Genome and transcriptome assembly

Lecture Outline

- Background and some mathematical history
- Results of genome assembly & quality metrics
- Improving genome assemblies
- De novo transcriptome assembly

Data

- Short-read sequences

-Typically paired end and Mate-pair sequences used

- Recent trend to link data generation to assembly strategy

    - e.g. ALLPATHS assembler: high-coverage short-insert paired-end + medium coverage medium insert mate-pair + low-coverage long insert mate-pair

Alternative assembly paradigms

1) Greedy

Makes locally optimal choices (e.g. join pairs of reads with best overlap). Examples: phrap, TIGR Assembler

2) Overlap-layout-consensus graph

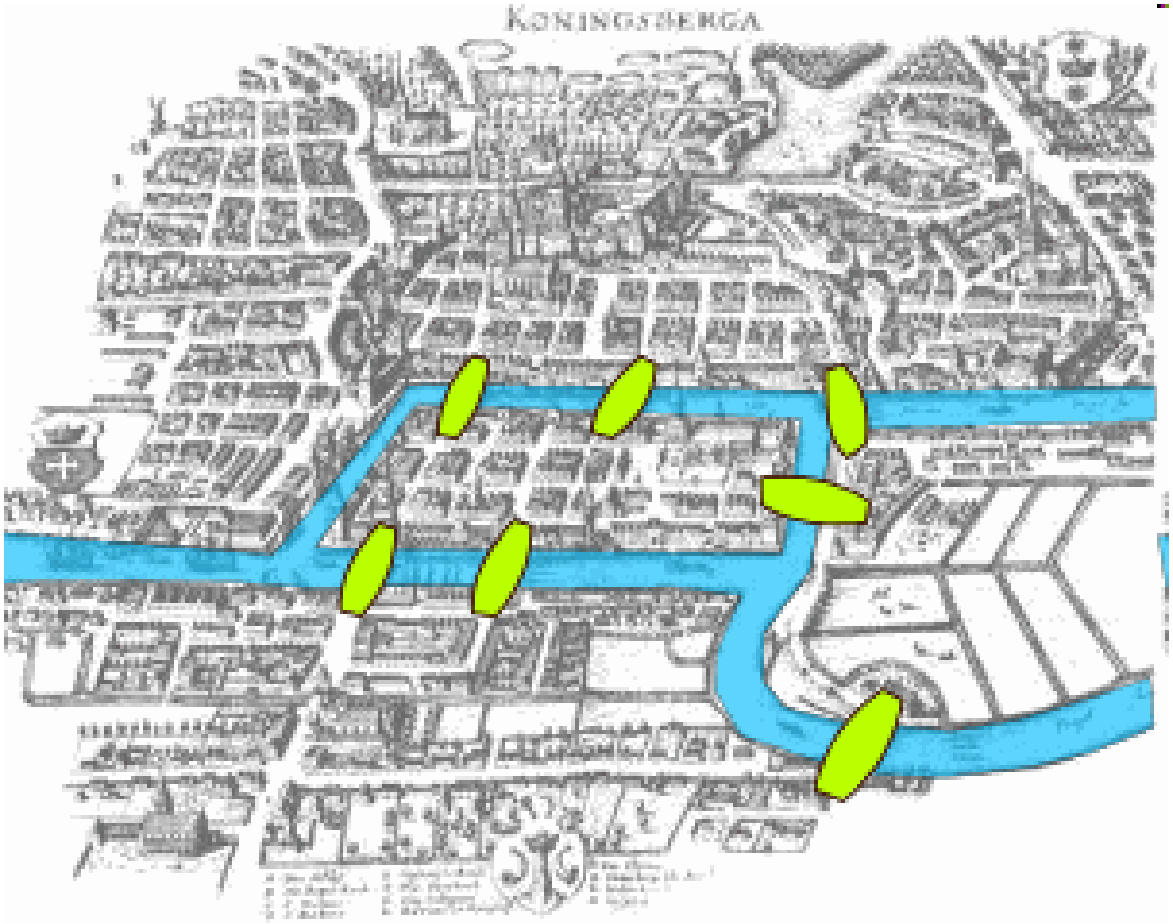
Construct graph with nodes consisting of reads and edges linking reads that overlap sufficiently well. Memory requirements an issue. E.g. Celera Assembler

