Genome-scale technologies 2 / Algorithmic and statistical aspects of DNA sequencing

De novo whole genome assembly: part I

Ewa Szczurek
szczurek@mimuw.edu.pl

Instytut Informatyki
Uniwersytet Warszawski

The problem of assembly

DNA copies of the genome

Sequence Reads

Assembled genome

Previous talk

This talk

Definition

Assembly

Set of sequences which best approximate the original sequenced material.

Exercise

Here is a set of reads :

```
TACAGT
  CAGTC
   AGTCA
      CAGA
```

1. What sequence do you think these reads come from?

Solution

Here is a set of reads :

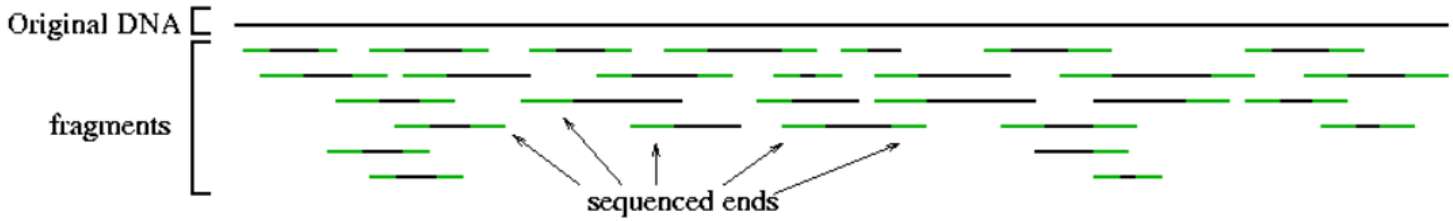
```
TACAGT
  CAGTC
   AGTCA
      CAGA
```

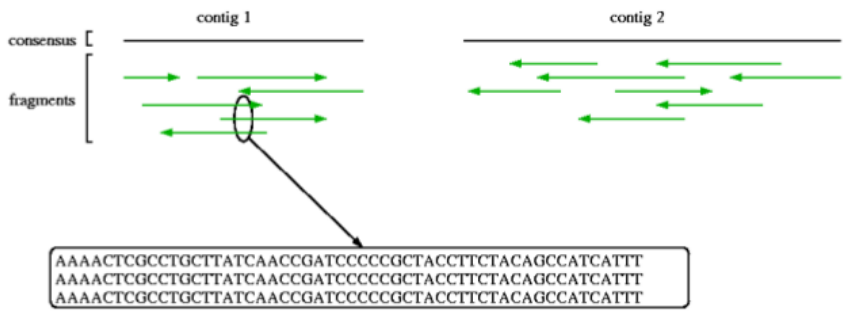
1. What sequence do you think these reads come from?
   TACAGTCAGA

Important notions in this lecture

Read	Any sequence that comes out of the sequencer,
Paired	$read_1$, gap$\leq$ 500 bp, $read_2$,
Mate	$read_1$, gap$\leq$ 1 Kbp, $read_2$,
Single	Unpaired read,
Contig	Gap-less assembled sequence,
Scaffold	Sequence which may contain gaps (filled with N).

