Multiple Whole Genome Alignment

BMI/CS 776

www.biostat.wisc.edu/bmi776/

Spring 2016

Anthony Gitter

gitter@biostat.wisc.edu

Goals for Lecture

Key concepts

- the large-scale multiple-alignment task
- progressive alignment
- breakpoint identification
- undirected graphical models
- minimal spanning trees/forests

Multiple Whole Genome Alignment: Task Definition

Given

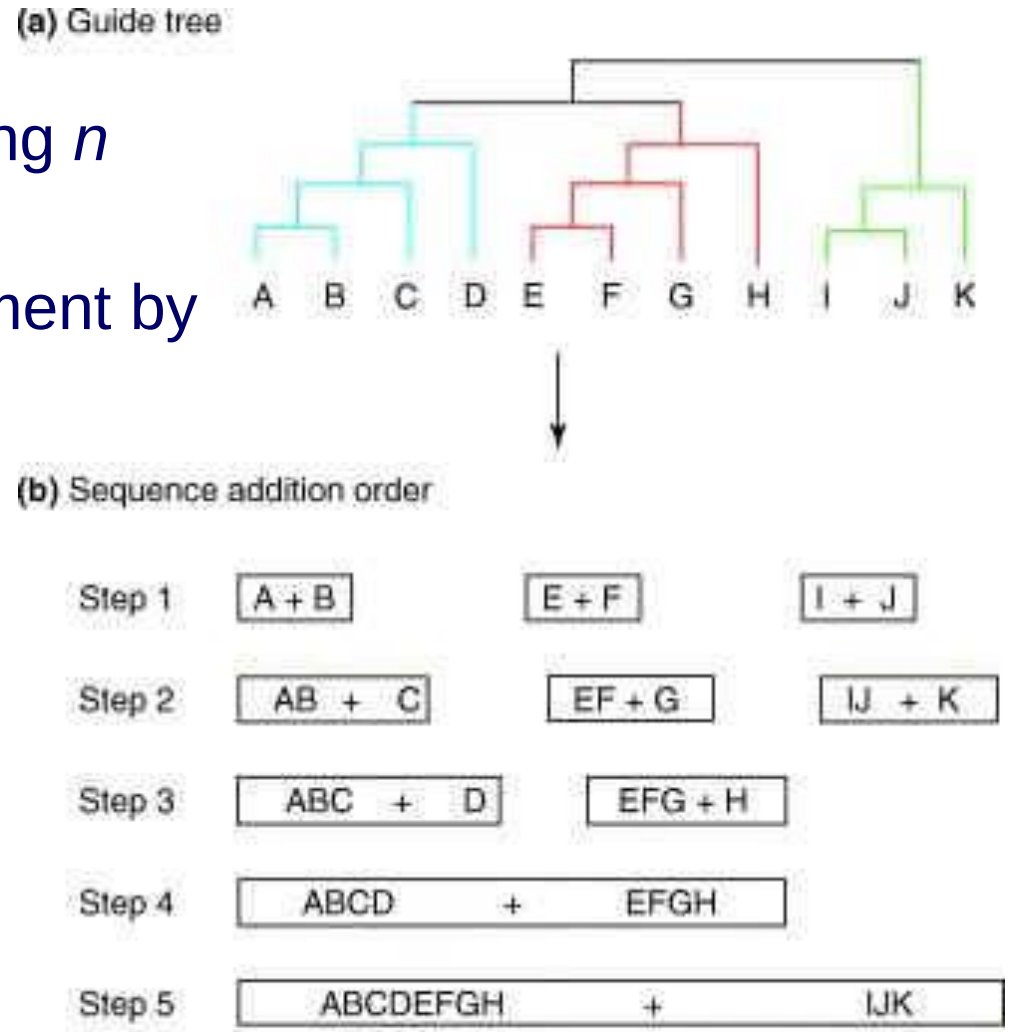
- A set of $n > 2$ genomes (or other large-scale sequences)

Do

- Identify all corresponding positions between all genomes, allowing for substitutions, insertions/deletions, and *rearrangements*

Progressive Alignment

- Given a *guide tree* relating *n* genomes
- Construct multiple alignment by performing *n*-1 pairwise alignments

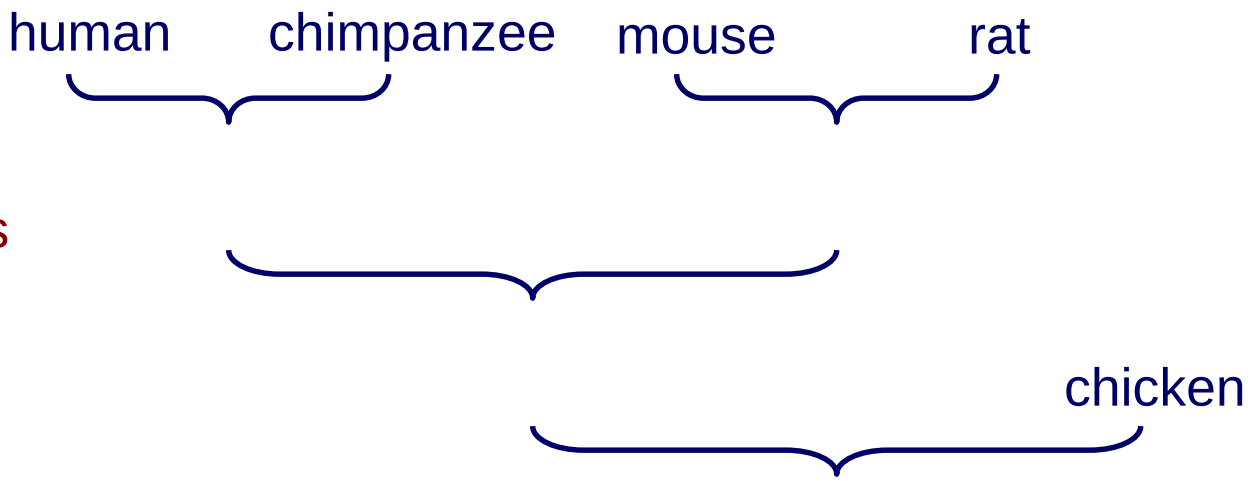


Progressive Alignment: MLAGAN Example

align pairs
of sequences

align multi-sequences
(alignments)

