Lecture 14: DNA Sequencing

Study Chapter 8.9
DNA Sequencing

- Shear DNA into millions of small fragments
- Read 500 – 700 nucleotides at a time from the small fragments (Sanger method)
Fragment Assembly

• Assembles the individual overlapping short fragments (reads) into a genomic sequence
• Shortest Superstring problem from last time is an overly simplified abstraction

• Problems:
  – DNA read error rate of 1% to 3%
  – Can’t separate coding and template strands
  – DNA is full of repeats

• Let’s take a closer look
Traditional DNA Sequencing

DNA

Shake & Break
(by Digestion or Sonication)

DNA fragments

Clone

Vector
Circular genome
(bacterium, plasmid)

Known location
(restriction site)
## Different Types of Vectors

<table>
<thead>
<tr>
<th>VECTOR</th>
<th>Size of insert (bp)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plasmid</td>
<td>2,000 - 10,000</td>
</tr>
<tr>
<td>Cosmid</td>
<td>40,000</td>
</tr>
<tr>
<td>BAC (Bacterial Artificial Chromosome)</td>
<td>70,000 - 300,000</td>
</tr>
<tr>
<td>YAC (Yeast Artificial Chromosome)</td>
<td>&gt; 300,000 Not used much recently</td>
</tr>
</tbody>
</table>
Dideoxy (Sanger) Sequencing

Template strand - g t a a g a c t g t
c a t t c t g a c a

Coding strand -
c a t t c t g a c a

ddT Reaction -
c a t t
c a t t
c a t t c t

ddC Reaction -
c a t t t c
c a t t t c t g a c

ddG Reaction -
c a t t t c t g

ddA Reaction -
c a
c a t t t c t g a c a

dideoxynucleotides - missing a hydroxyl group on the sugar-phosphate backbone

Good for up to 1000 base pairs

10/11/11
Comp 590/Comp 790-87
Fall 2011
Challenging to Read Answers

Electropherogram

Model: 373
Version: 3.0
Software: GeneMapper ID (version 3.7)

Date: 10/11/11
Course: Comp 590/Comp 790-87
Semester: Fall 2011
Reading an Electropherogram

• Issues
  • Noisy start up due to anomalous migration of short fragments that carry bulky dyes
  • Traces become less uniform as run proceeds
  • Large dye responses can overwhelm succeeding lower amplitude responses
  • Occasional mismatches of reaction with template

• Methods for calling the nucleotides: **PHRED**
  – Base calls
  – Maintains quality scores
  – Monitors peak positions
Shotgun Sequencing

cut many times at random (Shotgun)

Get one or two reads from the ends of each segment

~500 bp ~500 bp
Cover region with ~7-fold redundancy
Overlap reads and extend to reconstruct the original genomic region
Read Coverage

Length of genomic segment: $L$
Number of reads: $n$ Coverage $C = n \frac{l}{L}$
Length of each read: $l$

How much coverage is enough?

Lander-Waterman model:
Assuming uniform distribution of reads, $C=10$ results in 1 gapped region per 1,000,000 nucleotides
Challenges in Fragment Assembly

- **Repeats:** A major problem for fragment assembly.

- > 50% of human genome is repeats:
  - over 1 million *Alu* repeats (about 300 bp)
  - about 200,000 LINE repeats (1000 bp and longer)

Green and blue fragments are interchangeable when assembling repetitive DNA.
Types of Genome Assemblies

• De Novo –
  An assembly based entirely on self-consistency or self-similarity of short reads (contigs).

• Comparative –
  Refers an assembly of a genome using the sequence of a close relative as a scaffold or reference. Sometimes called a "template assembly" or "a resequencing"

• Confounding problem for both types: Repeats
Repeat Types

- **Low-Complexity DNA** (e.g. ATATATATACATA…)
- **Microsatellite repeats** $\left(a_1…a_k\right)^N$ where $k \sim 3-6$ (e.g. CAGCAGTAGCAGCACCAG)
- **Transposons/retrotransposons**
  - **SINE** Short Interspersed Nuclear Elements (e.g., $Alu$: ~300 bp long, $10^6$ copies)
  - **LINE** Long Interspersed Nuclear Elements ~500 - 5,000 bp long, 200,000 copies
  - **LTR retroposons** Long Terminal Repeats (~700 bp) at each end
- **Gene Families** genes duplicate & then diverge
- **Segmental duplications** ~very long, very similar copies
Overlap-Layout-Consensus

