Bacterial Genome Assembly and Annotation

A workshop for the ISCB DC Area Regional Student Group - Summer 2016 Workshop

Jonathan Goodson

What we will cover

- The general methods used to generate nucleic acid sequences
- The idea behind assembly of genomic sequences from different types sequencing information
- How to practically go from sequencing data to semi-completed genomes
- How to begin to utilize assembled genomes to ask biological questions

What won’t happen today

- You will not become an expert
- We will not discuss the best way to do anything
  - Is there a best way?
- We won’t cover the most cutting-edge techniques

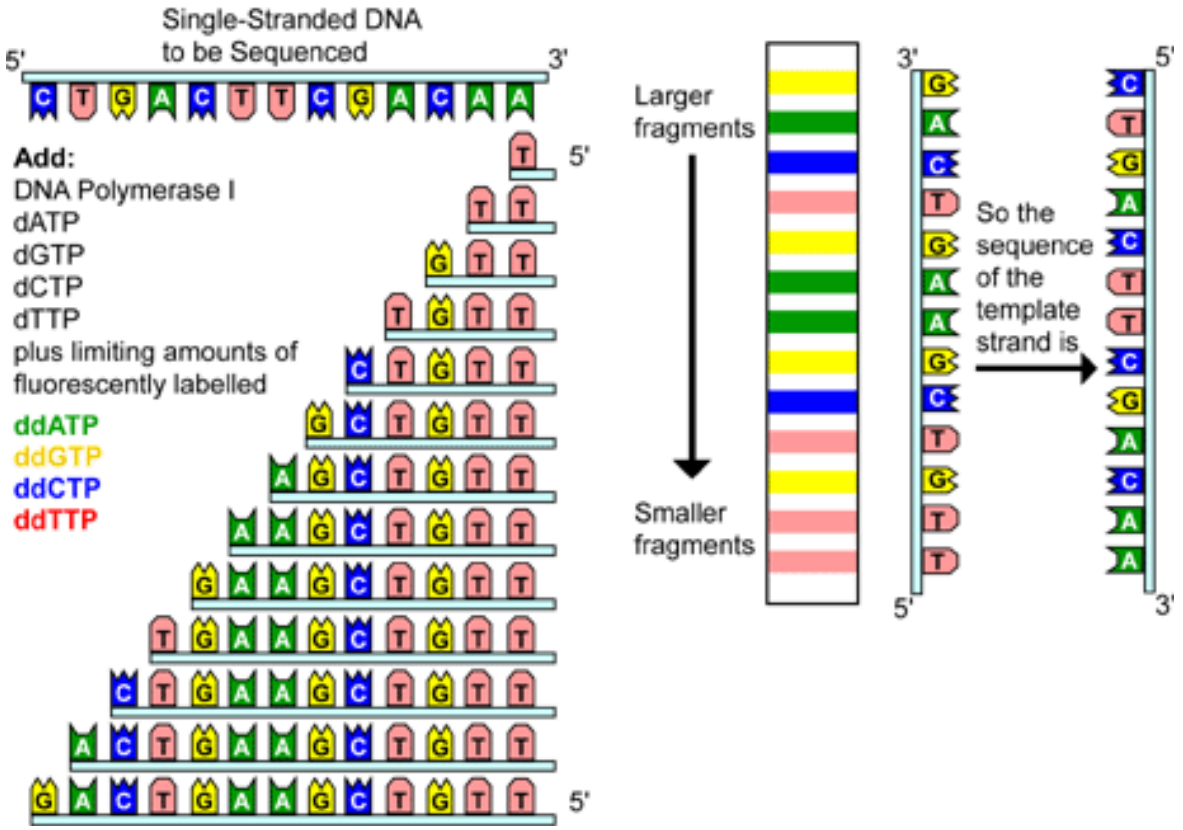
Why would we want to sequence a bacterial genome?