align multi-sequence
with sequence



Progressive Alignment: MLAGAN Example

Suppose we're aligning the multi-sequence X/Y with Z

1. anchors from X-Z and Y-Z become anchors for X/Y-Z
2. overlapping anchors are reweighted
3. LIS algorithm is used to chain anchors

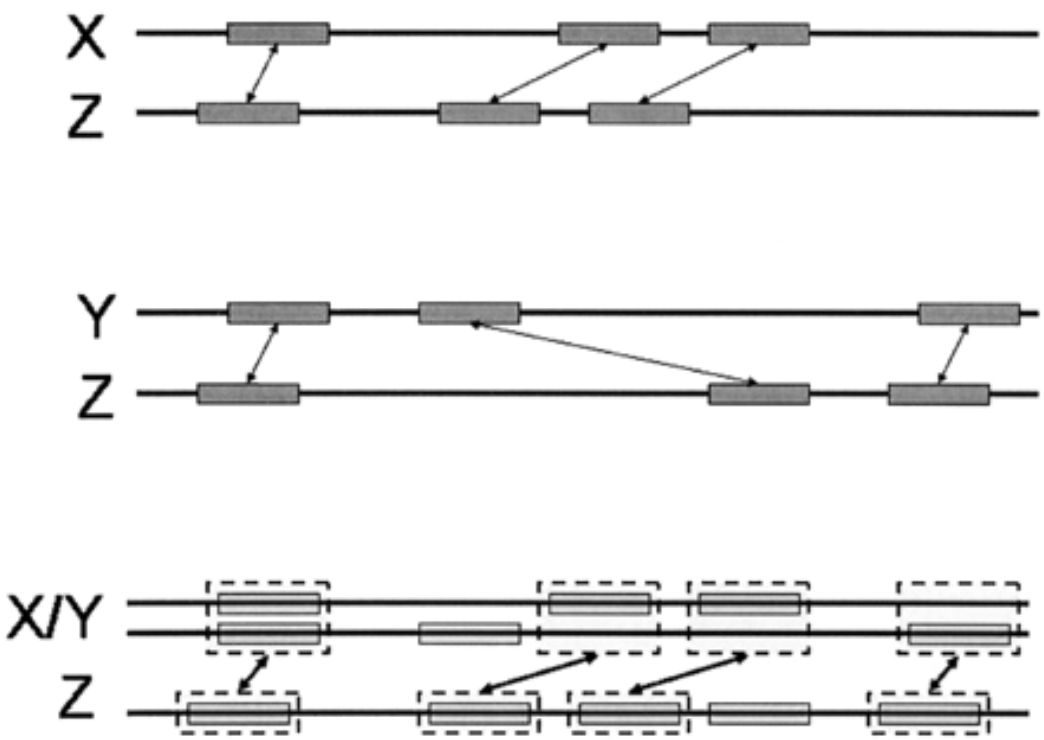
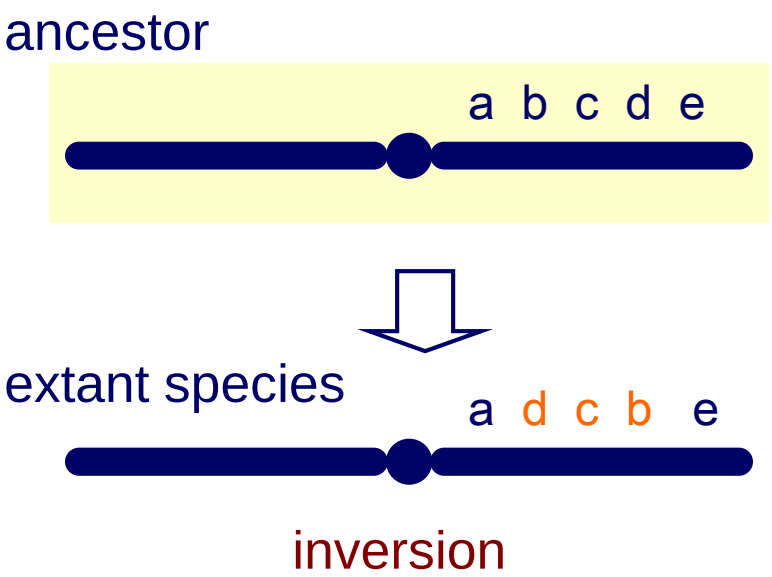
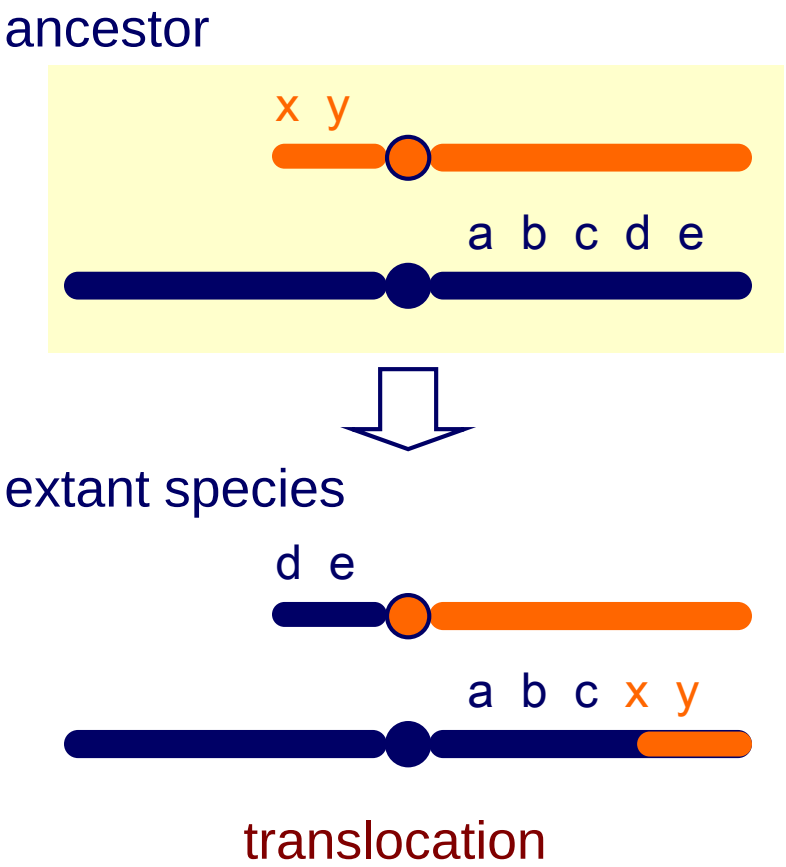