3) de Bruijn graph

Construct graph with nodes representing k-mers and edges between k-mers that overlap by k-1 bases. Methods are time and memory efficient but get into trouble with high sequence error rates. Error correction methods are used. E.g. Velvet, SOAPdenovo, ALLPATHS

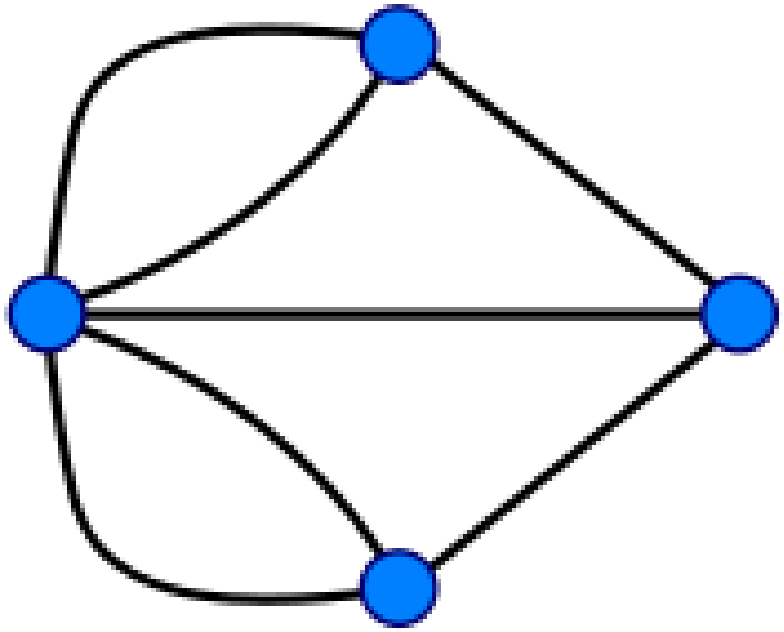
Source: Nagarajan & Pop, 2013

What is a de Bruijn graph? We start with Leonhard Euler and the bridges of Konigsberg problem (solved in 1735)

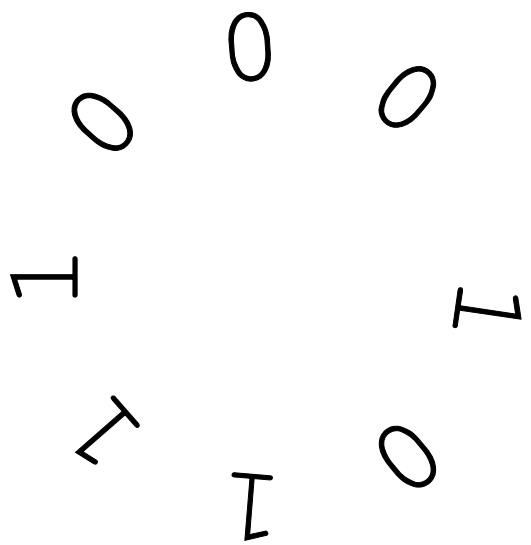


Leonhard Euler and the bridges of Konigsberg problem (solved in 1735)

