SEQUENCE ASSEMBLY - MAIN STEPS

1. Compare read-pairwise, to determine potentially overlapping reads.
2. Determine layout, i.e. pattern of overlaps.
3. Determine consensus sequence (i.e. best guess at clone sequence).

Finding Pairwise Matches:

Because reads are overall fairly accurate, finding matches is usually easier than finding subtle sequence similarities. One can assume two overlapping reads will have a perfectly matching word in common of significant length (e.g. 16 bases).

Strategy:
1. Find all pairs of reads having perfectly matching words of length 16 or more.
2. For each pair, check matching words defining a diagonal in an S/W matrix. Enlarge subwindows to a band. Take union of bands, do a S/W search of each band to obtain...

Finding word matches:

Can be done in O(m) time where m is # of matches using suffix trees. In practice, methods which use suffix trees are more memory efficient (e.g. suffix arrays - faster, more preferable).

Simple method:

Scan across pointers to all positions in all reads. Determine order of pointers for increasing distance of pointers to left linguistically. Assign string of pointers to each order.

Sort O by the above partial order. Then matching words are found by examining a subset of O that are near each other in partial sort list.
**CONSTRUCTING THE LAYOUT**

Hardest problem in assembly!

What makes it hard:

1. **Complexity**
   - (1) Repeats in sequence
   - (2) Imperfect match

Are discrepancies "real", or are they due to sequencing errors?

Use error probabilities to help decide! (p/n)

(8) Record each pairwise alignment.

(9) Computing error probs into account.

Compute LSR for H0 vs. H1. Record discord in real vs. H0, record discord in real vs. H1.

(10) Compute LSR for H0 vs. H1, record discord in real vs. H1.

(11) Construct LSR using greedy algorithm, "LSR algorithm".

**Greedy Algorithm for Layout**

**Construction** (simplified)

Data structure for layout: each "contig" consists of a list of reads, each for each read an approximation (�-adjusted or uncorrected) and an LSR (paired vs. paired, unpaired vs. unpaired)

Initially each read forms a separate contig.

Collect read pair alignments.

For each pair consider potential merges between contigs. The selected pair defines an offset, construct new contigs (one from each contig) generated with the offset.

If there are no new LSR scoring pairs found (i.e., the positive LSR scoring pairs span the region of overlap), accept the merge.

**LLR Scores for read pair alignments**

Given an alignment of two reads, wish to compare:

- prob of observed data under H1 (assuming real)

- H1 reads are from 9% similar reads.

Assumptions:

1. Alignment positions are independent of each other.
2. When a base-calling error occurs at a read position, then that error occurs in the two reads disagreeing at that position.

Correlation required to avoid this assumption is very small.

3. Errors in reads are independent of each other.

Consider each position in alignment

Two cases: base calls agree, or they disagree.

Let $e, f$ be error probs for the base calls at each position.

Case 1 (agrees):

$P(\text{Agree} | H_0, \text{contig}) = (1-e)(1-f)$

$P(\text{Agree} | H_1, \text{contig}) = 0.1e(1-f)$

$\text{LLR} = \log\left(\frac{P(\text{Agree} | H_0, \text{contig})}{P(\text{Agree} | H_1, \text{contig})}\right) = \log\left(\frac{1-e}{0.1e}\right)$

Case 2 (disagrees):

$P(\text{Disagree} | H_0) = e+f-ef$

$P(\text{Disagree} | H_1) = 0.05+0.35(0.95-ef)$

$\text{LLR} = \log\left(\frac{P(\text{Disagree} | H_0)}{P(\text{Disagree} | H_1)}\right)$

Full LSR for entire alignment: 1 sym above taken over all aligned positions.

Usually the greedy algorithm gets the layout correct (due to ability of LSR scores to distinguish most repeats). When it fails, the layout is mostly correct, but a small number of false gins. In such cases, break the layout at all weak gins (places where reads have positive LSR scores to other locations in layout). Then consider all possible ways of joining the pieces to take advantage of (or avoid overlap by appropriate criteria).