Figure from: Brudno et al. *Genome Research*, 2003

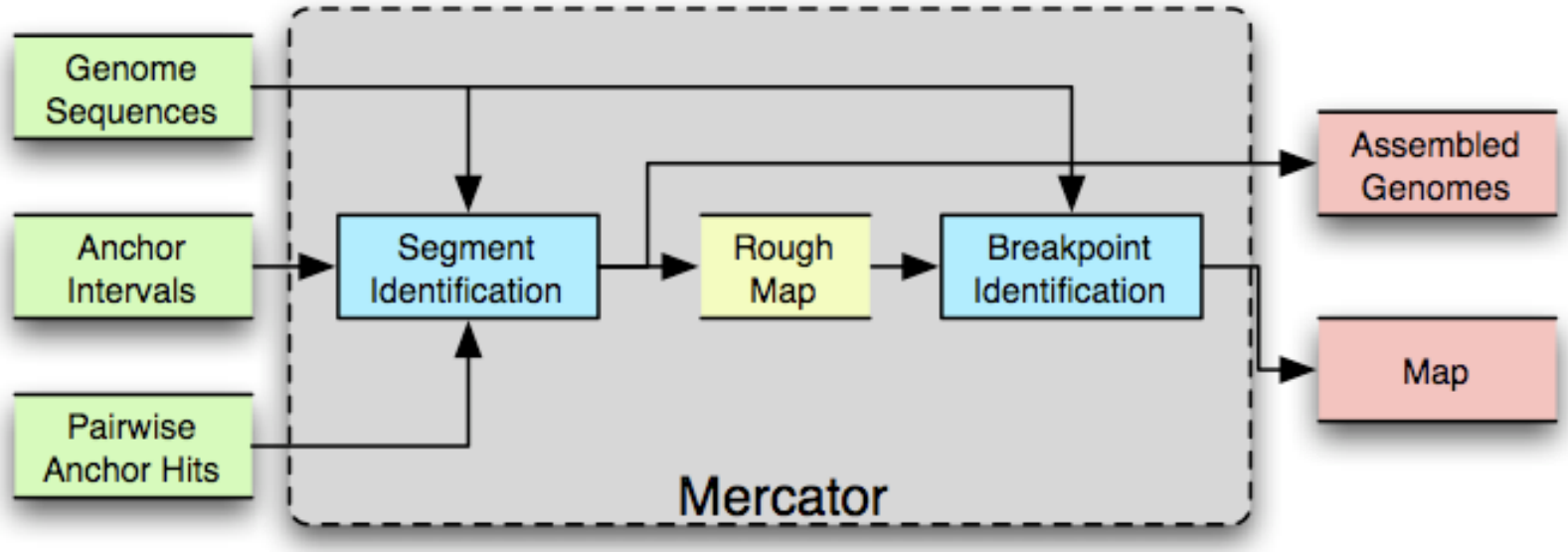
Genome Rearrangements





- Can occur within a chromosome or across chromosomes
- Can have combinations of these events

