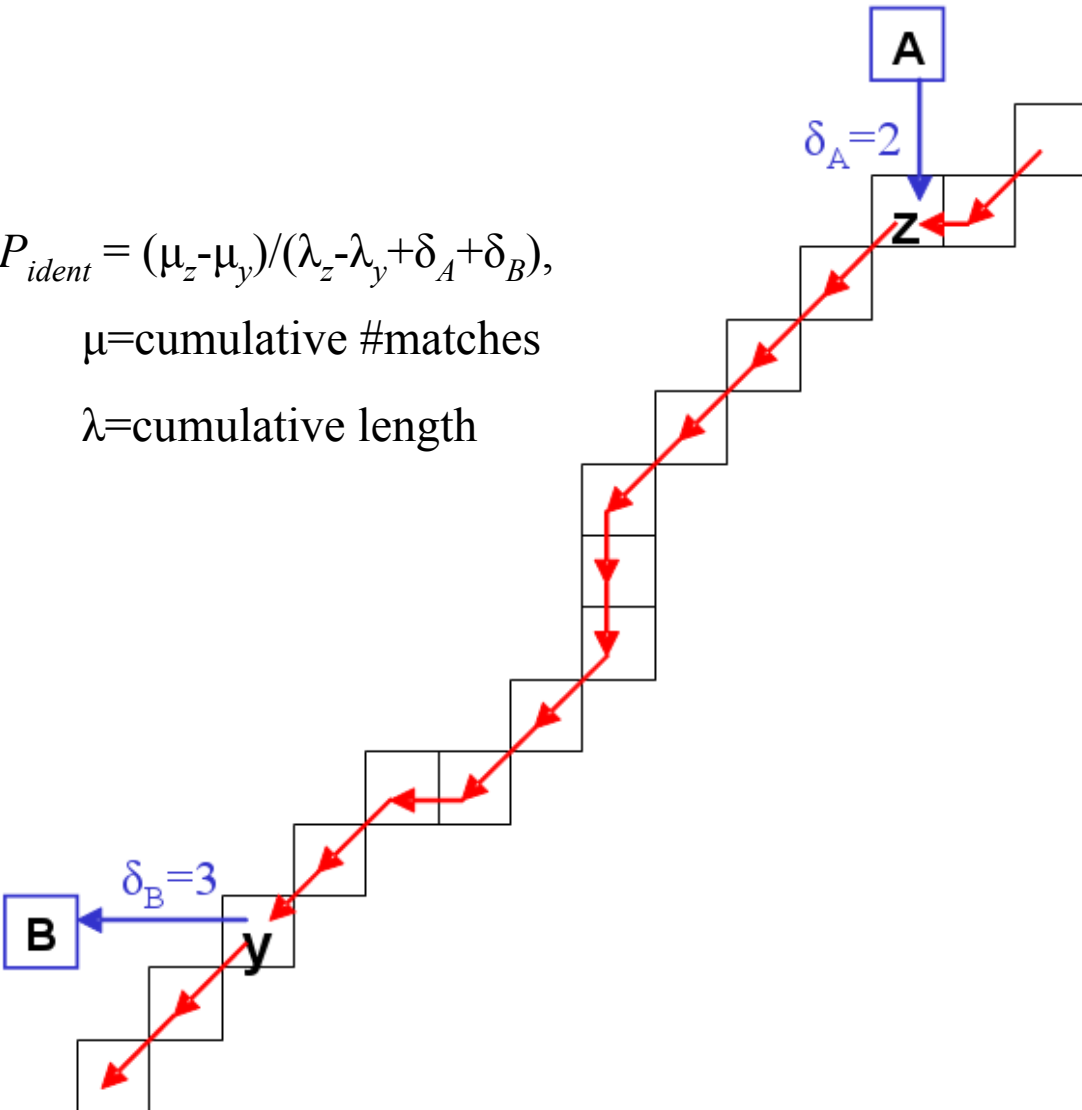


# Approximating $P_{ident}$ from Alignment

$$P_{ident} = (\mu_z - \mu_y) / (\lambda_z - \lambda_y + \delta_A + \delta_B),$$

$\mu$  = cumulative #matches

$\lambda$  = cumulative length



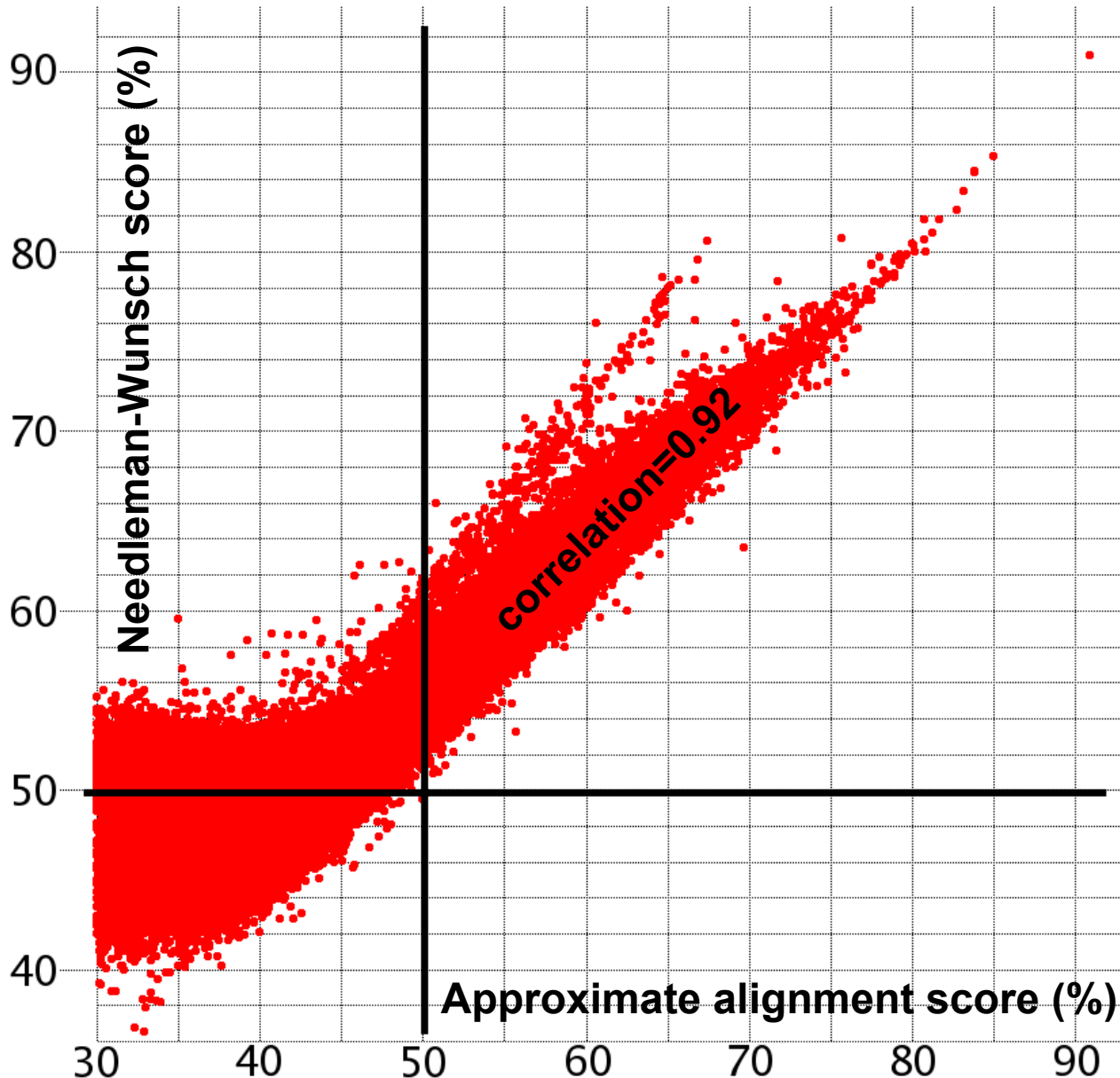
The “guide” alignment (red) is superimposed onto the OASIS matrix.

The alignment cells store prefix sums of alignment scores.

Jumps to and from the guide alignment incur indel penalties  $\delta_A$  and  $\delta_B$ .

The remaining portion of the alignment is scored in constant time by simple subtraction.