Assembler programs ARACHNE, PHRAP, CAP, TIGR, CELERA

Common Approach:
Overlap: find potentially overlapping reads

Layout: merge reads into contigs and then combine contigs into supercontigs

Consensus: requires many overlapping reads to derive the DNA sequence and to correct for read errors

..ACGATTACAATAGGTT..
Overlap

- Find the best match between the suffix of one read and the prefix of another (shortest superstring)

- Due to sequencing errors, most algorithms use dynamic programming to find the optimal overlap alignment

- Filter out fragment pairs that do not share a significantly long common substring
Overlapping Reads

- Make histogram all $k$-mers of reads ($k \sim 20-24$)
- Find read-pairs sharing a $k$-mer
- Extend alignment – throw away if not $>95\%$ similar
Histogram Example

v = tagattacacagattattga

- Histogram of 3-mers (18 total)

<table>
<thead>
<tr>
<th></th>
<th>A_2</th>
<th>C_2</th>
<th>G_2</th>
<th>T_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A_1</td>
<td>0:0:0:0</td>
<td>2:0:0:0</td>
<td>2:0:0:0</td>
<td>0:0:0:3</td>
</tr>
<tr>
<td>C_1</td>
<td>0:1:1:0</td>
<td>0:0:0:0</td>
<td>0:0:0:0</td>
<td>0:0:0:0</td>
</tr>
<tr>
<td>G_1</td>
<td>0:0:0:2</td>
<td>0:0:0:0</td>
<td>0:0:0:0</td>
<td>0:0:0:0</td>
</tr>
<tr>
<td>T_1</td>
<td>0:1:1:1</td>
<td>0:0:0:0</td>
<td>1:0:0:0</td>
<td>2:0:1:0</td>
</tr>
</tbody>
</table>
Overlapping Reads and Repeats

• Does this really speed up the process?
• A $k$-mer that appears $N$ times, initiates $N^2$ comparisons
• For an Alu that appears $10^6$ times $\rightarrow 10^{12}$ comparisons – too much
• **How to avoid repeats:**
  Discard all $k$-mers that appear more than $t \times$ Coverage, ($t \sim 10$)
Finding Overlapping Reads

k-mer table makes it easy to create local multiple alignments from the overlapping reads
Finding Overlapping Reads (cont’d)

• Correct errors using multiple alignment and consensus scoring

• Score alignments

• Accept alignments with good scores
Repeats are still a major challenge
Do two aligned fragments really overlap, or are they from two copies of a repeat?
Solution: repeat masking – hide the repeats!!!
Masking results in high rate of misassembly (up to 20%)
Misassembly means a lot more work at the finishing step
2. Merge Reads into Contigs

- Overlap graph:
  - Nodes: reads $r_1 \ldots r_n$
  - Edges: overlaps $(r_i, r_j, \text{shift, orientation, score})$

Note: of course, we don't know the “color” of these nodes.

Reads that come from two regions of the genome (blue and red) that contain the same repeat.
2. Merge Reads into Contigs

We want to merge reads up to potential repeat boundaries
2. Merge Reads into Contigs

- Ignore non-maximal reads
- Merge only maximal reads into contigs

repeat region
2. Merge Reads into Contigs

- Remove transitively inferable overlaps
  - If read $r$ overlaps to the right reads $r_1$, $r_2$, and $r_1$ overlaps $r_2$, then $(r, r_2)$ can be inferred by $(r, r_1)$ and $(r_1, r_2)$
2. Merge Reads into Contigs
2. Merge Reads into Contigs

- Ignore “hanging” reads, when detecting repeat boundaries
Overlap graph after forming contigs

A \hspace{1cm} \text{Target} \hspace{1cm} B \hspace{1cm} X' \hspace{1cm} X'' \hspace{1cm} C

\begin{itemize}
  \item \text{Fragments}
  \item \text{overcollapsed unitig}
  \item \text{repeat boundary}
\end{itemize}

Unitigs:
Gene Myers, 95
Repeats, errors, and contig lengths

• Repeats shorter than read length are easily resolved
  – Read that spans across a repeat disambiguates order of flanking regions

• Repeats with more base pair diffs than sequencing error rate are OK
  – We throw overlaps between two reads in different copies of the repeat

• To make the genome appear less repetitive, try to:
  – Increase read length
  – Decrease sequencing error rate