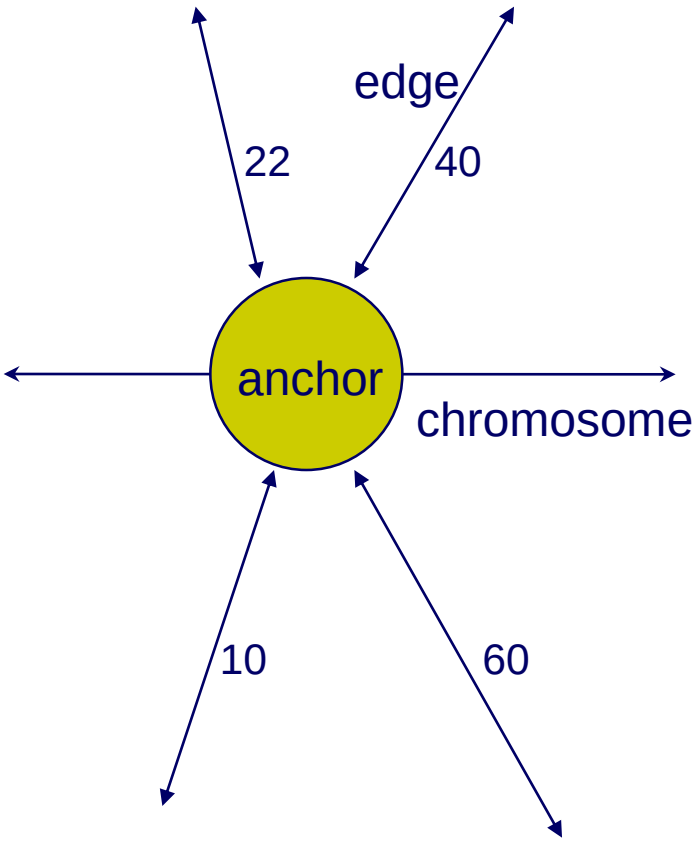
Mercator



- Orthologous segment identification: graph-based method
- Breakpoint identification: refine segment endpoints with a graphical model

Establishing Anchors Representing Orthologous Segments

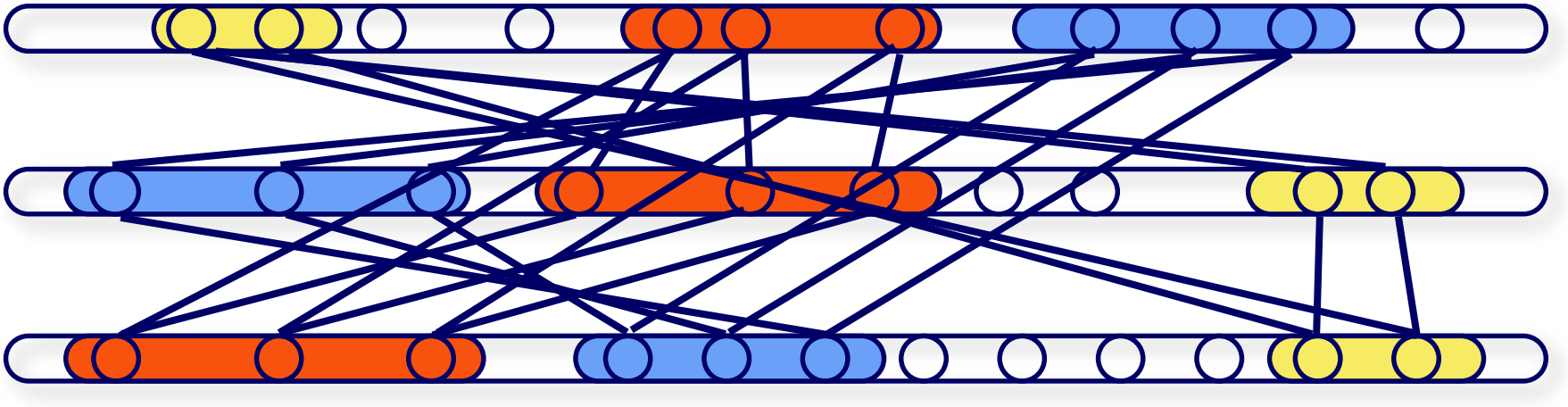
- Anchors can correspond to genes, exons or MUMS
- E.g., may do all-vs-all pairwise comparison of genes
- Construct graph with anchors as vertices and high-similarity hits as edges (weighted by alignment score)



Rough Orthology Map

k-partite graph with edge weights

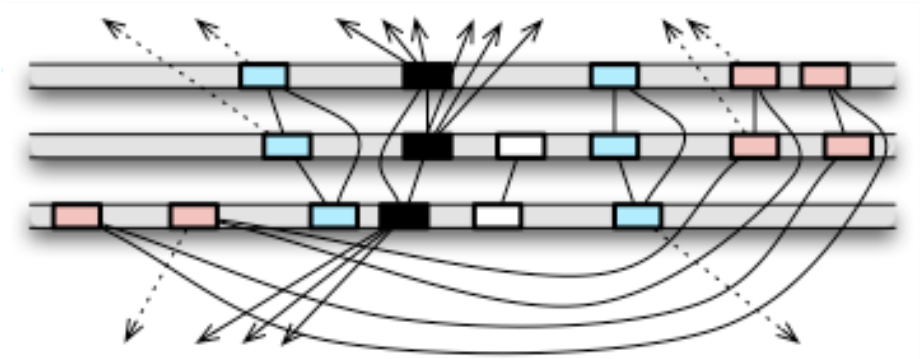
vertices = anchors, edges = sequence similarity



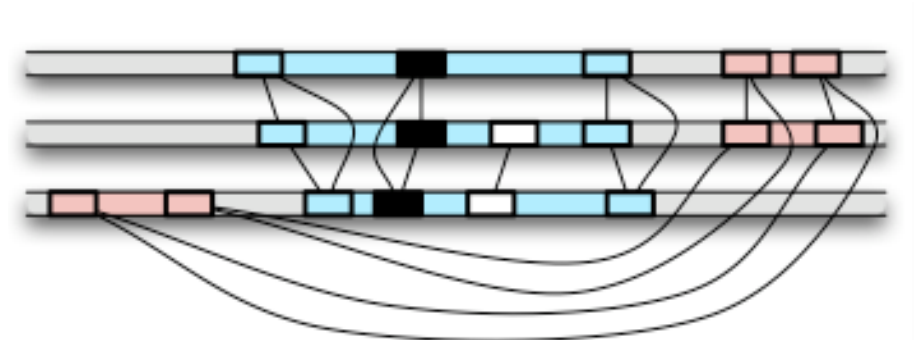
Greedy Segment Identification

- for i = k to 2 do
  - identify repetitive anchors (depends on number of high-scoring edges incident to each anchor)
  - find “best-hit” anchor cliques of size ≥ i
  - join colinear cliques into *segments*
  - filter edges not consistent with significant segments