- Environmental isolate doing something interesting (production of molecule, special metabolism..)
- Analysis of important pathogen isolate
- Metagenomic analysis of mixed populations (not discussed today)
- Much easier than larger, more complex genomes!
  - Still not trivial

Background on DNA sequencing

- Modern methods started back in late 70s with Sander chain-termination sequencing
- Modern high-throughput methods include
  - Illumina short-reads (similar concept to Sanger dye chain-termination of a massive scale)
  - 454 pyrosequencing (less common now)
  - IonTorrent (characteristics similar to 454, cheaper, relatively uncommon)
  - Pacific Biosciences (PacBio) very long-read, lower coverage/accuracy
  - Oxford Nanopore single molecule sequencing (similar characteristics to PacBio, very new)

Background on DNA sequencing

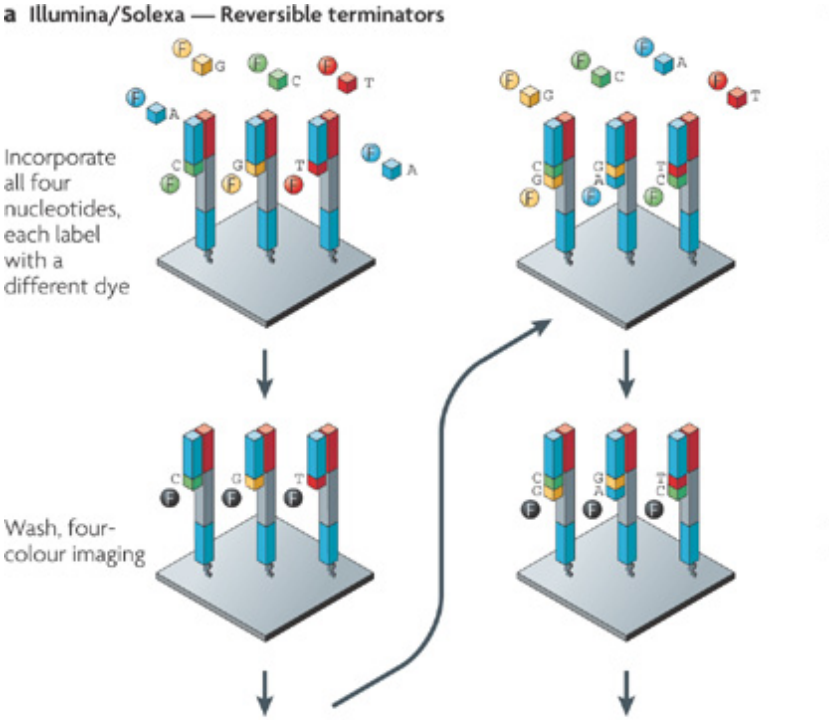


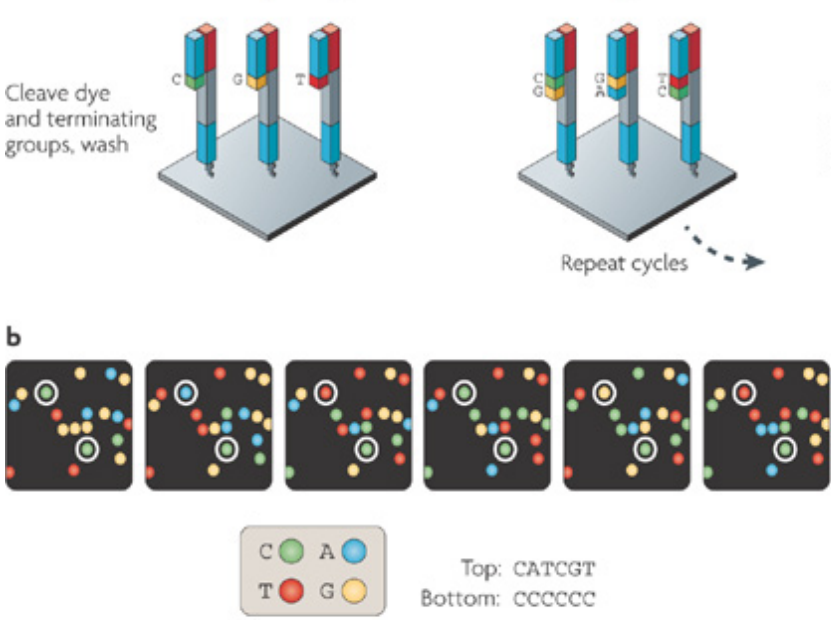
National Human Genome Research Institute

Details about Illumina

- Generates “short” reads (25-300bp)
- Normally “paired” sequencing
  - Sequences both ends of a DNA molecule
  - Can be used to get extra information about DNA fragment size
- Cheapest method, cost for single bacterial genome ranges from $200-$1000 depending on number of genomes sequenced at once
- High-per base accuracy

