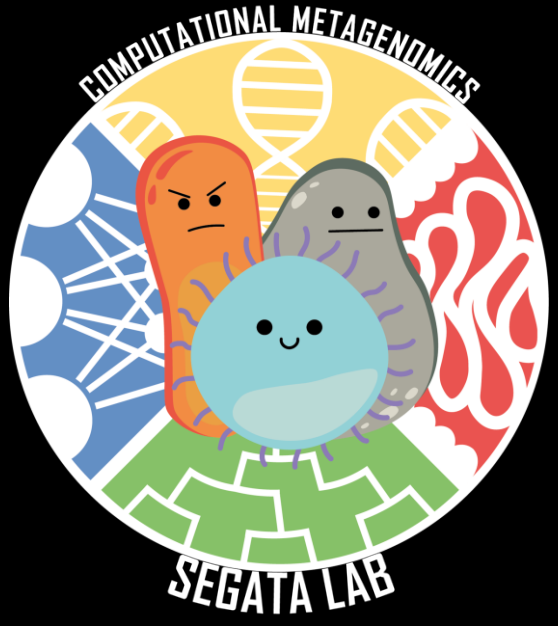
QCB 2021/2022
Computational Microbial Genomics

06: Sequence Assembly



@cibiocm
@nsegata

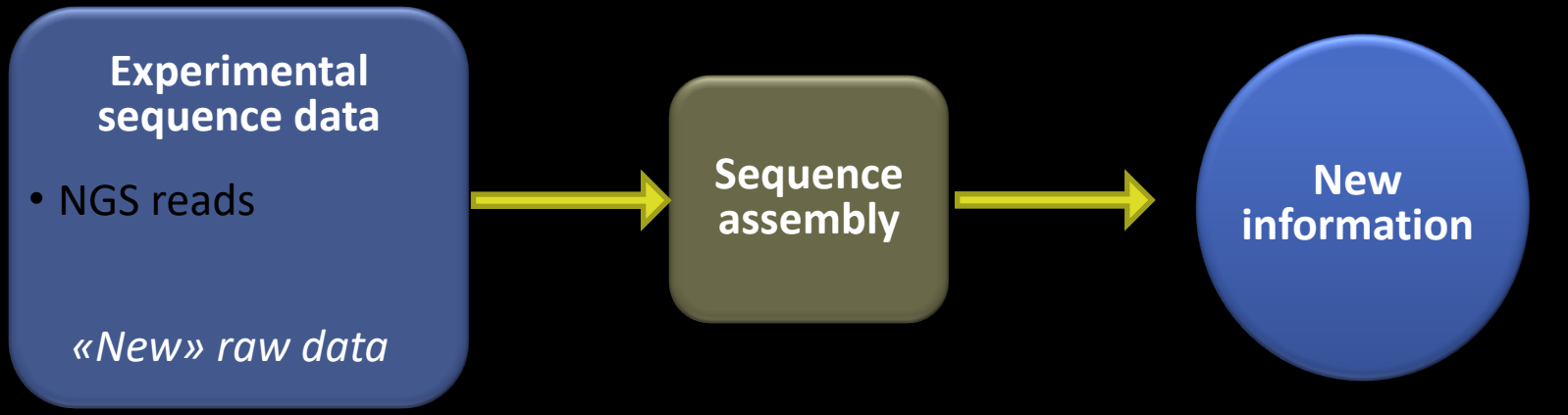
Nicola Segata

UNIVERSITY OF TRENTO

Department of Cellular, Computational, and Integrative Biology (CIBIO)
Trento, Italy

CIBIO

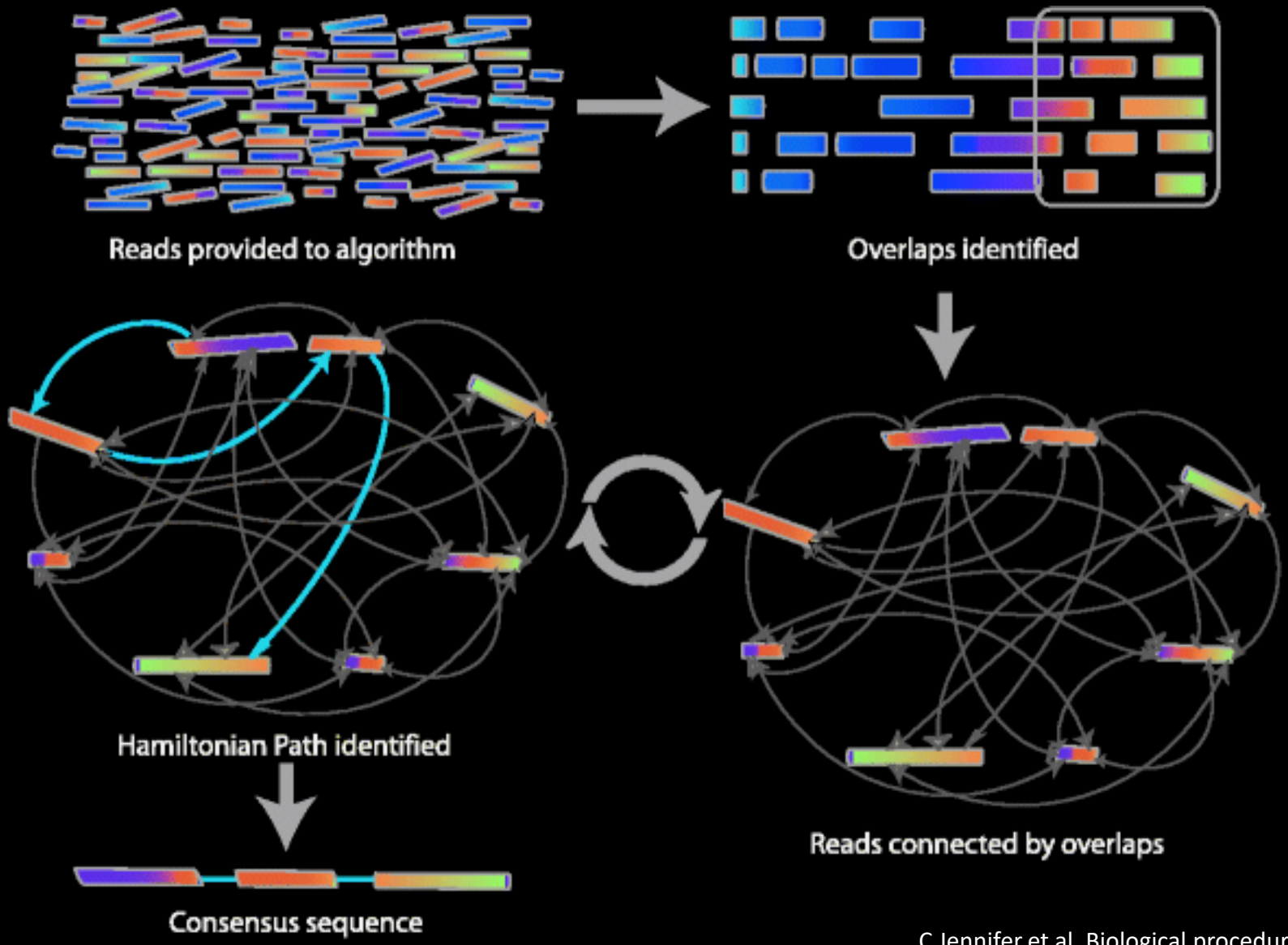
Sequence assembly



e.g. Illumina DNA-seq of a bacterial strain

e.g. the genome of a bacterium

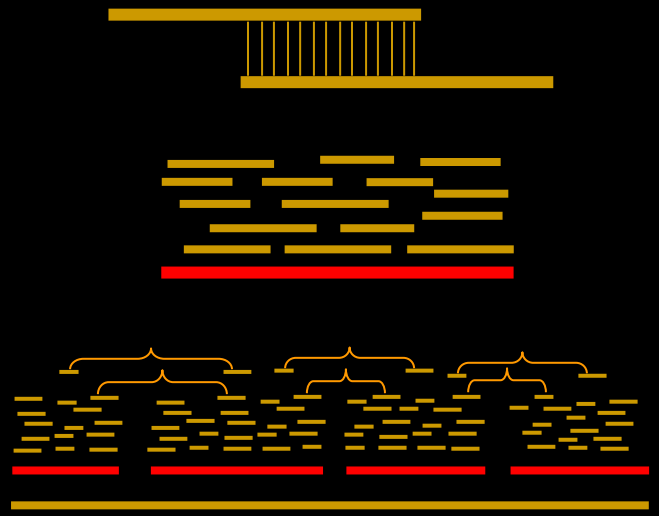
Sequence assembly: no “easy” approaches!