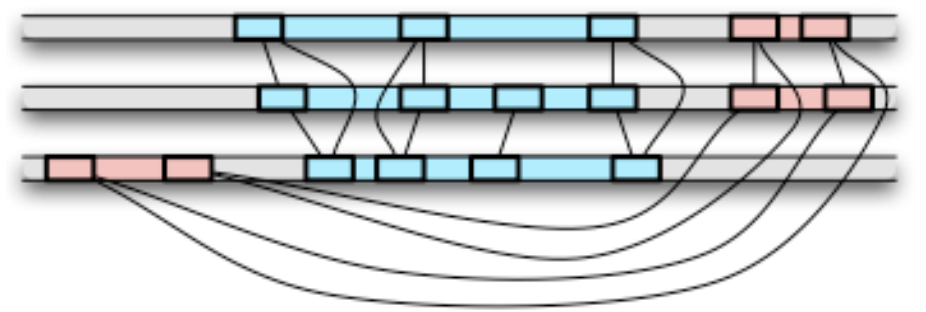
Mercator Example



Repetitive elements (black anchors) are identified; 3-cliques (red and blue anchors) are found



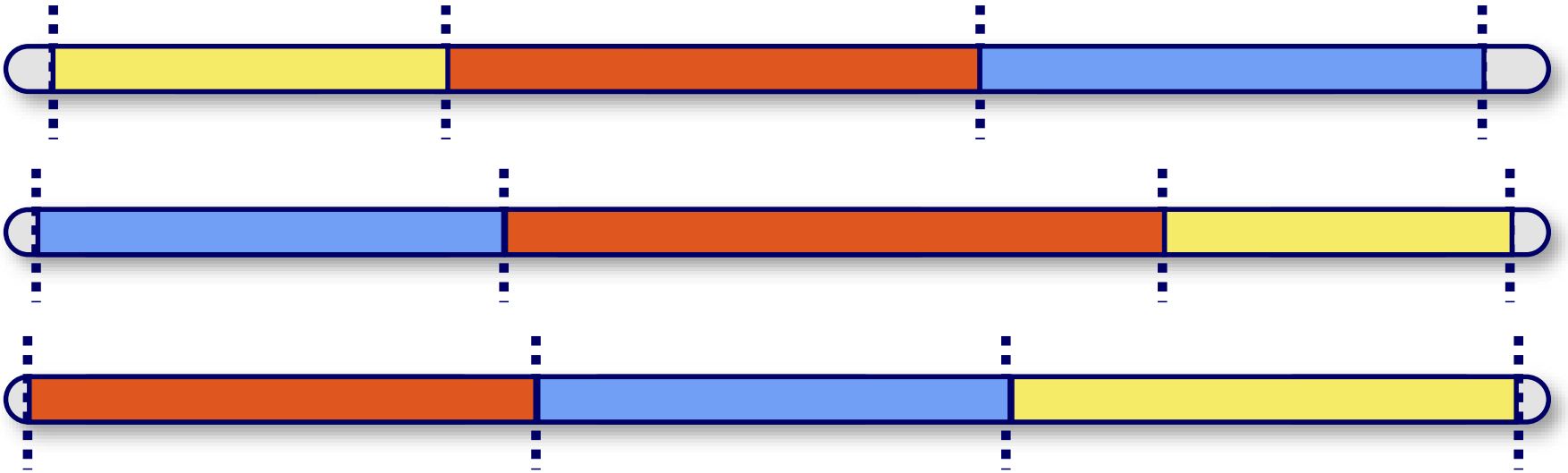
Segments are formed by red and blue anchors; inconsistent edges are filtered



2-cliques are found and incorporated into segments

Refining the Map: Finding Breakpoints

- *Breakpoints*: the positions at which genomic rearrangements disrupt colinearity of segments

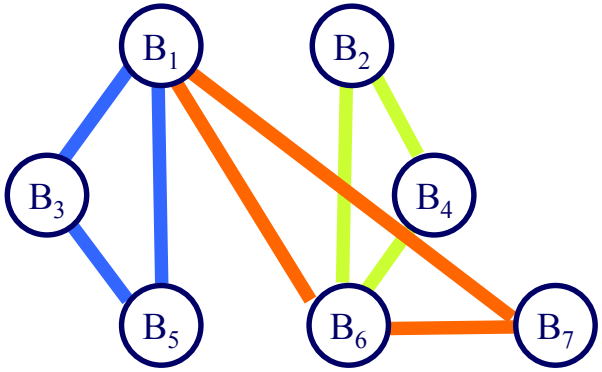


- Mercator finds breakpoints by using inference in an *undirected graphical model*

Undirected Graphical Models

- An undirected graphical model represents a probability distribution over a set of variables using a factored representation

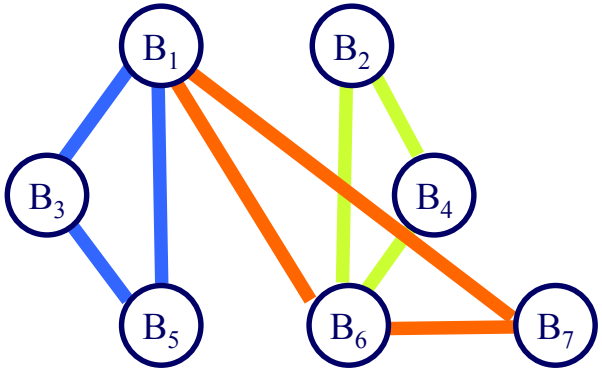
$$p(\mathbf{b}) = \frac{1}{Z} \prod_{C \in cliques} \psi_C(\mathbf{b}_C)$$



