Genome assembly

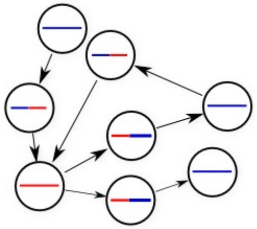
Rayan Chikhi

ENS Cachan Brittany, France

Genome assembly: outline

Bioinformatics context

Problem formulation



Work and perspectives

Genome sequencing

Genome: string s of nucleotides (5 < log10(n) < 10)

$$s \in \{A, C, T, G\}^n$$

Genoscope – Sequencing room

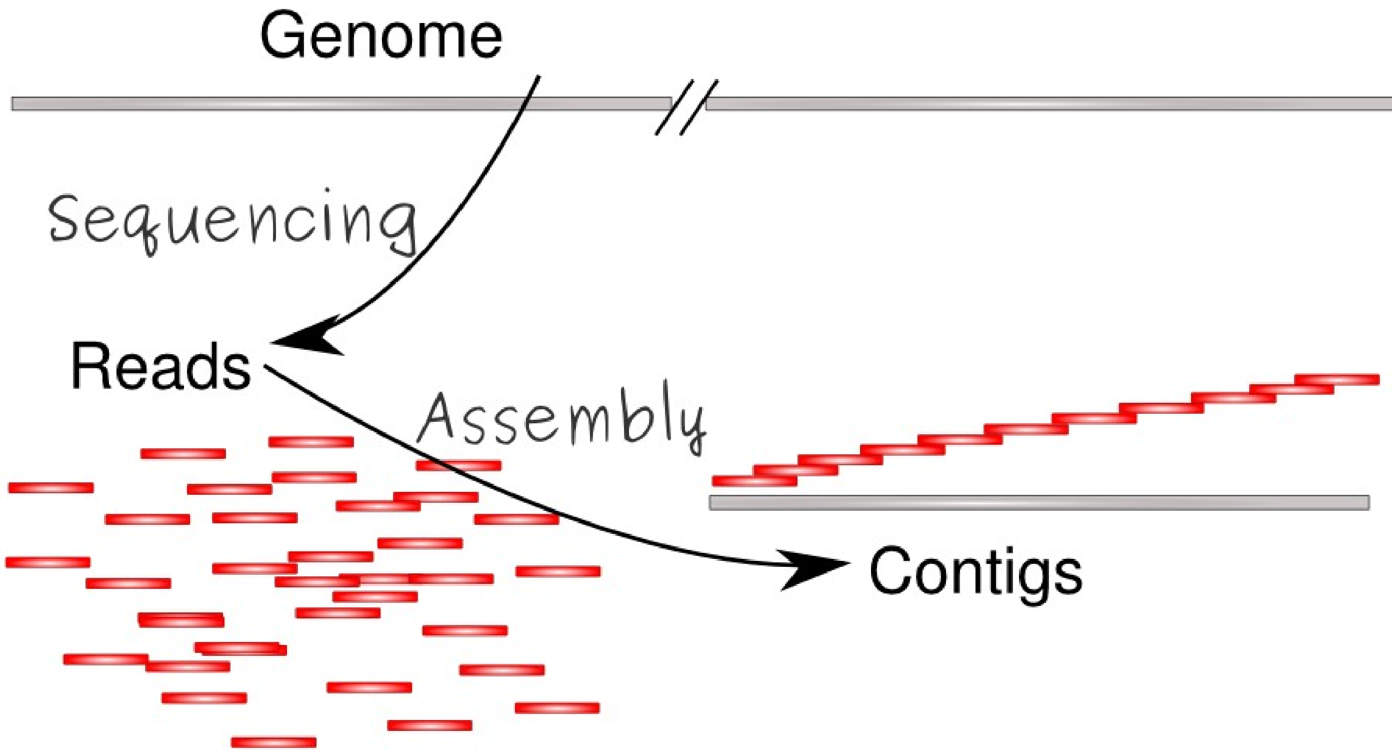
The sequencing process:

Clone the genome many times
Output random fragments

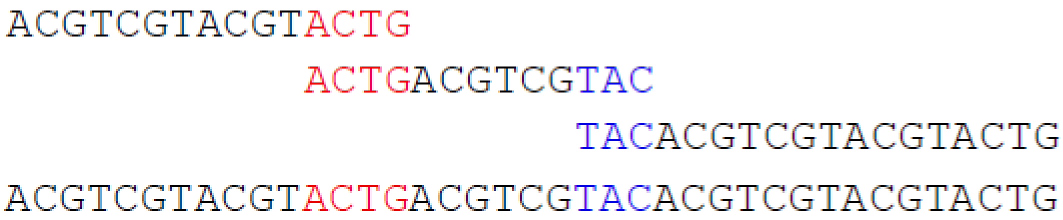
Reads: collection of m substrings of s (6 < log10(m) < 11)

$$\{s_k = s[i_k \ldots i_k + r]\}, i_k \in [1..\mathrm{n}]$$

Assembly



Intuition:



```
ACGTCGTACGTACTG
          ACTGACGTCGTAC
                    TACACGTCGTACGTACTG
ACGTCGTACGTACTGACGTCGTACACGTCGTACGTACTG
```

Actual scenario

Human genome:

~ 3 Gbp
~ 10 billion short reads

Assembly:

2 days
140 Gb memory
~ 1 million contigs

Shortest Common Superstring

Find the shortest string that contains {reads} as substrings.

Max-SNP hard

GREEDY <= 4 OPT (conjectured: 2)

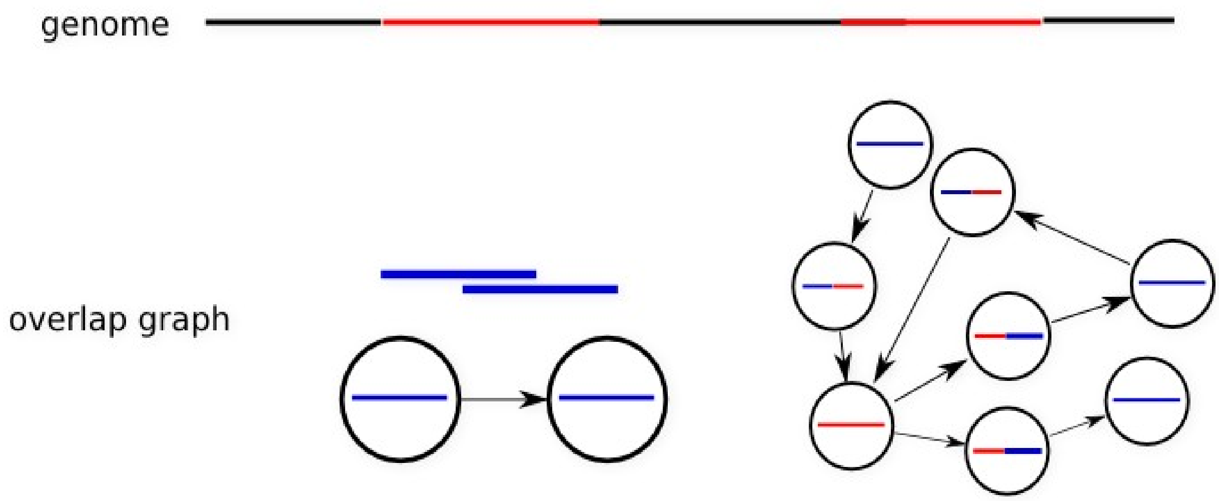
Genome != SCS

Tandem repeats collapsing: ARRRRRB → ARRB

Overcollapsing: ARBRCRD → ARBR'DR'D

where R'=R[1..r]+R[|R|-r..|R|]

Assembly problem [Myers, 2005]



V= { reads }

E= { (r1→r2), s.t. a k-suffix of r1 matches a k-prefix of r2 (*overlap*) }

(+ Remove contained reads and transitively inferable edges.)

Assembly problem: find a generalized Hamiltonian path in G (visit every node at least once) of minimum length

Can we approximate it?