C Jennifer et al. Biological procedures (2009): 52-78.

Sequence assembly: the general pipeline

1. Find overlapping reads
2. Merge some “good” (pairs of) reads into longer contigs
3. Link contigs to form scaffolds (a.k.a. supercontigs)
4. Derive consensus sequence



..ACGATTACAATAGGTT..

- **CONTIG:** a contiguous sequence formed by several overlapping reads with no gaps
- **SCAFFOLD (SUPERCONTIG):** an ordered and oriented set of contigs, usually by exploiting mate pairs
- **CONSENSUS SEQUENCE (GENOME, CHROMOSOME):** a set of possibly unordered scaffolds from the same organism

On the feasibility of sequence assembly

PERSPECTIVE

Human Whole-Genome Shotgun Sequencing

James L. Weber[1,3] and Eugene W. Myers[2]

[1]Center for Medical Genetics, Marshfield Medical Research Foundation, Marshfield, Wisconsin 54449;
[2]Department of Computer Science, University of Arizona, Tucson, Arizona 85721

Sequencing the human genome with shotgun sequencing + assembly is the only feasible strategy

Weber, James L., and Eugene W. Myers. "Human whole-genome shotgun sequencing." Genome Research 7.5 (1997): 401-409.

PERSPECTIVE

Against a Whole-Genome Shotgun

Philip Green[1]

Department of Molecular Biotechnology, University of Washington, Seattle, Washington 98195

Computational assembly of shotgun sequencing data is simply unfeasible, and a bad idea anyway

Green, Philip. "Against a whole-genome shotgun." Genome Research 7.5 (1997): 410-417.

They were both right!

(…well, Weber and Myers were a bit more right from the practical viewpoint…)

Exercise!

- 3 groups
- One piece of English text to assemble
- Groups are given sets of reads
  - Differences in read length
  - Differences in coverage
  - Difference in error rates,
- Time: ~45 mins
- Goal: assemble as much text as possible (possibly the full text!)
- You can (should!) use pens and paper
- You cannot use computers, google, smartphones

The solution

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness, it was the spring of hope, it was the winter of despair, we had everything before us, we had nothing before us, we were all going direct to heaven, we were all going direct the other way - in short, the period was so far like the present period, that some of its noisiest authorities insisted on its being received, for good or for evil, in the superlative degree of comparison only.

Charles Dickens, «A Tale of Two Cities»

Results from last year

	Group 1	Group 2	Group 3
Coverage	5	5	4
Read length	10 PE	10	14
Mut. Rate (every 1k)	20	20	50
Number of corr. words	?	?	?
Number of sentences	?	?	?
Reconstruction perc.	?	?	?
Time (mins)	45	45	45

Results from last year

	Group 1	Group 2	Group 3
Coverage	5	5	4
Read length	10 PE	10	14
Mut. Rate (every 1k)	20	20	50
Number of corr. words	20/120	21/120	73/120
Number of sentences	2	3	7
Reconstruction perc.	16.6	17.5	60.8
Time (mins)	45	45	45

- Error correction?
- Short vs long reads
- High vs low coverage
- Knowing English helped?
- Word-based vs overlapping based approaches

Results from two years ago

	Group 1	Group 2	Group 3	Group 4	Group 5
Coverage	5	10	10	4	3
Read length	10	10	10	14	12
Mut. Rate (every 1k)	20	20	40	50	20
Number of corr. words	?	?	?	?	?
Number of sentences	?	?	?	?	?
Reconstruction perc.	?	?	?	?	?
Time (mins)	50	50	50	50	50

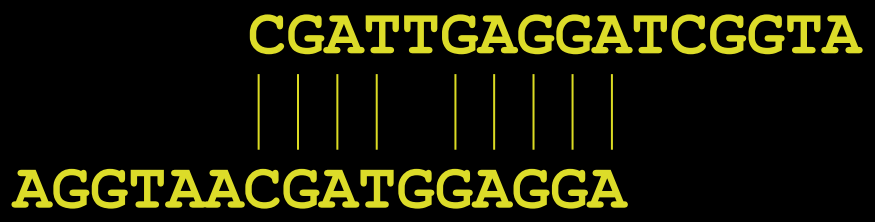
Results from two years ago

	Group 1	Group 2	Group 3	Group 4	Group 5
Coverage	5	10	10	4	3
Read length	10	10	10	14	12
Mut. Rate (every 1k)	20	20	40	50	20
Number of corr. words	61/120	63/120	70/120	85/120	51
Number of sentences	13	13	13	13	11
Reconstruction perc.	~50%	~50%	~55%	65%	~45%
Time (mins)	50	50	50	50	50

- Error correction?
- Short vs long reads
- High vs low coverage
- Knowing English helped?
- Word-based vs overlapping based approaches

Merging overlapping reads

Basic principle: the strongest the similarity between the end of one read and the beginning of another the highest the likelihood the reads are coming from the same overlapping sequence region.

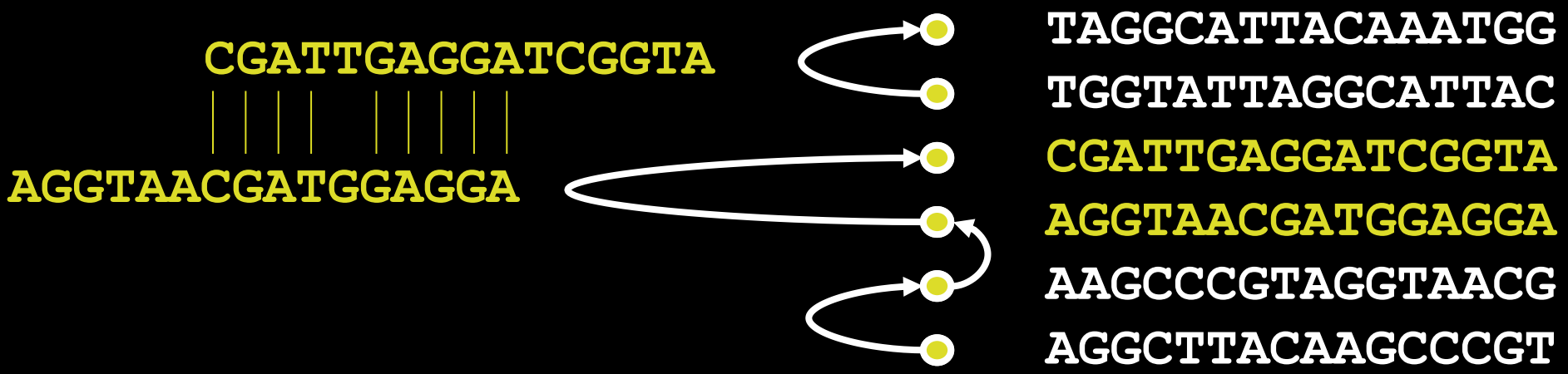


TTGATCCAAGGTAACGATGGAGGATCGGTATTAGATTACCGC

Reasons for not perfect read overlapping

1. Sequencing error/noise
2. Assembly error: reads are coming from different regions of the genome
3. The genome is diploid

Mapping overlapping reads in the dataset



In this way, we are building a graph.