B_i	random variable
$\mathbf{b}$	assignment of values to all variables (breakpoint positions)
$\mathbf{b}_C$	assignment of values subset of variables in C
ψ_C	function (called a potential) representing the "compatibility" of a given set of values
Z	normalization term

Undirected Graphical Models

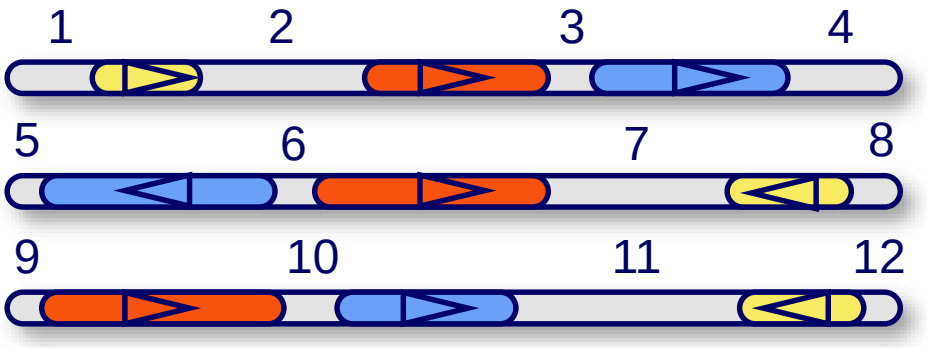
$$p(\mathbf{b}) = \frac{1}{Z} \prod_{C \in cliques} \psi_C(\mathbf{b}_C)$$

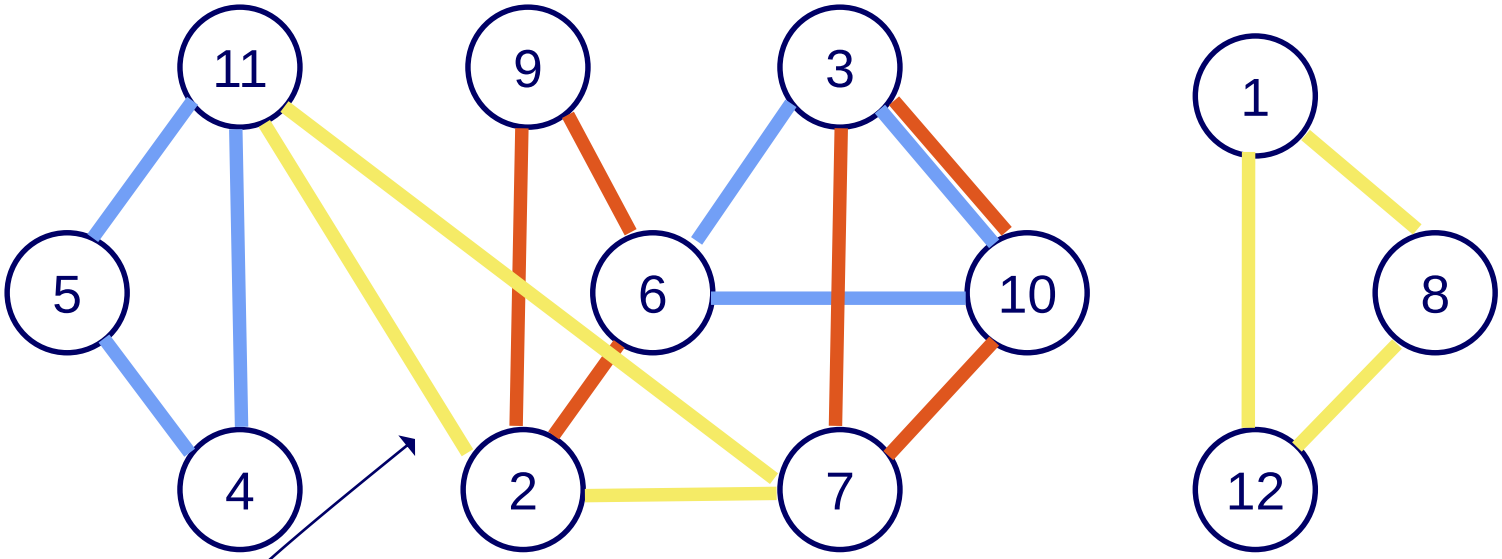


for the given graph:

$$p(\mathbf{b}) = \frac{1}{Z} \psi_1(b_1, b_3, b_5)\, \psi_2(b_1, b_6, b_7)\, \psi_3(b_2, b_4, b_6)$$

The Breakpoint Graph





some prefix of region 2 and some prefix of region 11 should be aligned

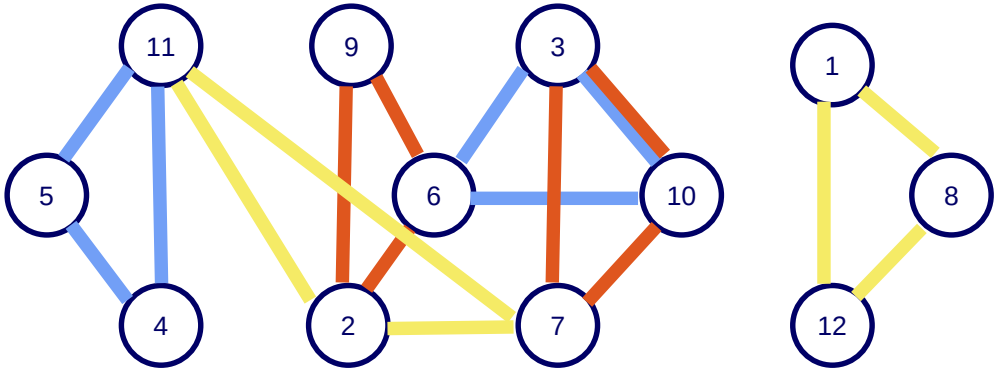
Breakpoint Undirected Graphical Model

- Mercator frames the task of finding breakpoints as an inference task in an undirected graphical model