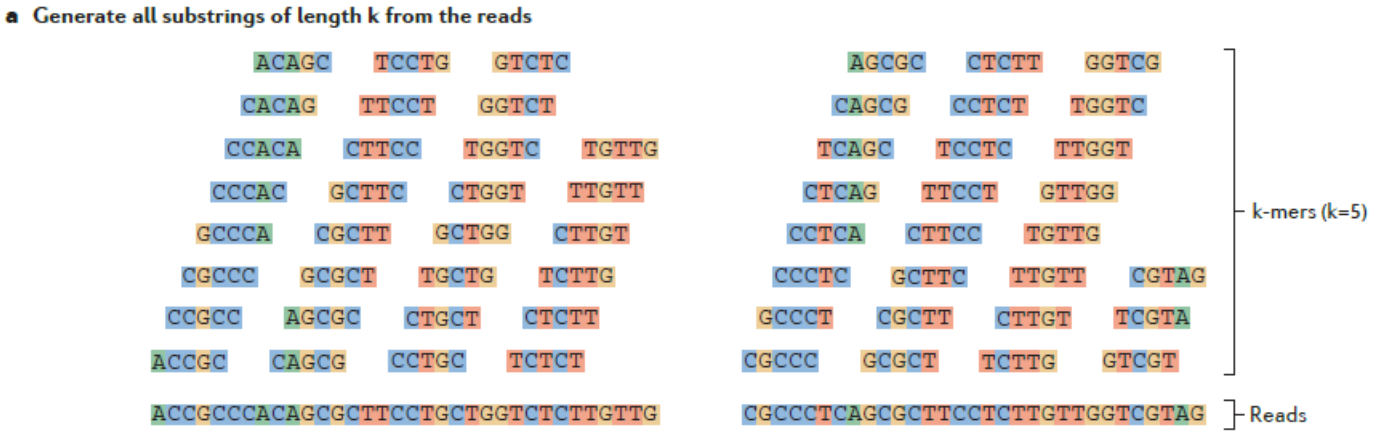
A graph

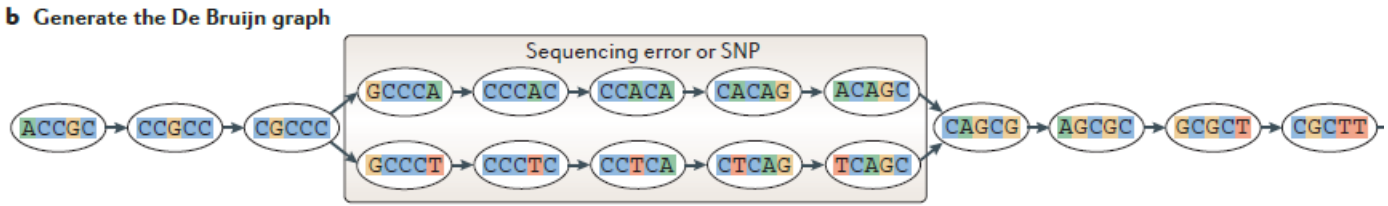


Nicolas de Bruijn: What's the smallest circular string of 1's and 0's such that all the substrings of length *k* are found in the string. E.g. below for *k* = 4