L-reduction to SCS → fixed constant approximation algo

No published approx. algorithm for AP

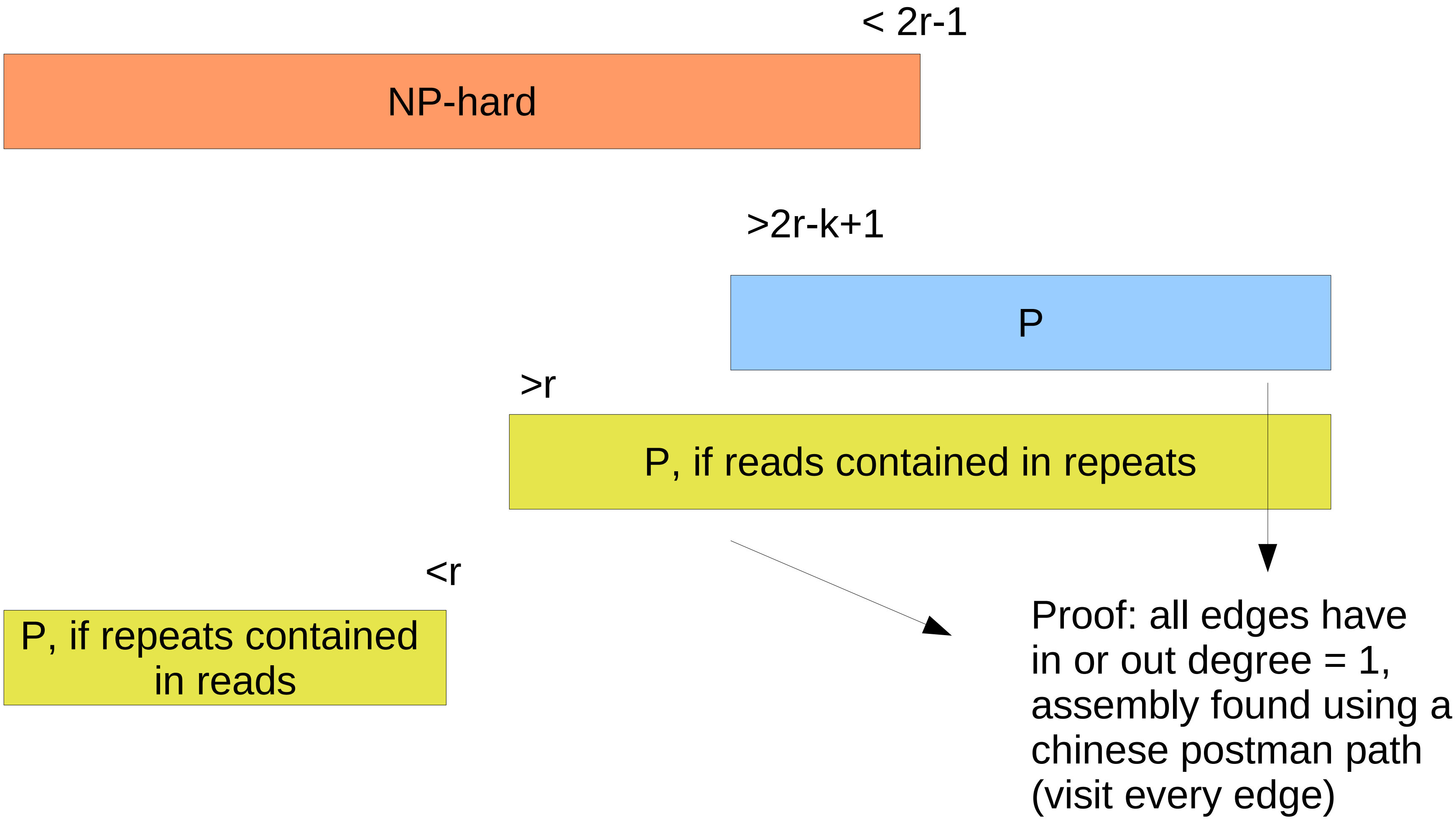
Bad biological news: many solutions with minimal cost

Heuristics: output all linear subgraphs

Parametrized complexity results

Hardness is due to repeats [Nagarajan 09]:

Suppose we have only such repeat sizes:



< 2r-1

NP-hard

>2r-k+1

P

>r

P, if reads contained in repeats

<r

P, if repeats contained in reads

Proof: all edges have in or out degree = 1, assembly found using a chinese postman path (visit every edge)

Given « good » reads, AP can be solved with as an instance of the Chinese Postman Problem:

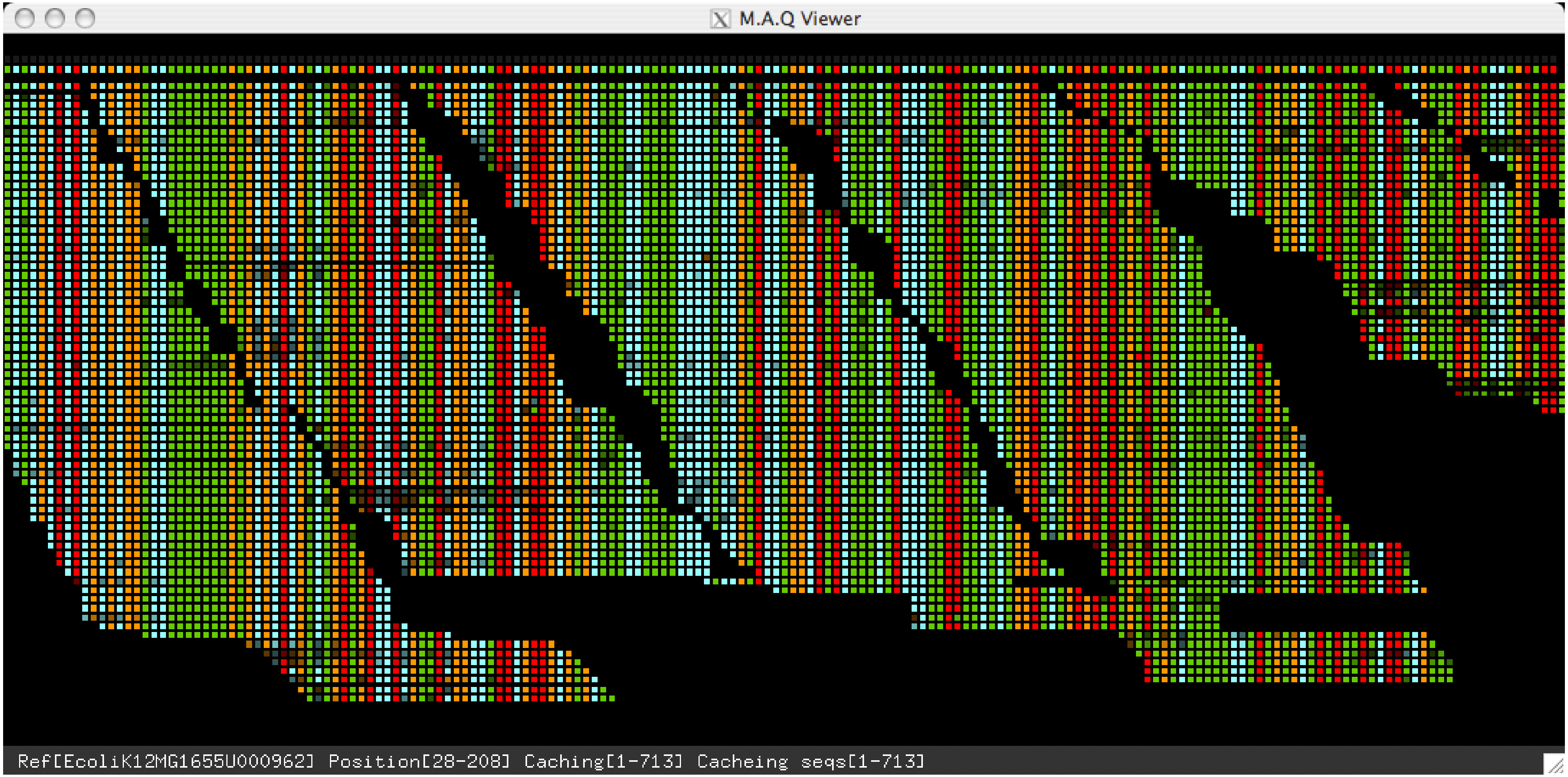
But many reconstructions are possible. (and #CPP is #P-complete)

Maybe find Chinese Postman paths that satisfy a copy-number for each node or given the genome length? NP-hard [Skiena 93].

Nagarajan Conjecture :

If r < 2k, AP is in P.