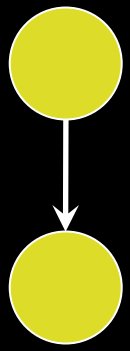
A definition of graph

A Graph $G=(V,E)$ is a mathematical object consisting of an ordered pair of vertices (V) and edges (E)

$V = \{v_1, v_2, \ldots, v_n\}$ (Vertices or Nodes)

$E = \{(v_i, v_j), \ldots, (v_x, v_z)\}$ *with* $v_i, v_j, \ldots, v_x, v_z \in V$

Directed graphs are graphs in which the order in the vertice pairs does matter



regulatory interactions (protein-DNA)

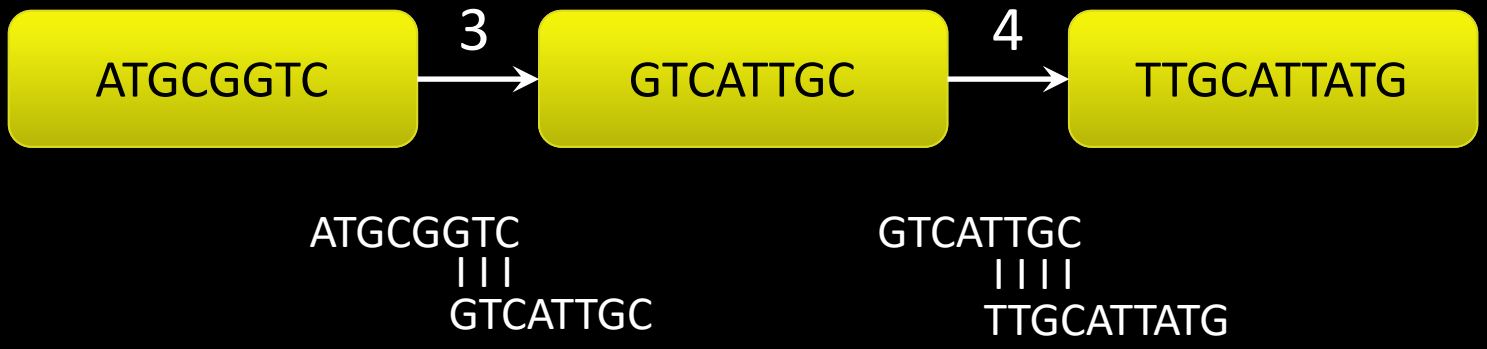
functional complex
B is a substrate of A
(protein-protein)

Assembly graphs

Assembly: overlap graphs

V = {*a :* ATGCGGTC, *b :* GTCATTGC, *c :* TTGCATTATG}

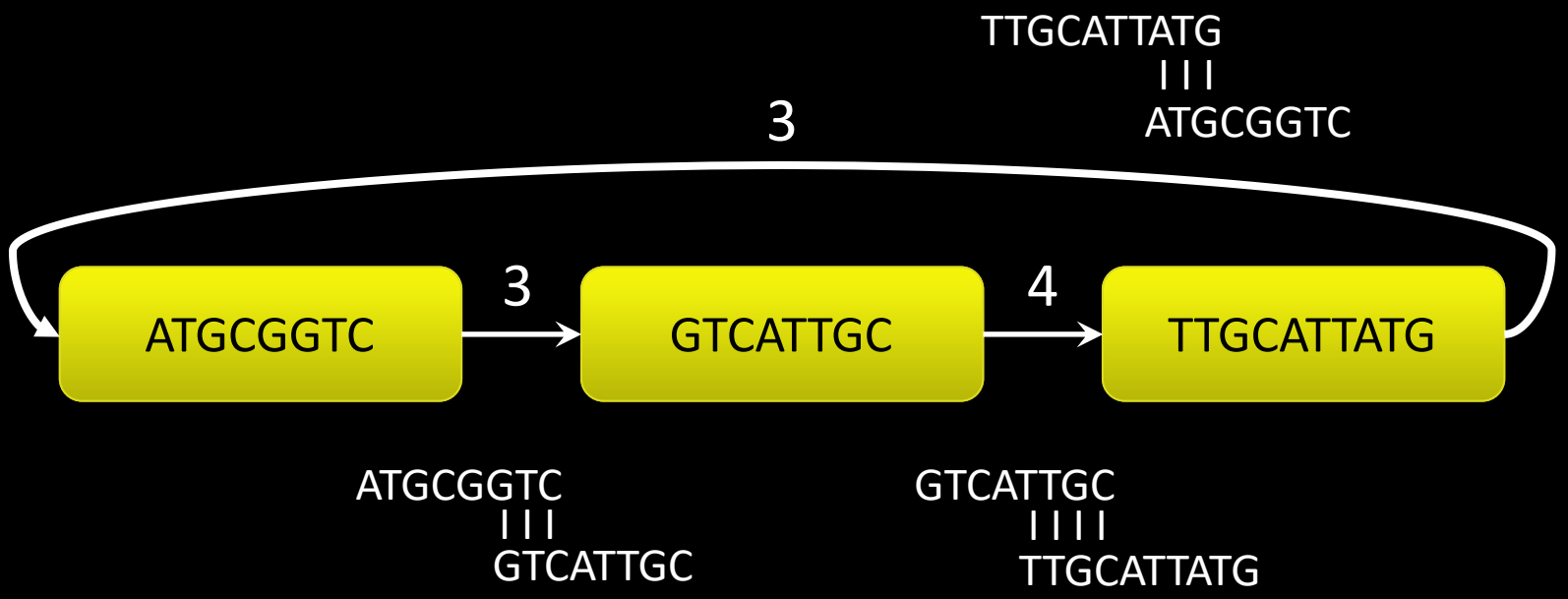
E = {(a,b),(b,c)}



Notice the weights…

Overlap graphs can contain cycles

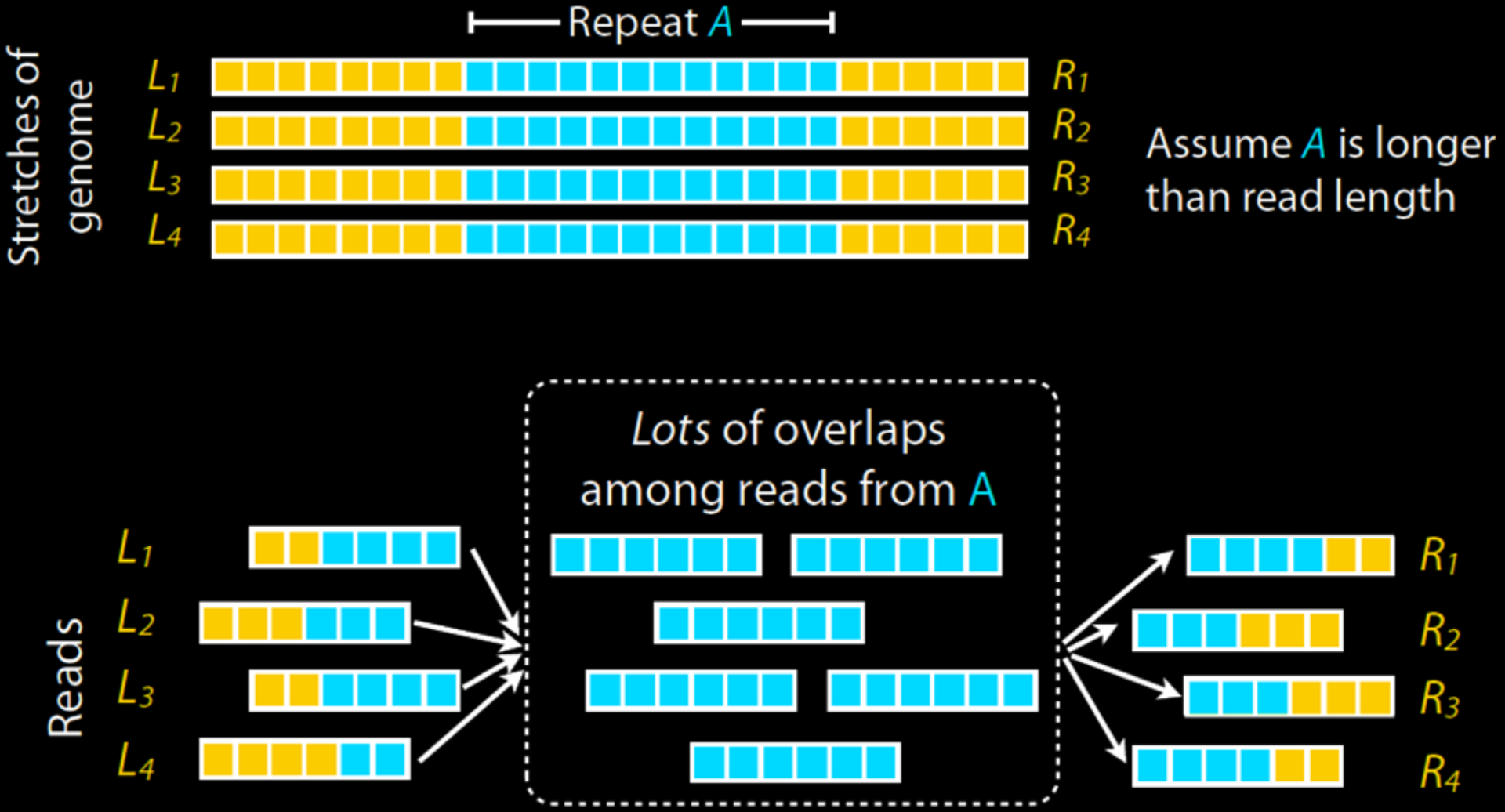
V = {a : ATGCGGTC, *b :* GTCATTGC, *c :* TTGCATTATG*}*
E = {(a,b),(b,c),(c,a)}