Details about Illumina





Michael L. Metzker **Nature Reviews Genetics**

Details about Pac Bio

- Generates longest reads possible of any current technology (Nanopore not reliably longer currently)
- 2-25kb reads, can achieve 20kb reliably
- Costs ~$2000 for coverage on one complete bacterial genome (without short-reads)
- Low per-base accuracy
  - Can be overcome via consensus with sufficient coverage

Details about Nanopore

- Developing technology
- Low startup cost (no expensive sequencer)
- Sequences individual molecules by physical measurements as they pass through a very small pore
- Potential for very long reads
- Currently low-quality sequence and very variable read length

Methods of sequence assembly

- Alignment based (Overlap-Layout-Consensus)
  - Find **overlaps** between sequence reads (alignment)
    - Slowest part (Dynamic programming or Suffix Trees)
    - Scales with square of number of reads (Not suitable for Illumina/IonTorrent-type sequencing)
  - Layout stretches of overlapping reads into **contig**uous sequence blocks
    - Based on overlap graph (simplified with removal of transitive edges)
    - Non-branching stretches are contigs
    - Branching areas are unresolvable repeats
  - Choose most likely **consensus** sequence for each block

Overlap-Layout-Consensus

X: CTCGGCCCTAGG

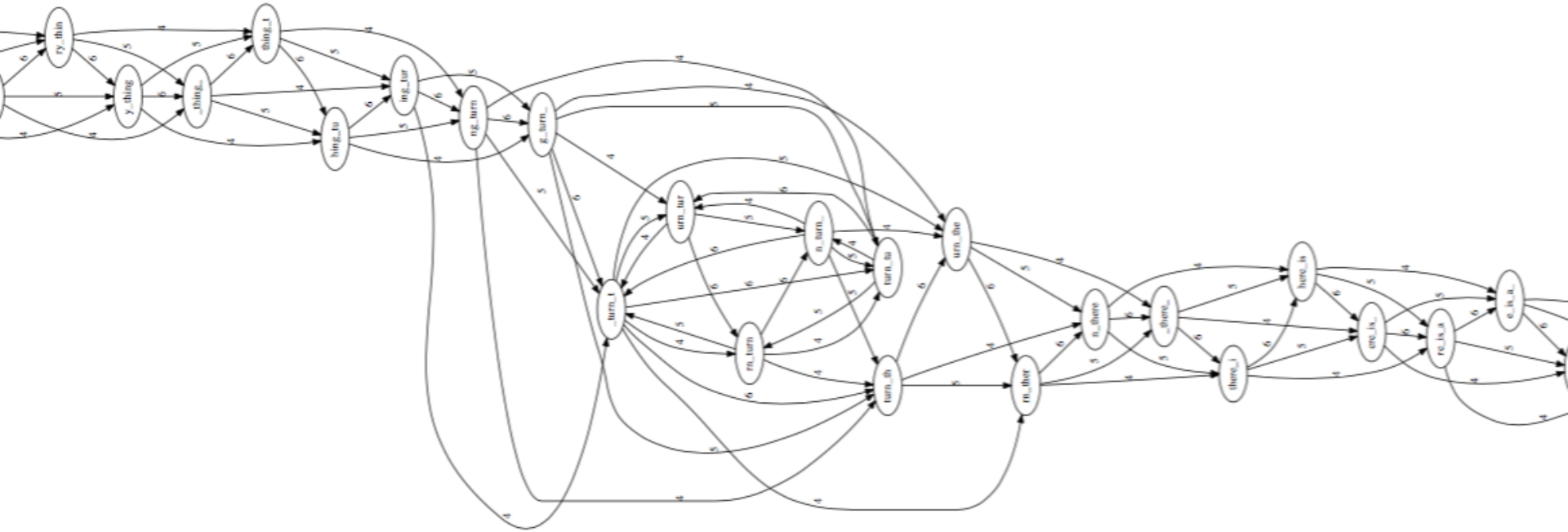
Y: GGCTCTAGGCCC

TAGATTACACAGATTACTGA TTGATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAAACTA
TAG TTACACAGATTATTGACTTCATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA

Take reads that make up a contig and line them up

TAGATTACACAGATTACTGACTTGATGGCGTAA CTA

Take *consensus*, i.e. majority vote



Ben Langmead

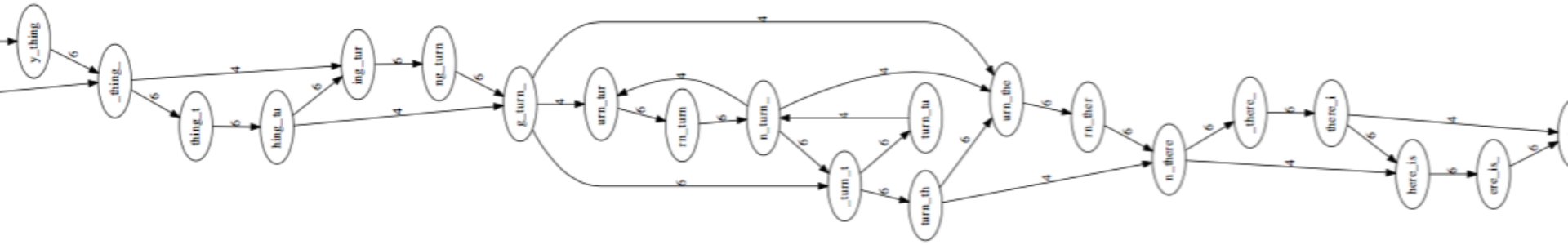
Overlap-Layout-Consensus

```
X: CTCGGCCCTAGG
       |||  |||||
Y:     GGCTCTAGGCCC
```

```
TAGATTACACAGATTACTGA TTGATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAAACTA
TAG TTACACAGATTATTGACTTCATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA
          ↓          ↓  ↓  ↓       ↓
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA
```

Take reads that make up a contig and line them up

Take *consensus*, i.e. majority vote



Ben Langmead

Overlap-Layout-Consensus

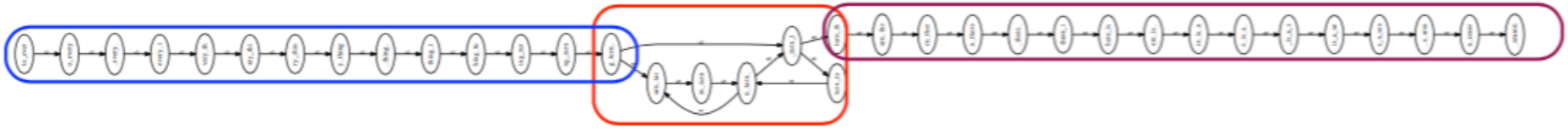
X: CTCGGCCCTAGG
||| |||||
Y: GGCTCTAGGCCC

TAGATTACACAGATTACTGA TTGATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAAACTA
TAG TTACACAGATTATTGACTTCATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA
TAGATTACACAGATTACTGACTTGATGGCGTAA CTA