From reads to scaffolds





Contigs

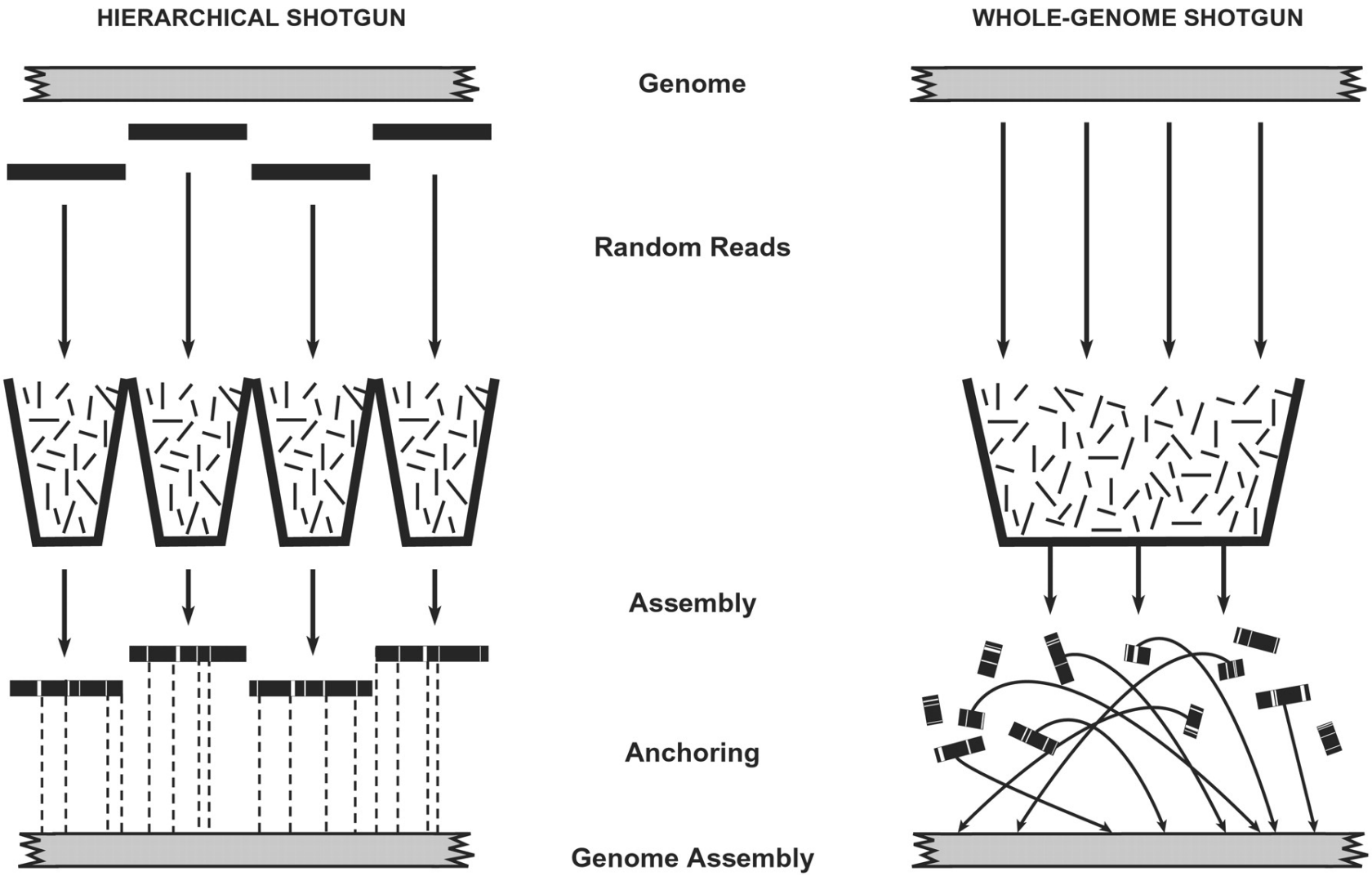
Scaffolding

Sequencing the whole genome: technologies

- Hierarchical shotgun + Sanger sequencing (HGP)
- Shotgun + Sanger sequencing (Celera Genomics)
- Now: shotgun + NGS

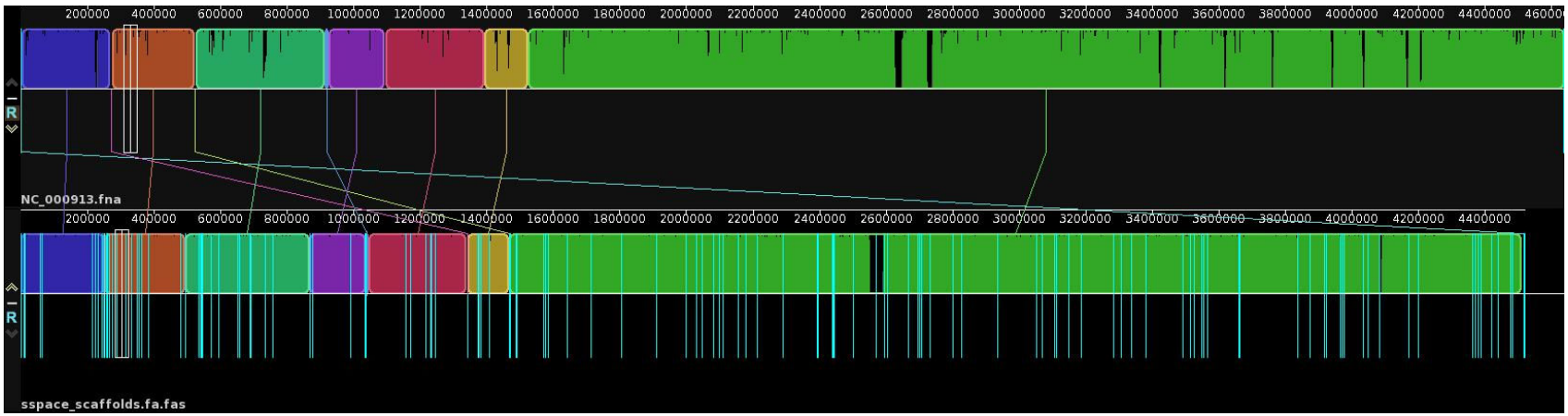
In this lecture we will focus on assembly methods which use longer and less error-prone reads, like those coming from Sanger sequencing.

Sequencing the whole genome: HGP vs Celera



Sequencing the whole genome: in practice

Example of a reference genome (top), and an assembly aligned to it (bottom, sequences separated by blue lines).



The aligned assembly is :

- smaller than the reference,
- fragmented

Algorithms for *de novo* genome assembly

A string S is an ordered list of characters. Characters are drawn from an alphabet Σ. Nucleic acid alphabet: $\{A, C, G, T\}$

1. Before we start: string indexing for efficient dealing with whole genome sequencing data
   - DNA fragment $\Leftrightarrow$ a string over the alphabet $\{A, C, G, T\}$
   - Indexing: preprocessing the string so that there is efficient access to its substrings
   - Suffix tries
   - Suffix trees
2. Shortest common superstring (SCS) approach
3. Overlap Layout Consensus (OLC) approach
4. de Bruijn graph (DBG) assembly