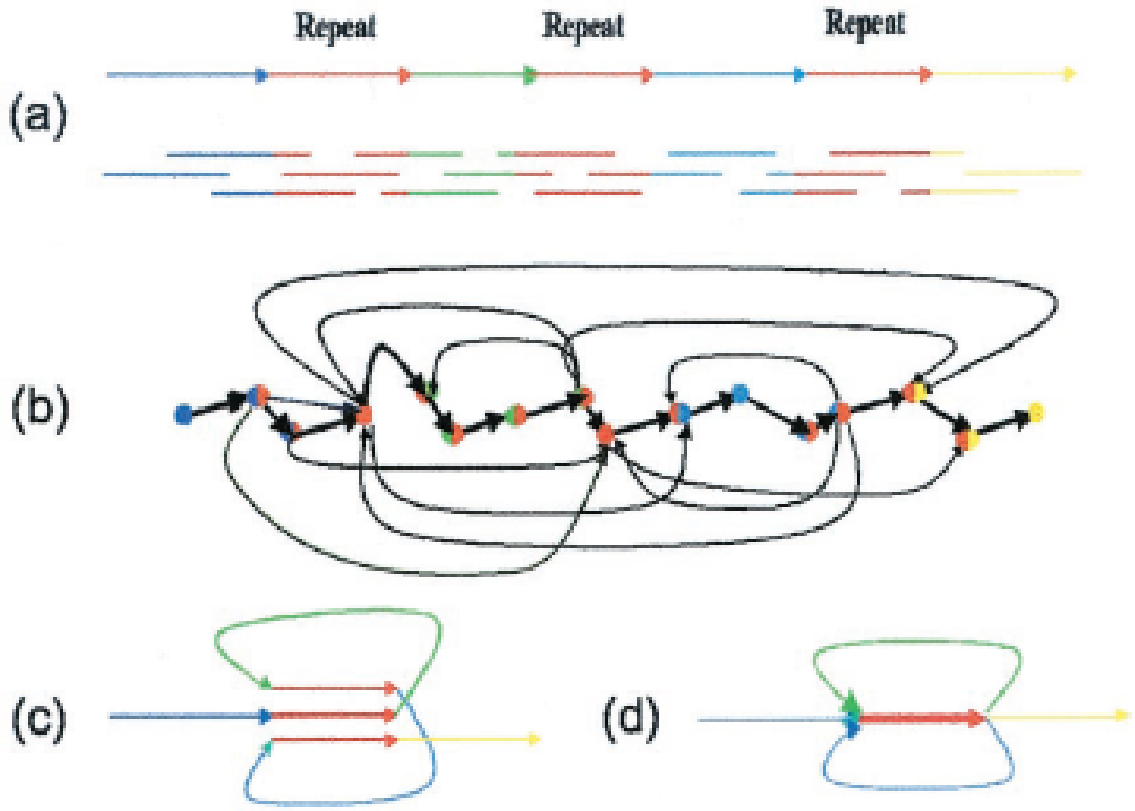


De Brujn graph for sequence assembly



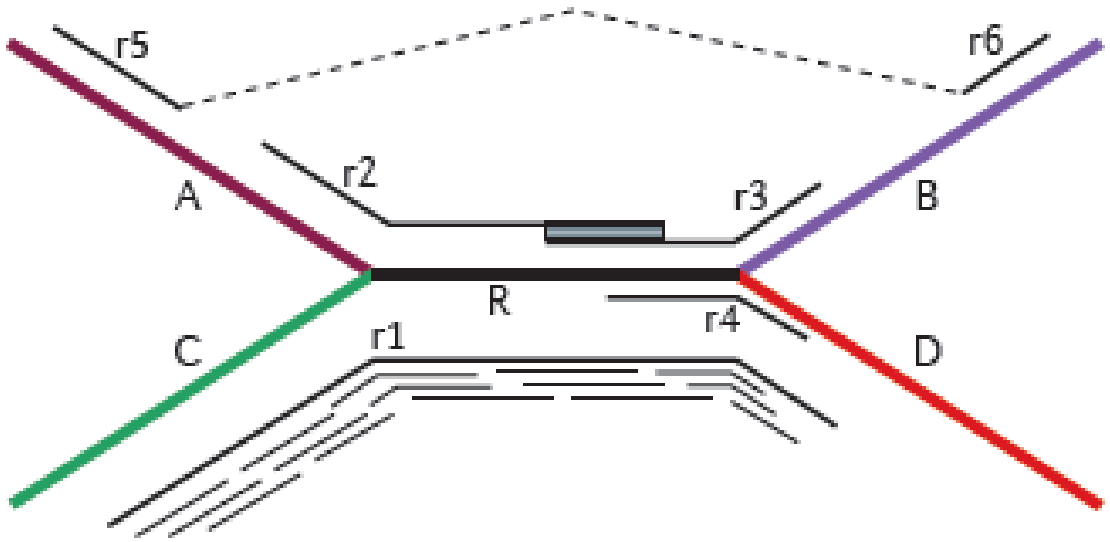


Comparison of overlap-layout-consensus and de Bruijn approaches



Pevzner et al. PNAS, 2001

Repetitive regions can break assemblies. Consider unique genome below with unique regions A, C, B, D and repeat region R (this is a ‘repeat graph’)



The long read r1 can infer that the correct assemblies are ARB and CRD. But this cannot be inferred from the short reads (e.g. r2,r3). Long-insert mate-pair reads (r5,r6) can also resolve the correct ordering of the regions.

Source: Nagarajan & Pop, 2013

Results of Genome Assembly

Contigs: Contiguous stretches of DNA sequence, fully assembled. Typically in fasta format (although a new format on the way for genome assemblies – fastg)

Scaffolds: Contigs can be assembled into scaffolds

Assembly Quality:

Contiguity, Completeness, Accuracy

Contiguity: Contig N50; Scaffold N50

N50: Contig/scaffold size, x, such that 50% of the genome is in contigs/scaffolds > x

Accuracy is harder to measure – e.g. from comparison to known well-assembled genomic regions