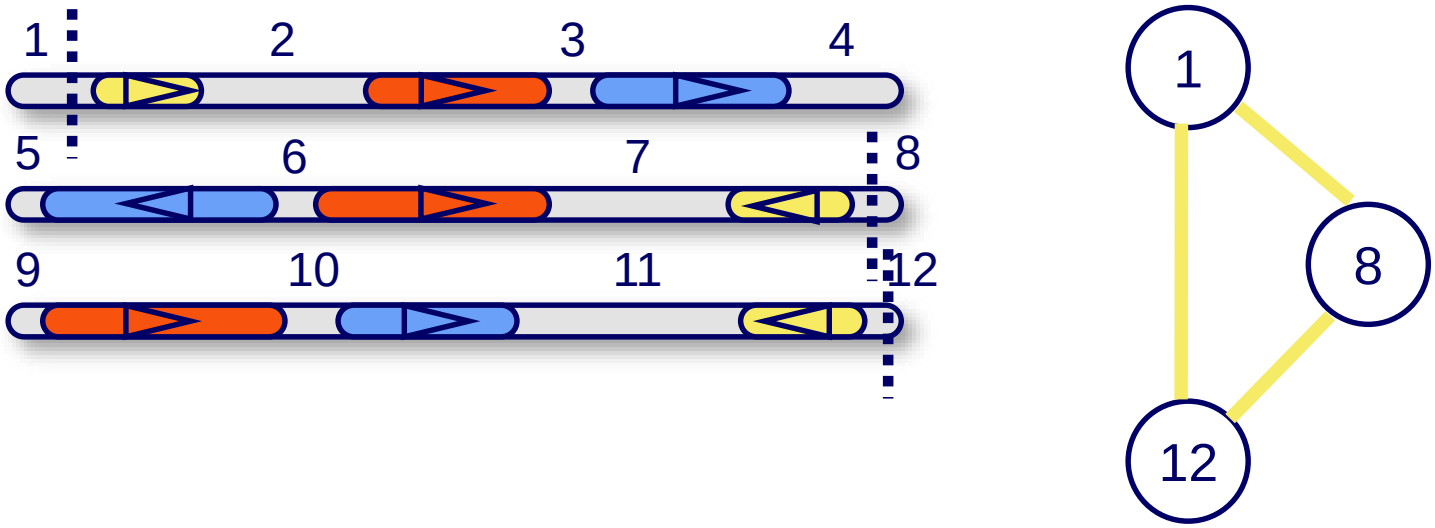
$$p(\mathbf{b}) = \frac{1}{Z} \prod_{C \in cliques} \psi_C(\mathbf{b}_C)$$

configuration of breakpoints

potential function representing score of multiple alignment of sequences in clique C for breakpoints in b



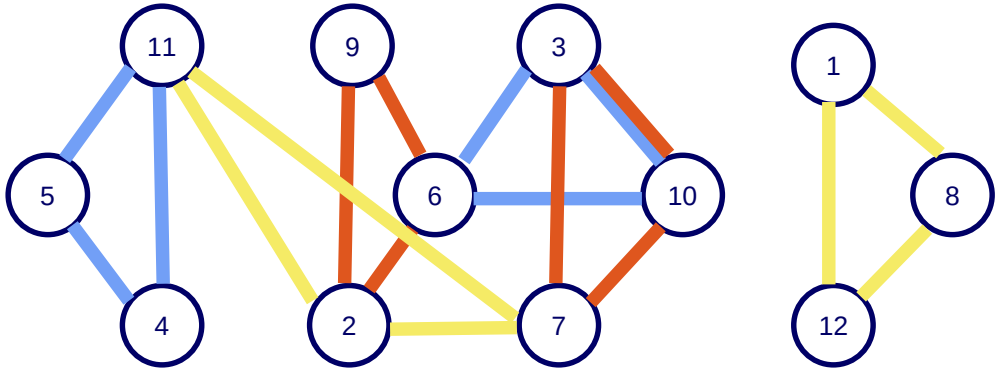
Breakpoint Undirected Graphical Model



- The possible values for a variable indicate the possible coordinates for a breakpoint
- The potential for a clique is a function of the alignment score for the breakpoint regions split at the breakpoints $\mathbf{b}_C$

Breakpoint Undirected Graphical Model

$$p(\mathbf{b}) = \frac{1}{Z} \prod_{C \in cliques} \psi_C(\mathbf{b}_C)$$



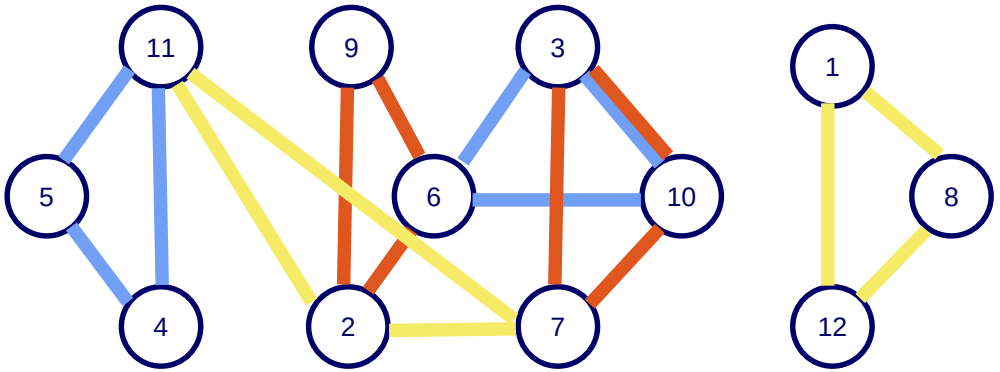
- *Inference task*: find most probable configuration $\mathbf{b}$ of breakpoints