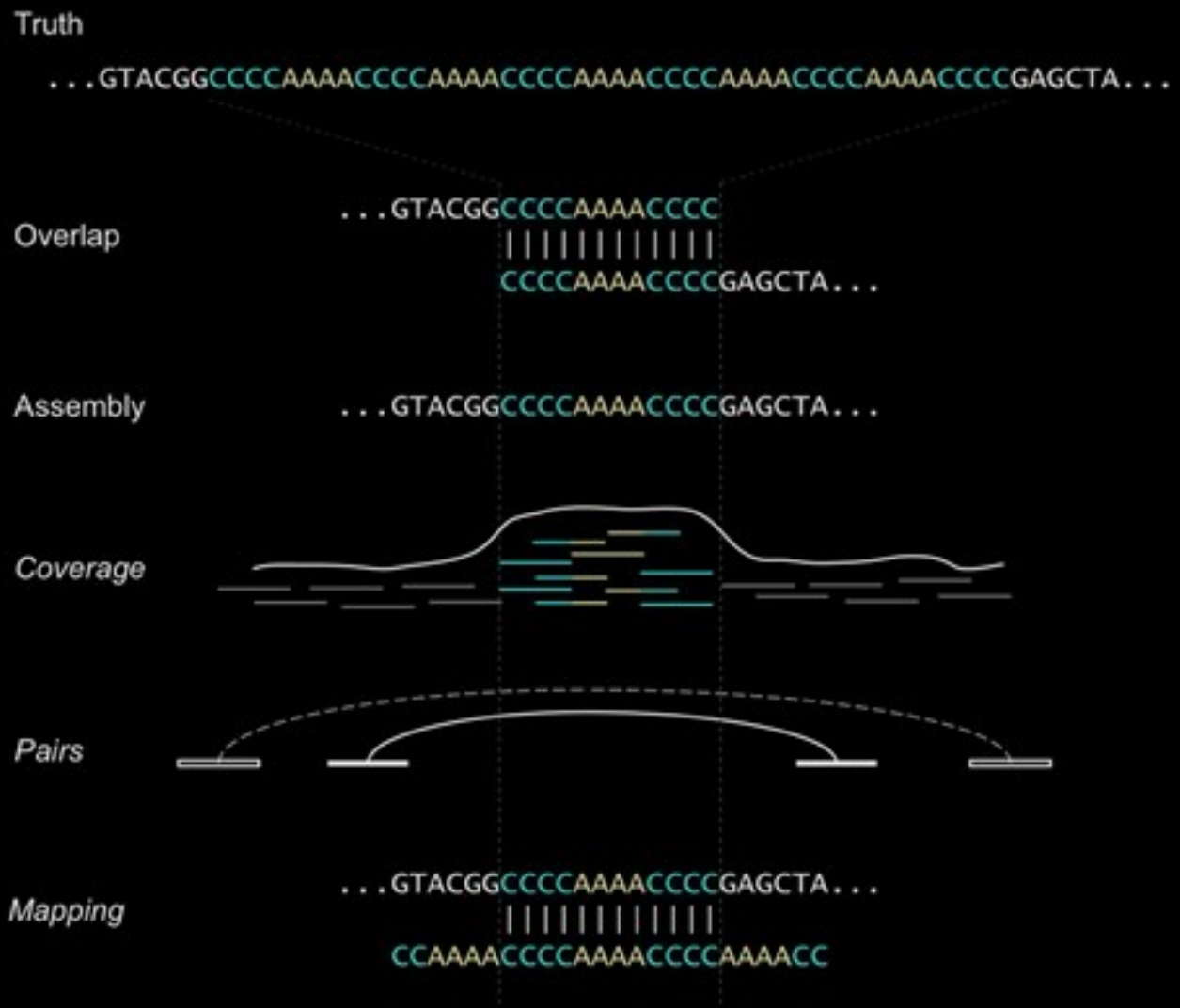
WHY?