Genome Assembly improvement

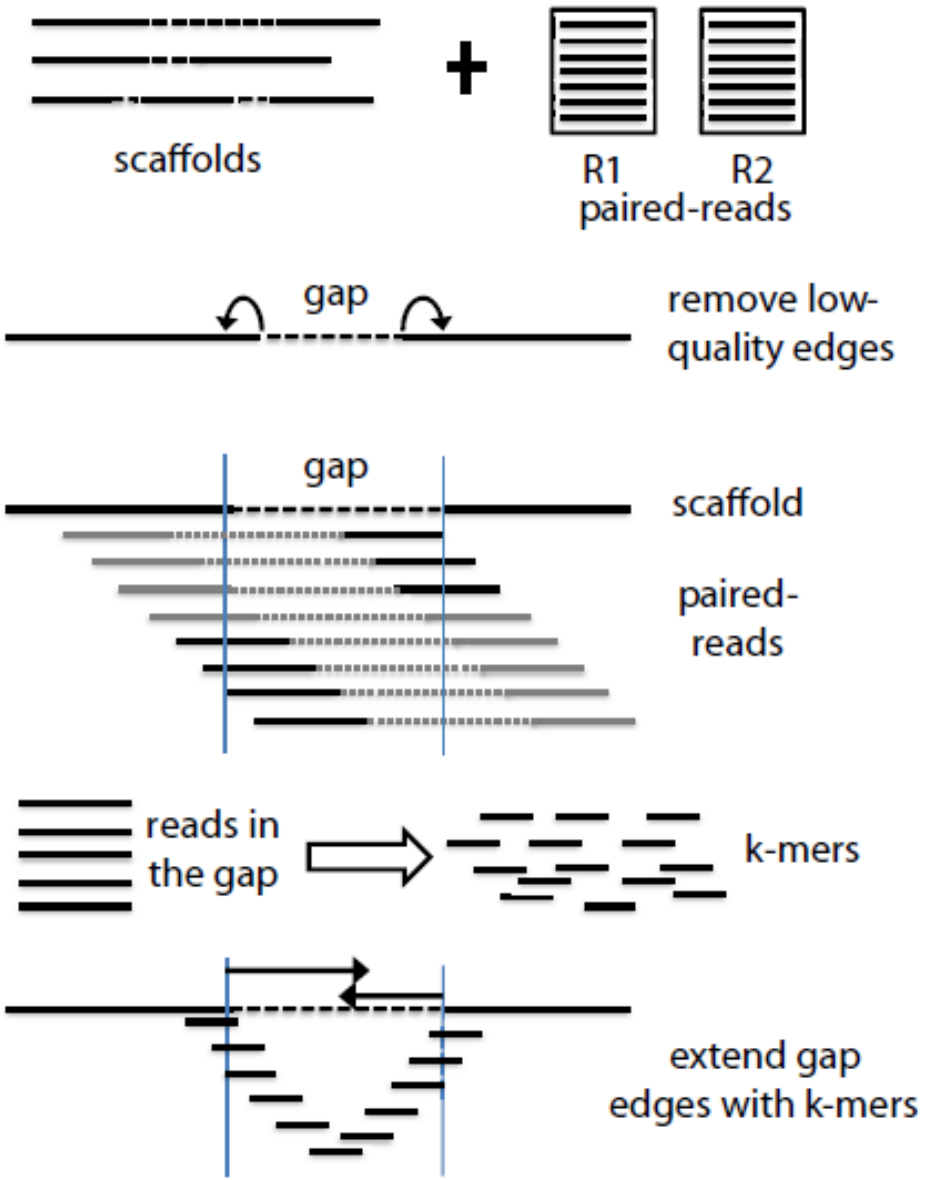
Objective: Close gaps in scaffolds, improve contig/scaffold ordering

e.g. GapFiller; Pagit (mainly for prokaryotes)

Pagit

1. ABACAS: ordering contigs using a related reference genome
2. IMAGE: gap filler
3. iCORN: iterative correction of errors in assembly
4. RATT: transfer of annotation information from a related genome

GapFiller



Source: Boetzer & Pirivano

De novo Transcriptome Assembly

- Apply assembly methods to the RNA-seq data directly
- Align reads to the assembled transcriptome

e.g. Trinity; TransAbyss

What are the advantages/disadvantages?

Advantages

- Does not require reference genome

- Not affected by errors in reference genome (most genome assemblies quite inaccurate)

Disadvantages

- Computationally much more intensive than alignment to a reference genome (requires high memory)

- Requires much greater sequence coverage than reference alignment strategy (coverage particularly an issue for low expression genes)

- Artefacts not removed (e.g. contaminants will not align to reference genome, but will be included in de novo transcriptome assembly)

- All annotation work remains to be done

- Miss assembly, e.g. resulting from paralogues could result in unpredictable errors