- Not tractable in this case
  - graph has a high degree of connectivity
  - multiple alignment is difficult

- So Mercator uses several heuristics

Making Inference Tractable in Breakpoint Undirected Graphical Model

$$p(\mathbf{b}) = \frac{1}{Z} \prod_{C \in cliques} \psi_C(\mathbf{b}_C)$$



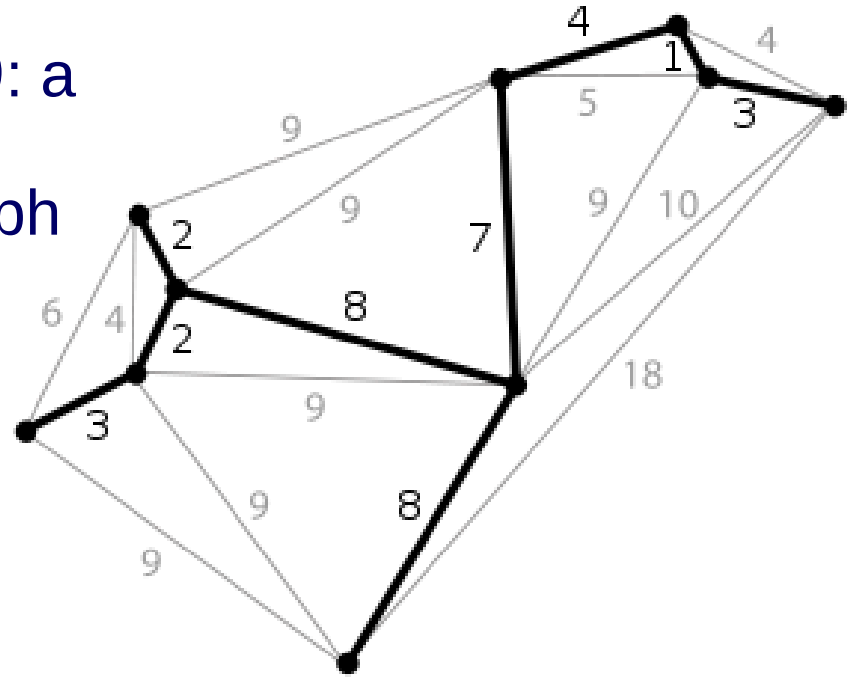
- Assign potentials, based on pairwise alignments, to edges only

$$p(\mathbf{b}) = \frac{1}{Z} \prod_{(i,j) \in edges} \psi_{i,j}(b_i, b_j)$$

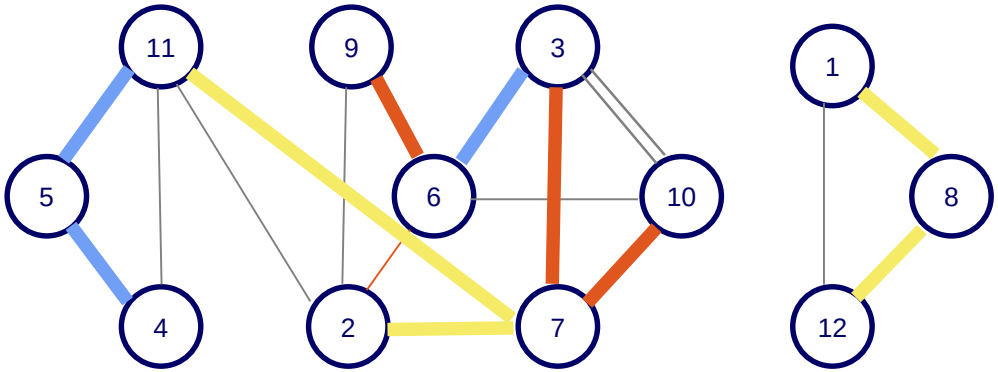
- Eliminate edges by finding a *minimum spanning forest*, where edges are weighted by phylogenetic distance

Minimal Spanning Forest

- *Minimal spanning tree (MST)*: a minimal-weight tree that connects all vertices in a graph



- *Minimal spanning forest*: a set of MSTs, one for each connected component



Breakpoint Finding Algorithm

1. construct breakpoint segment graph
2. weight edges with phylogenetic distances
3. find minimum spanning forest (MSF)
4. perform pairwise alignment for each edge in MSF
5. use alignments to estimate $\psi_{i,j}(b_i, b_j)$
6. perform max-product inference (similar to Viterbi) to find maximizing b_i

Comments on Whole-Genome Alignment Methods

- Employ common strategy
  - find seed matches
  - identify (sequences of) matches to anchor alignment
  - fill in the rest with standard methods (e.g. DP)
- Vary in what they (implicitly) assume about
  - the distance of sequences being compared
  - the prevalence of rearrangements
- Involve a lot of heuristics
  - for efficiency
  - because we don’t know enough to specify a precise objective function (e.g. how should costs should be assigned to various rearrangements)