1. Circular genomes (e.g. most bacteria, mitochondria)
2. Repeats in the genome
3. Random overlapping

The problem of the repeats

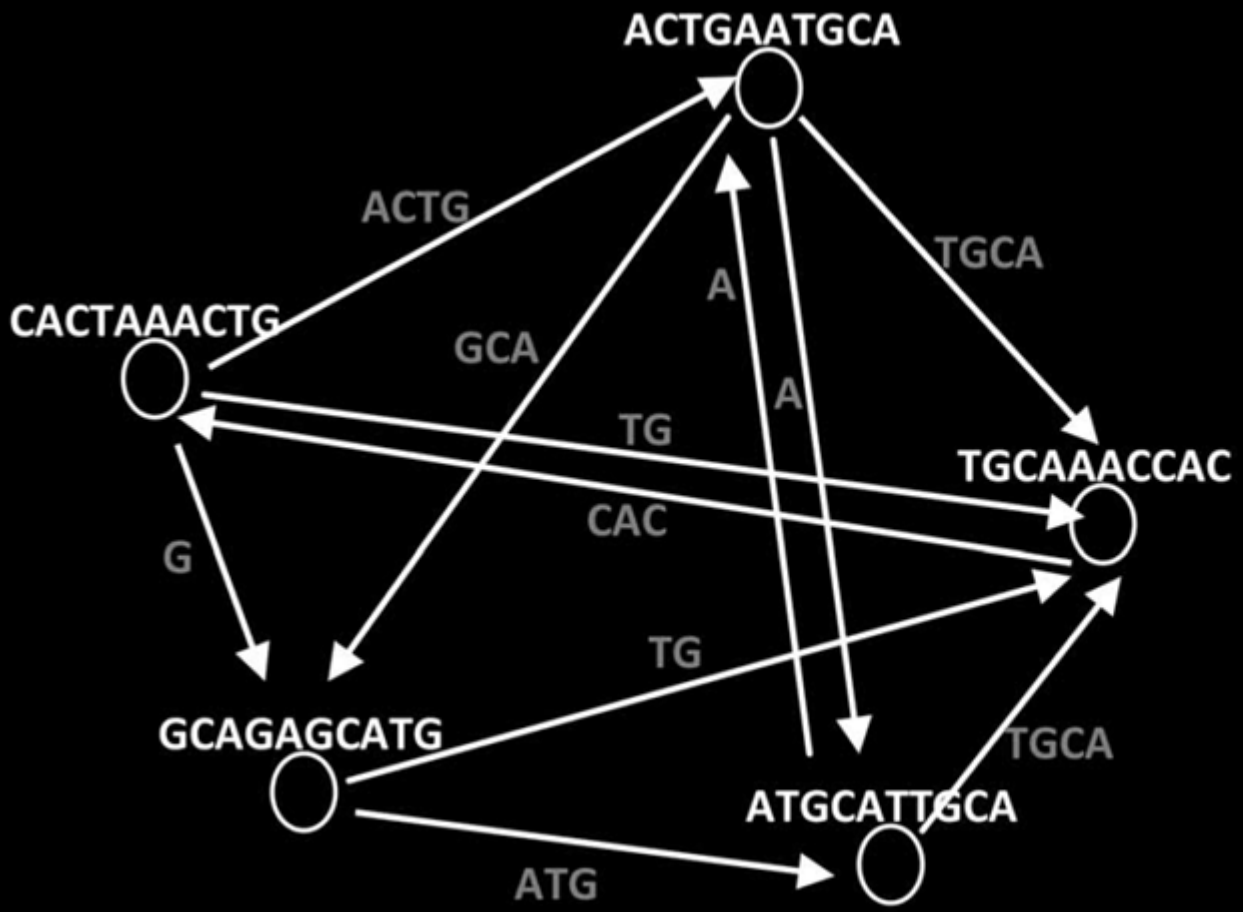


The problem of the repeats



Schatz M C et al. Brief Bioinform 2013;14:213-224

Overlap graphs: an example



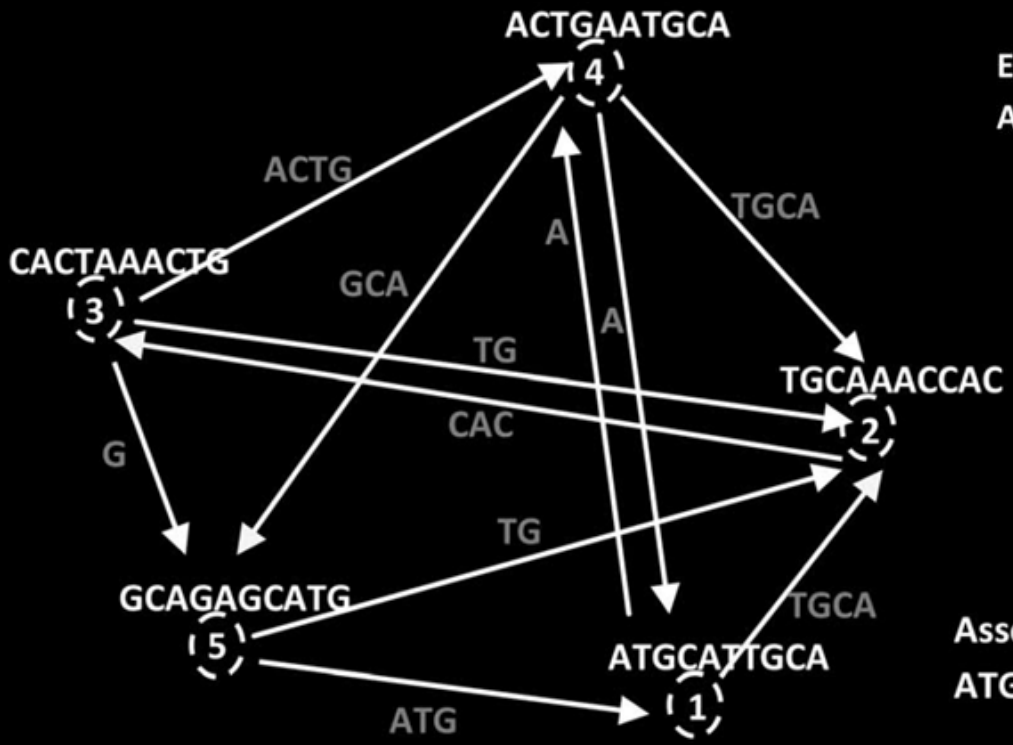
Reads :

ACTGAATGCA
CACTAAACTG
GCAGAGCATG
ATGCATTGCA
TGCAAACCAC

Potential solutions: all paths touching all (connected) nodes once (i.e. Hamiltonian paths)

Schwartz, David C., and Michael S. Waterman. "New generations: Sequencing machines and their computational challenges." Journal of computer science and technology 25.1 (2010): 3-9.

A possible Hamiltonian path



Example of a Hamiltonian Path:

ATGCATTGCA
TGCA**AAC**CAC
CAC**TAA**ACTG
ACTG**AAT**GCA
GCA**GAGCATG**

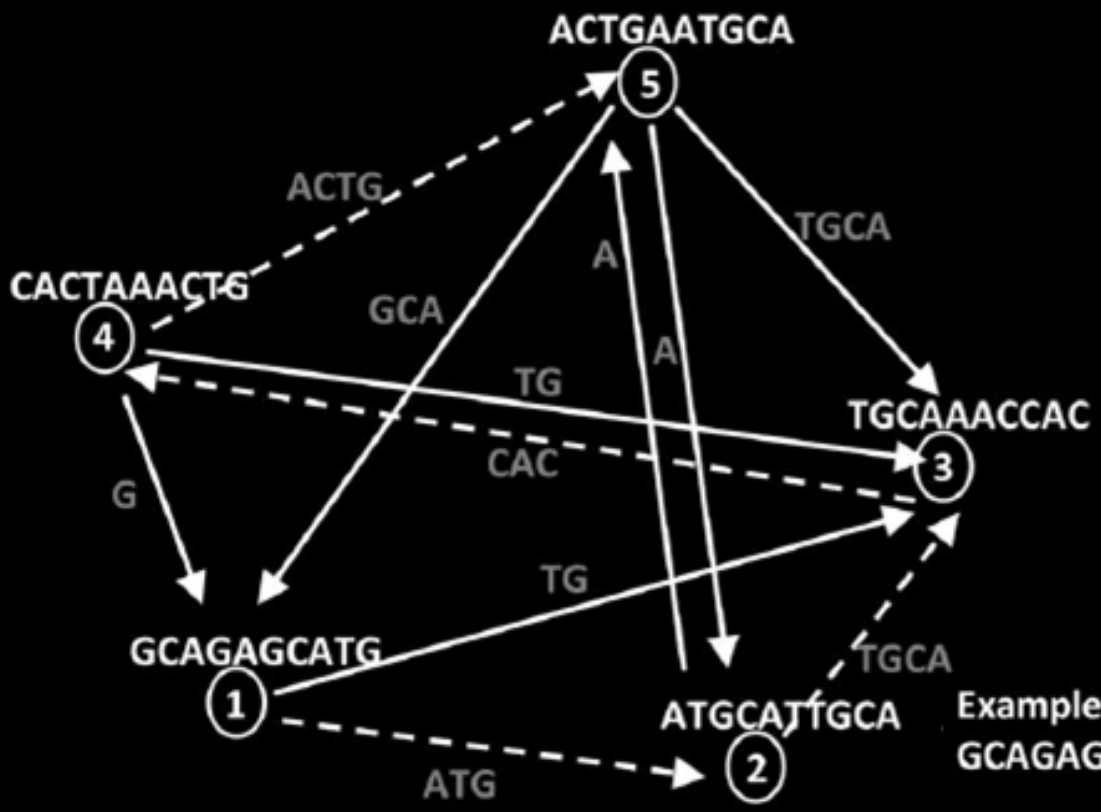
Assembled Reads :

ATGCATTGCA **AAC**CAC**TAA**ACTG **AAT**GCA **GAGCATG**

- Best possible solution: the Hamiltonial path which maximizes the length of the overlaps.
- Which is equivalent to shortest common superstring (SCS) problem.