# Approx. vs. Exact Alignment

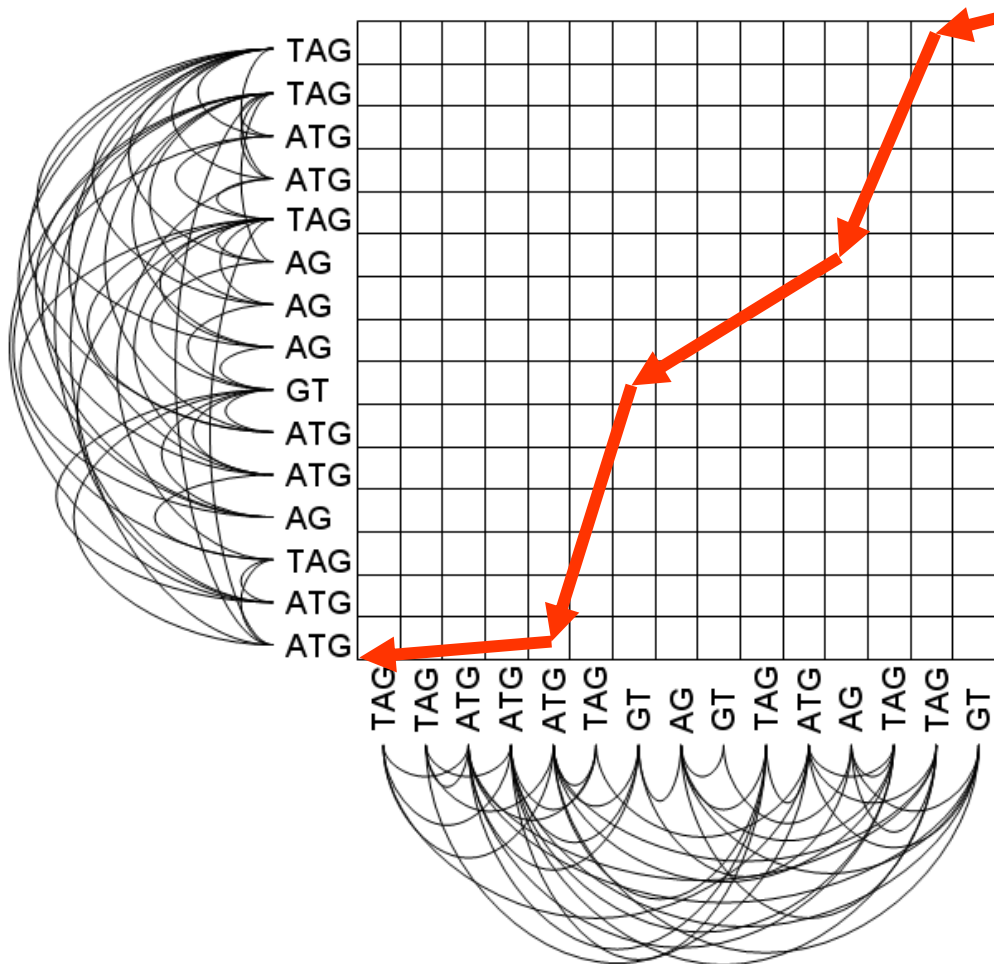


Correlation between approximate alignment scores (x) and full Needleman-Wunsch alignment scores (y), using percent identity as the alignment score in both cases.

Correlation coefficient was 0.92 for points above (50%,50%).

However, there is much variability, and scores are underestimated at the low end (slope<1).

# Obtaining a Pair of Predictions



Tracing back through the optimal trellis path highlights two corresponding paths in the parse graphs.

These paths outline the selected gene predictions in the two genomes.

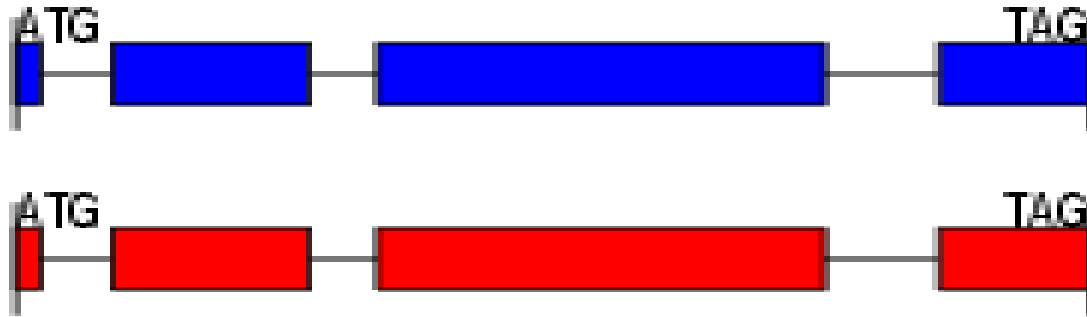
# Preliminary Results

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

	nucleotide accuracy	exon sensitivity	exon specificity	exact genes
Genscan	95%	50%	52%	22%
TigrScan	99%	78%	73%	54%
TWAIN	99%	89%	85%	74%

Accuracy results for OASIS applied to *Aspergillus fumigatus* using *A. nidulans* as the reference genome; TigrScan applied to *A. fumigatus*; and Genscan (trained for human) as a baseline for comparison.  $Sensitivity = TP / (TP + FN)$ ,  $specificity = TP / (TP + FP)$ ,  $TP$ =true positives,  $FP$ =false positives,  $TN$ =true negatives,  $FN$ =false negatives.  $Nucleotide\ accuracy = (TP + TN) / (TP + TN + FP + FN)$  where a positive is a coding nucleotide. For exons a true positive had both begin and end coordinates exactly correct. For genes a true positive had all exons correct. *Exact genes* shows the percentage of genes for which all coding exons were predicted correctly and where the predicted amino acid sequence is 100% correct.

# How Does Homology Help TWAIN?



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%

# Next Steps

- 1) Other species pairs, at different evolutionary distances (eg., ENCODE).
- 2) More than two species simultaneously
- 3) Allowing inserted introns
- 4) Paired duration distributions
- 5) Larger numbers of guide alignments
- 6) etc...



# TWAIN is Open Source Software



TWAIN

<http://www.tigr.org/software/pirate/twain/twain.html>

## **ACKNOWLEDGEMENTS**

This work was supported in part by NIH grant R01-LM007938. ROSE and OASIS were developed by MP and WHM, respectively, under the supervision of SLS. We thank Jennifer Wortman, Jonathan Crabtree, Jay Sundaram, and Christopher Hauser for providing the training and test data for this study.