Take reads that make up a contig and line them up

TAGATTACACAGATTACTGACTTGATGGCGTAA CTA

Take *consensus*, i.e. majority vote



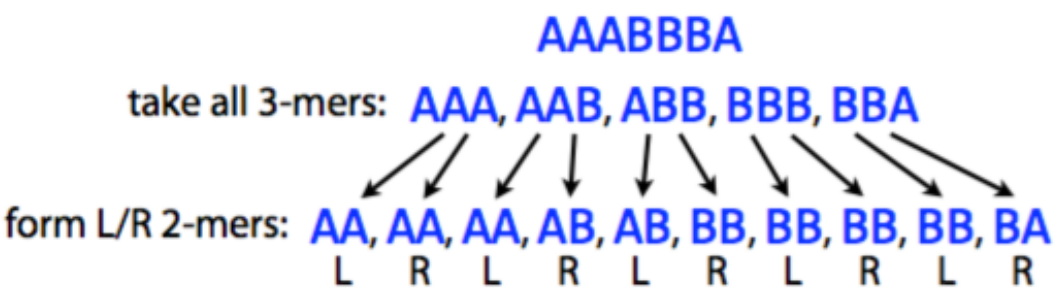
Ben Langmead

Methods of sequence assembly

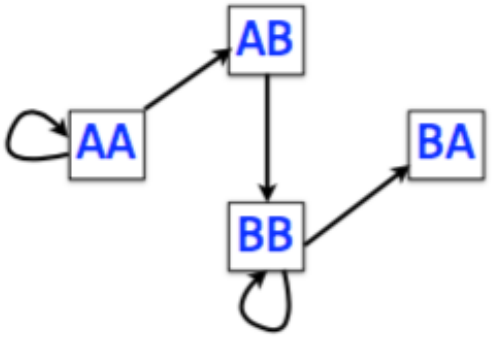
- De Bruijn graph-based
  - Construction of De Bruijn graph based on **k-mers**
  - Assembly performed by finding appropriate **walk** through the graph
  - Scales linearly with number of reads (instead of quadratically)
  - Memory requirement proportional to genome size (not read library size)
  - Cannot resolve repeats as well as overlap graph

De Bruijn Graph Assembly

Take each length-3 input string and split it into two overlapping substrings of length 2. Call these the *left* and *right 2-mers*.



Let 2-mers be nodes in a new graph. Draw a directed edge from each left 2-mer to corresponding right 2-mer:



Each *edge* in this graph corresponds to a length-3 input string

Ben Langmead

Genomic assembly concepts

- Read
- Contig
- Repeat
- K-mer
- Sequence quality
- Error correction
- Scaffolding
- Coverage

Notable Assemblers

- Celera Assembler
  - Alignment based, used for first shotgun sequenced multi-cellular and human genomes
  - Adapted for Nanopore/PacBio long reads
- NGS Assemblers
  - De Bruijn graph - SOAP, ABySS, Velvet, MIRA, **SPAdes**, Newbler, CLC
  - Alignment - HGAP, Canu

Practical Time!

- Assembly of bacterial genome using SPAdes
- SPAdes
  - De Bruijn graph based assembler focued on Illumina data, with some capability to integrate Sanger, 454, PacBio, and Nanopore reads
  - Integrated error correction steps
  - Simple to install/use
  - Top-of-the-line accuracy for bacterial genomes (not designed for larger/complex genoems)

Practical Time!

- Public datasets we will use (E. coli K12 MG1655)
  - Illumina 2x100bp (HiSeq 2500)
    - http://spades.bioinf.spbau.ru/spades_test_datasets/ecoli_mc/
  - Illumina 2x300bp (MiSeq) (10M/89M reads)
    - http://www.illumina.com/systems/miseq/scientific_data.html
  - PacBio P6C2 Chemistry 20kb library (3 flowcells)
    - https://github.com/PacificBiosciences/DevNet/wiki/E.-coli-Bacterial-Assembly
  - Oxford Nanopore R7.3 Chemistry **Error-Corrected**
    - http://schatzlab.cshl.edu/data/nanocorr/

What didn't we do?

- Read filtering/quality trimming
  - Removal of low-quality or contaminating read sequences
  - Many methods, different for every platform
- Optimization of parameters
- Use of Ion Torrent/454

What to do with a draft assembly?

- Check that it is reasonable
- Decide if it is sufficient for your purposes
- **Annotate**
- Compare
- Improve

Genome Annotation

- Coding sequence identification
  - Usually uses something like GeneMark or Glimmer
  - Pretty reliable in bacteria (no introns yay!)
- Comparison of coding sequences to annotated genomes
  - Usually using BLAST or Hmmer
  - Even less reliable than the existing annotation (which can be really bad!)
- Identification of non-coding sequence elements
  - Often searching against databases such as Rfam
- Many pipelines exist, both for transferring annotation from similar genomes (BG-7, RATT) or from scratch (RAST, Prokka, DIYA)
  - I suggest RAST, incredibly simple, fully automated, online

Draft vs Finished Genomes

- We generated draft genomes
  - Few-to-Many contigs
  - Potentially many mistakes
- Finished genomes
  - What you find for well-studied bacteria on GenBank
  - Single sequence for whole chromosome
  - Relatively few mistakes
- Finishing genomes
  - Targeted Sanger sequencing
  - High-coverage PacBio with targeted advanced assembly techniques (HGAP, SMRT Analysis)

Advice for practical usage

- The best experts on assembly are the authors of the software themselves
  - They are often very willing to help with your particular issues
- Data handling/filtering/trimming is very important
- Try many methods and compare
- Decide what you really want before you spend the money on the data
- Generation of material for sequencing has dramatic effects on assembly (and cannot be re-done easily)
- Learn the Unix command line (and maybe Python)

Thank you
for listening!