Unfortunately, the SCS problem is NP-hard (i.e. it cannot be solved efficiently)

A feasible “greedy” approach



1. Randomly select a starting node
2. Select the connected node with maximum overlap as the next visitor

Example of a Greedy Path:

GCAGAGCATG
ATGCATTGCA
TGCAAACCAC
CACTAAACTG
ACTGAATGCA

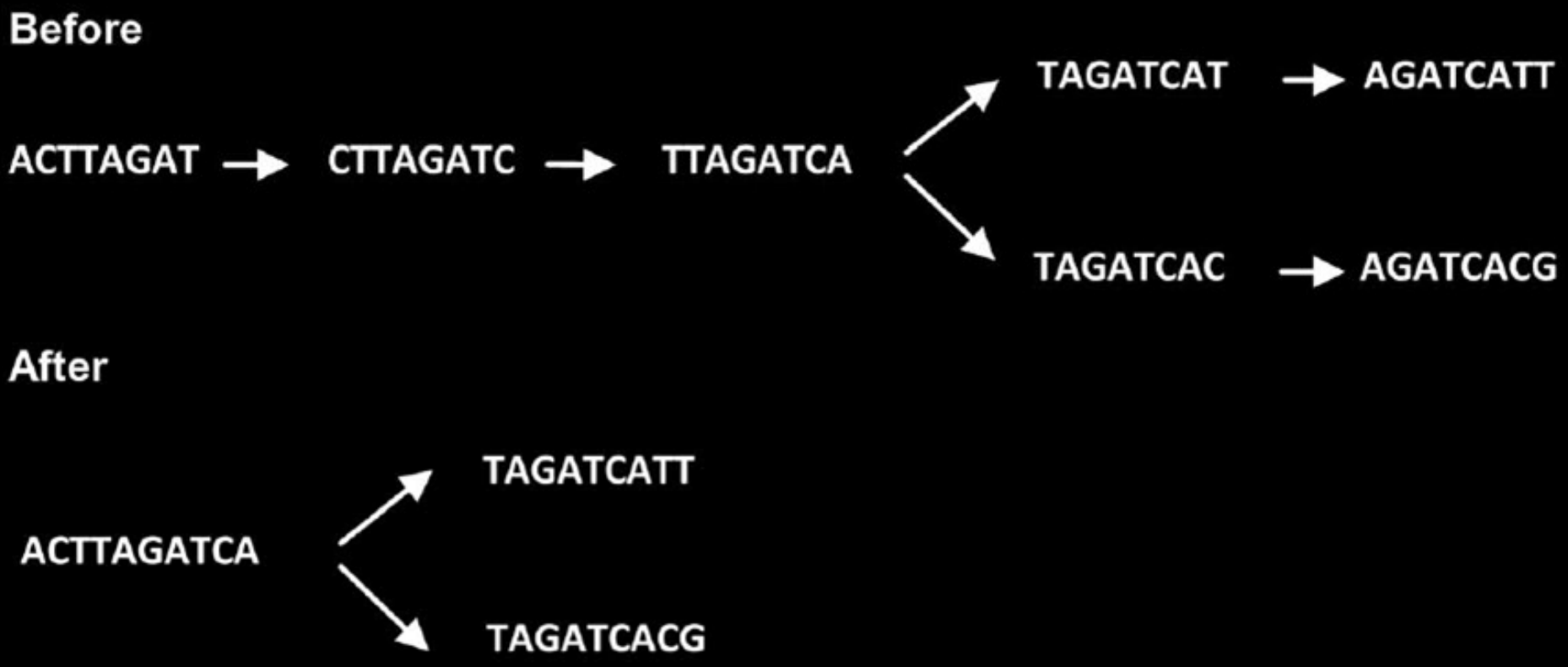
Assembled Reads :

GCAGAGCATGCATTGCAAACCACTAAACTGAATGCA

Node 5 is connected with node 1, so the genome can actually be closed!