**Role of error correction:**
Discards up to 98% of single-letter sequencing errors
decreases error rate
⇒ decreases effective repeat content
⇒ increases contig length
2. Merge Reads into Contigs

- Insert non-maximal reads whenever unambiguous
Link Contigs into Supercontigs

- Normal density
- Too dense: Overcollapsed?
- Inconsistent links: Overcollapsed?
Find all links between unique contigs

Connect contigs incrementally, if \( \geq 2 \) links
Fill gaps in supercontigs with paths of overcollapsed contigs
Define $G = (V, E)$

$V := \text{contigs}$

$E := (A, B)$ such that $d(A, B) < C$

**Reason to do so:** Efficiency; full shortest paths cannot be computed
Define T: contigs linked to either A or B

Fill gap between A and B if there is a path in G passing only from contigs in T
Consensus

• A consensus sequence is derived from a profile of the assembled fragments

• A sufficient number of reads is required to ensure a statistically significant consensus

• Reading errors are corrected
Derive Consensus Sequence

Derived multiple alignment from pairwise read alignments

Derive each consensus base by weighted voting
Some Assemblers

• PHRAP
  • Early assembler, widely used, good model of read errors
  • Overlap $O(n^2) \rightarrow$ layout (no mate pairs) $\rightarrow$ consensus

• Celera
  • First assembler to handle large genomes (fly, human, mouse)
  • Overlap $\rightarrow$ layout $\rightarrow$ consensus

• Arachne
  • Public assembler (mouse, several fungi)
  • Overlap $\rightarrow$ layout $\rightarrow$ consensus

• Phusion
  • Overlap $\rightarrow$ clustering $\rightarrow$ PHRAP $\rightarrow$ assemblage $\rightarrow$ consensus

• Euler
  • Indexing $\rightarrow$ Euler graph $\rightarrow$ layout by picking paths $\rightarrow$ consensus
EULER Fragment Assembly

• Traditional “overlap-layout-consensus” technique has a high rate of mis-assembly

• EULER uses the Eulerian Path approach borrowed from the SBH problem

• Fragment assembly without repeat masking can be done in linear time with greater accuracy
Each vertex represents a read from the original sequence. Vertices from repeats are connected to many others.

Find a path visiting every VERTEX exactly once: Hamiltonian path problem
Overlap Graph: Eulerian Approach

Repeat Repeat Repeat

Placing each repeat edge together gives a clear progression of the path through the entire sequence.

Find a path visiting every EDGE exactly once:
Eulerian path problem
Multiple Repeats

Can be easily constructed with any number of repeats
Construction of Repeat Graph

- **Construction of repeat graph from \( k \)-mers:** Emulates an SBH experiment with a huge (virtual) DNA chip.

- **Breaking reads into \( k \)-mers:** Transform sequencing data into virtual DNA chip data.
Construction of Repeat Graph (cont’d)

- Error correction in reads: “consensus first” approach to fragment assembly. Makes reads (almost) error-free BEFORE the assembly even starts.

- Using reads and mate-pairs to simplify the repeat graph (Eulerian Superpath Problem).
Approaches to Fragment Assembly

Find a path visiting every VERTEX exactly once in the OVERLAP graph:

Hamiltonian path problem

NP-complete: algorithms unknown
Approaches to Fragment Assembly (cont’d)

Find a path visiting every EDGE exactly once in the REPEAT graph:

Eulerian path problem

Linear time algorithms are known
Making Repeat Graph Without DNA

• Problem: Construct the repeat graph from a collection of reads.

• Solution: Break the reads into smaller pieces.
Repeat Sequences: Emulating a DNA Chip

- Virtual DNA chip allows the biological problem to be solved within the technological constraints.
Repeat Sequences: Emulating a DNA Chip (cont’d)

• Reads are constructed from an original sequence in lengths that allow biologists a high level of certainty.

• They are then broken again to allow the technology to sequence each within a reasonable array.
Minimizing Errors

- If an error exists in one of the 20-mer reads, the error will be perpetuated among all of the smaller pieces broken from that read.
Minimizing Errors (cont’d)

• However, that error will not be present in the other instances of the 20-mer read.

• So it is possible to eliminate most point mutation errors before reconstructing the original sequence.
Conclusions

• Graph theory is a vital tool for solving biological problems

• Wide range of applications, including sequencing, motif finding, protein networks, and many more
References