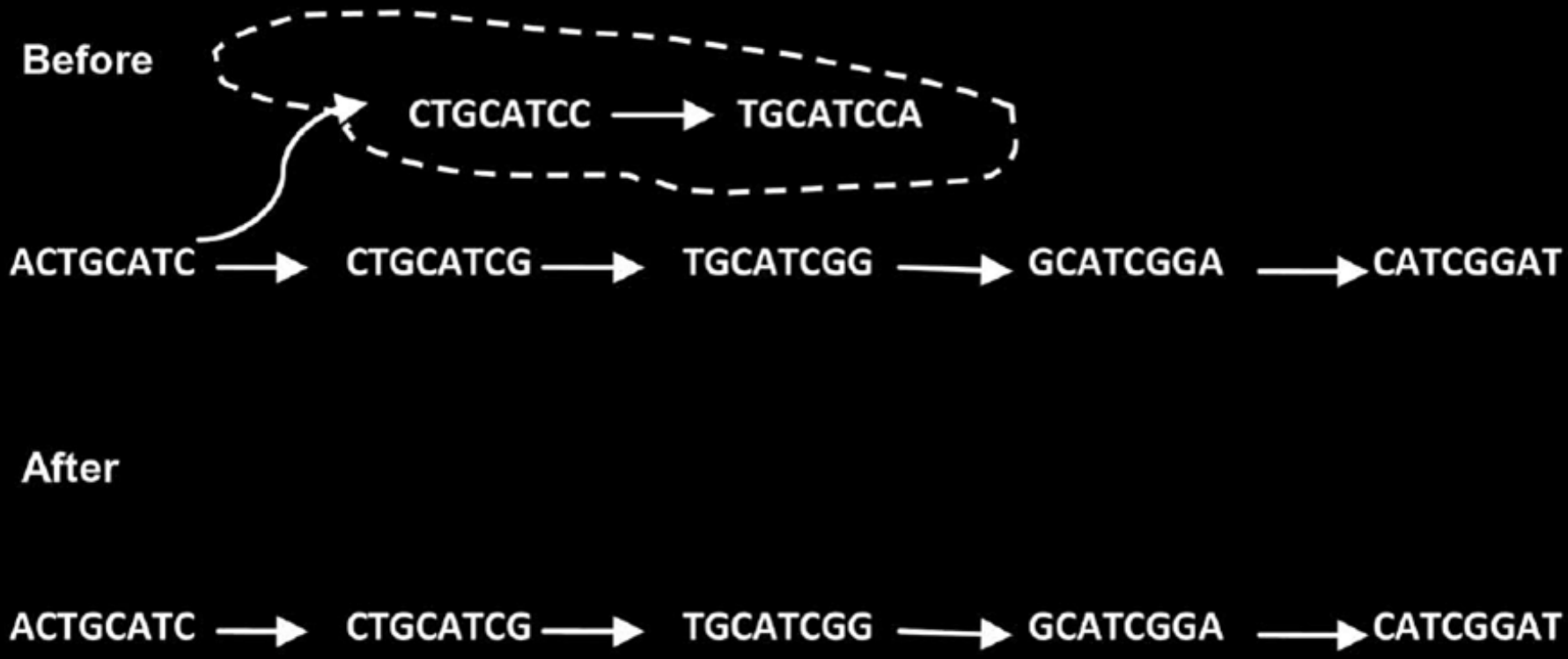
Graph simplification operations

Merging consecutive nodes



Graph simplification operations

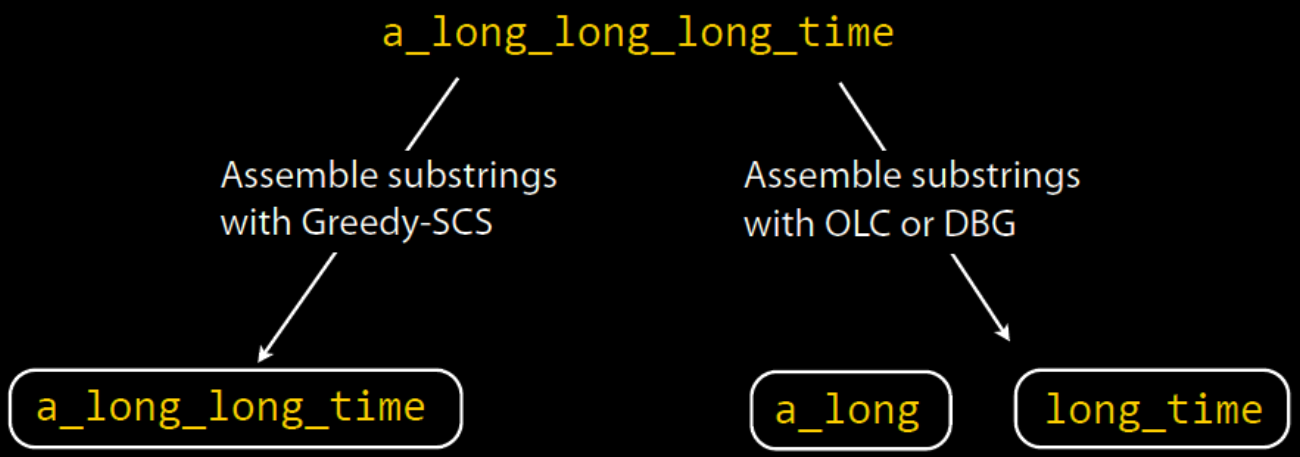
Dead end removal



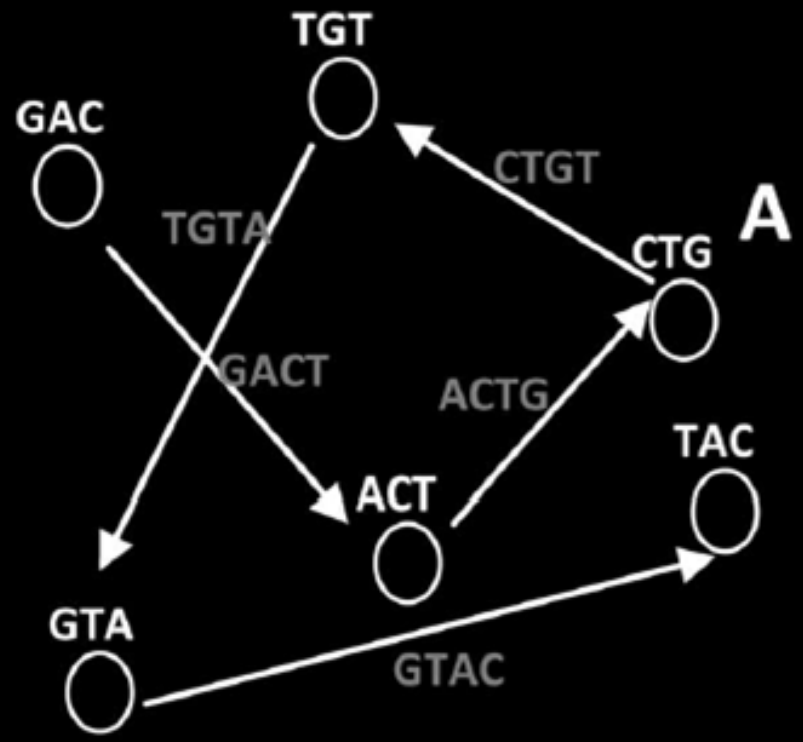
Assembly by overlap graphs has several problems

1. Very problematic when dealing with repeats (real genomes includes a lot of repeats!!):
   e.g. “a_long_long_long_time” will always be assembled as “a_long_long_time” because of the “shortest” constraint
2. It is not tractable (i.e. finding the optimal solution is not computationally feasible)

Real world assemblers are based on de Bruijn Graphs (DBG) and on Overlap Layout Consensus (OLC) approaches



de Bruijn graph assembly: the idea



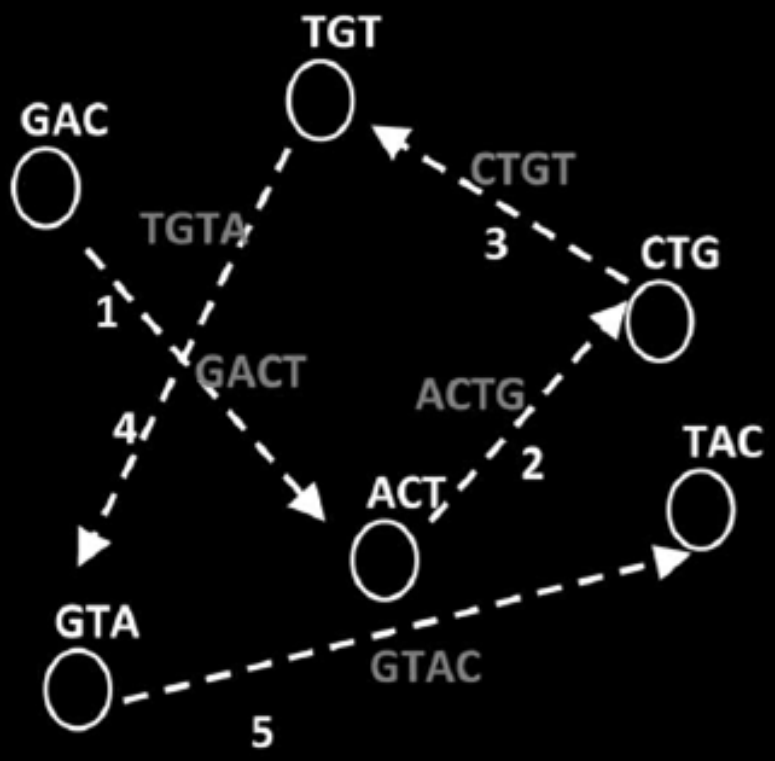
R_1= GACTGTA R_2= ACTGTAC

Set of 3-Kmers of R_1= GAC, ACT,CTG,TGT,GTA

Set of 3-Kmers of R_2= ACT,CTG,TGT,GTA ,TAC

1. Nodes are k mers present in the input reads
2. Edges are overlap k+1 mers

de Bruijn graph assembly: the idea



Example of an Eulerian path :

GACT
ACTG
CTGT
TGTA
GTAC

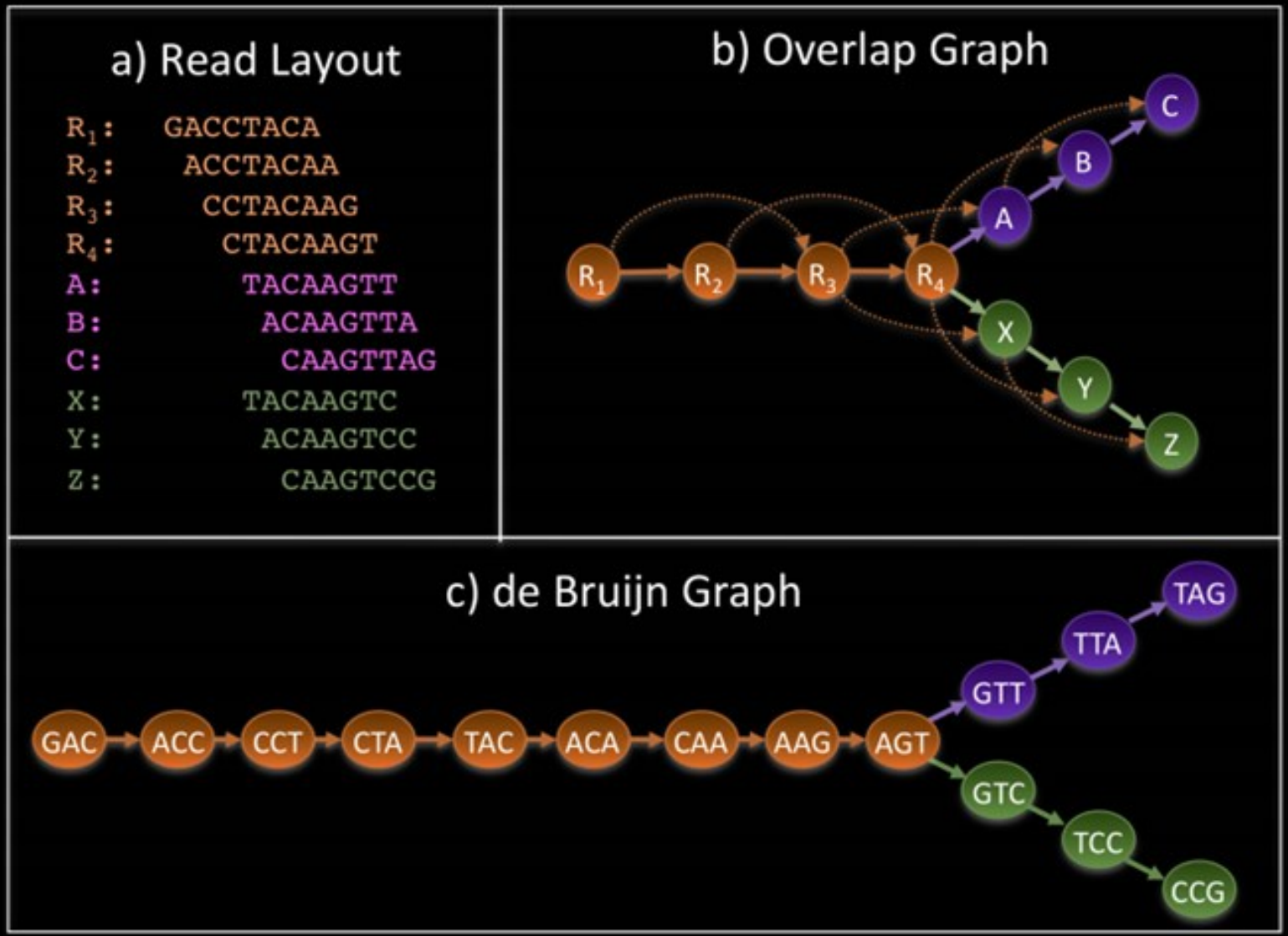
Assembled Reads :

GACTGTAC

Solutions are based on Eulerian paths.

1. Eulerian paths are paths that visit each edge exactly once
2. Eulerian paths can be found much more efficiently than Hamiltonian paths

Overlap vs de Bruijn graph assembly

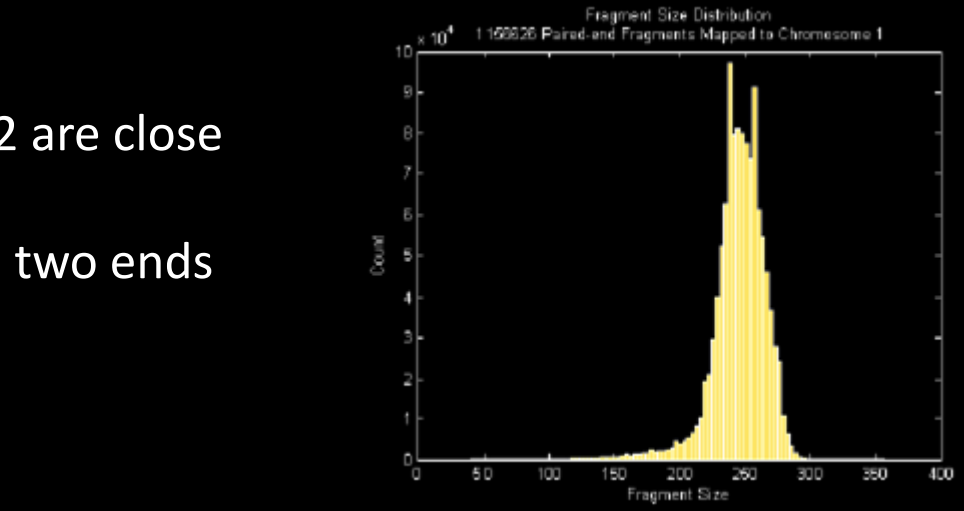


The last step: scaffolding

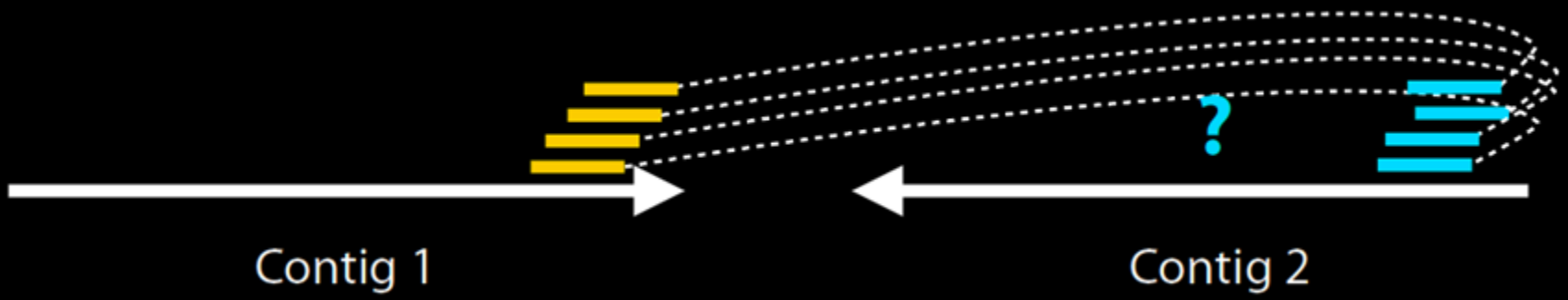
The most successful scaffolding strategy exploit paired-end sequencing

In a situation like that we can:

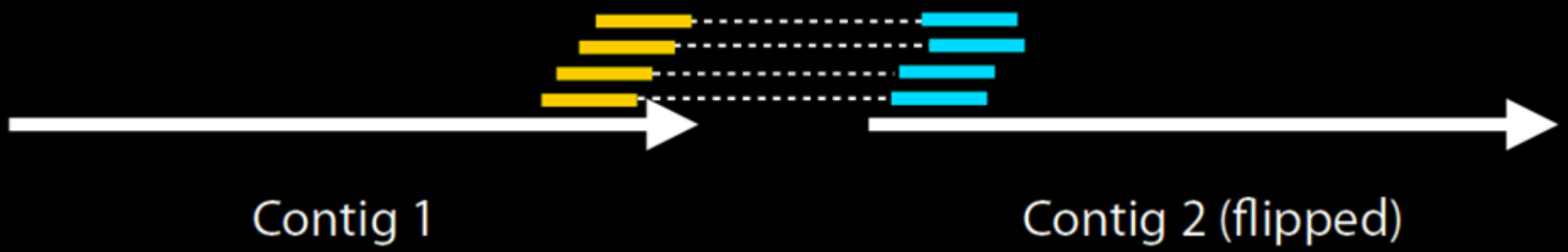
- conclude that Contig 1 and Contig 2 are close on the genome
- estimate the distance between the two ends (and fill them up with Ns)



The last step: scaffolding

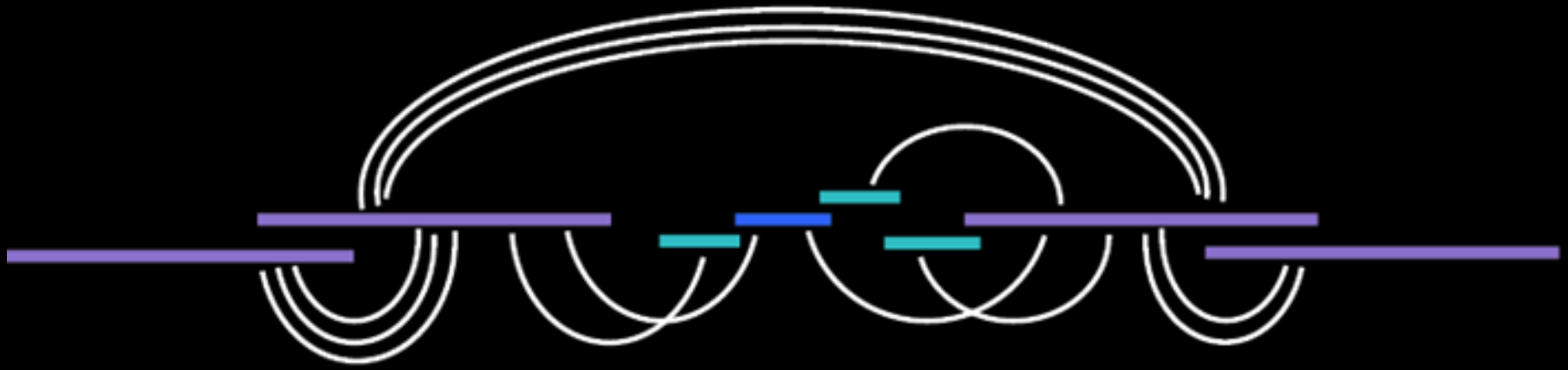


Pairs also tell us about contigs' relative orientation



The last step: scaffolding

More complex scenarios are usually employed



Evaluating assemblies: N50

The N50 size of a set of entities (e.g., contigs or scaffolds) represents the largest entity E such that at least half of the total size of the entities is contained in entities larger than E.

For example, given a collection of contigs with sizes 7, 4, 3, 2, 2, 1, and 1 kb (total size = 20kbp), the N50 length is 4 because we can cover 10 kb with contigs bigger than 4kb.
(http://www.cbcb.umd.edu/research/castats.shtml)

N50 length is the length ‘x’ such that 50% of the sequence is contained in contigs of length x or greater.

(Waterston http://www.pnas.org/cgi/reprint/100/6/3022.pdf)

Questions/comments/discussion