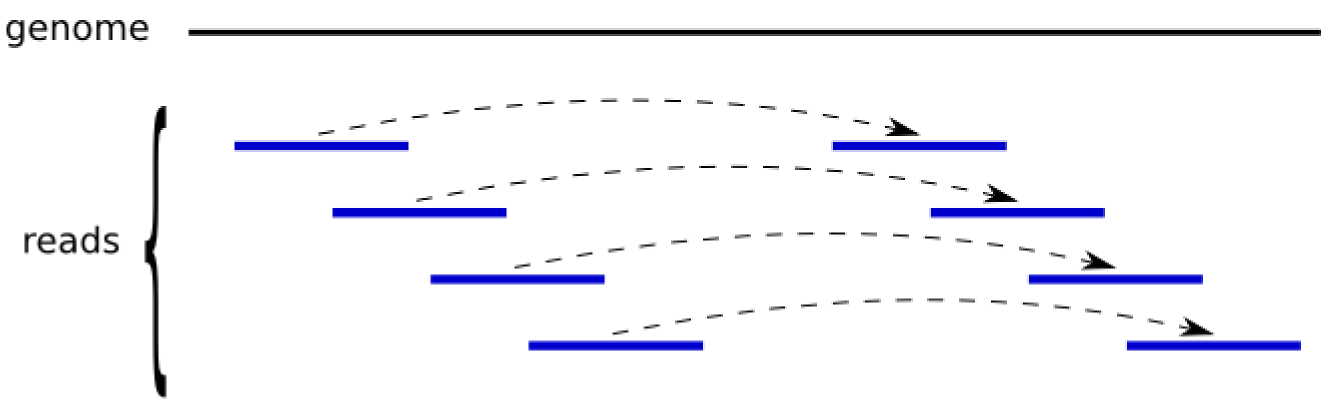
Actual sequencing



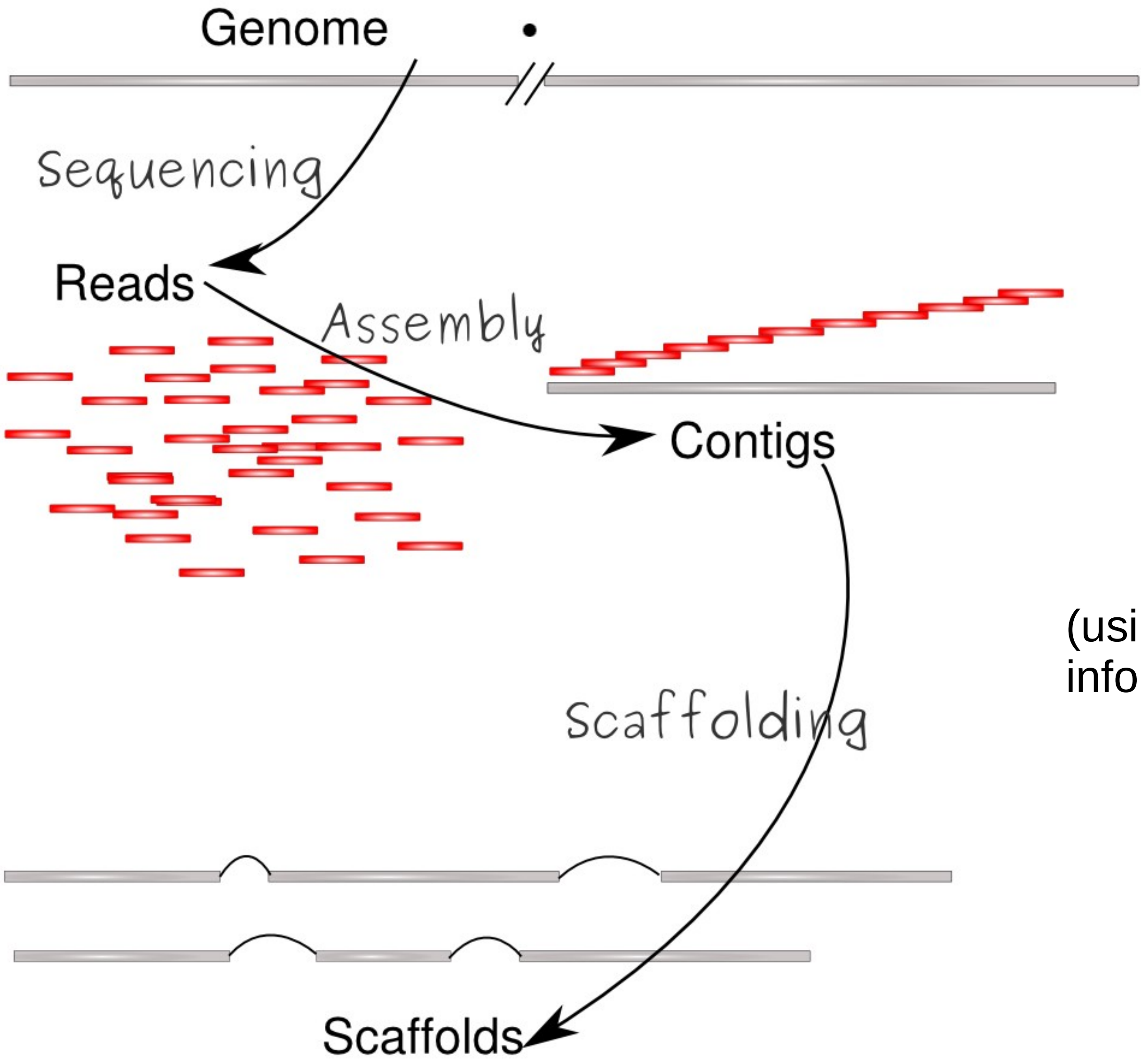
```
Ref[EcoliK12MG1655U000962] Position[28-208] Caching[1-713] Cacheing seqs[1-713]
```

Non-uniform coverage + sequencing errors + DNA is double-stranded

Paired-end assembly



Assembly with paired reads



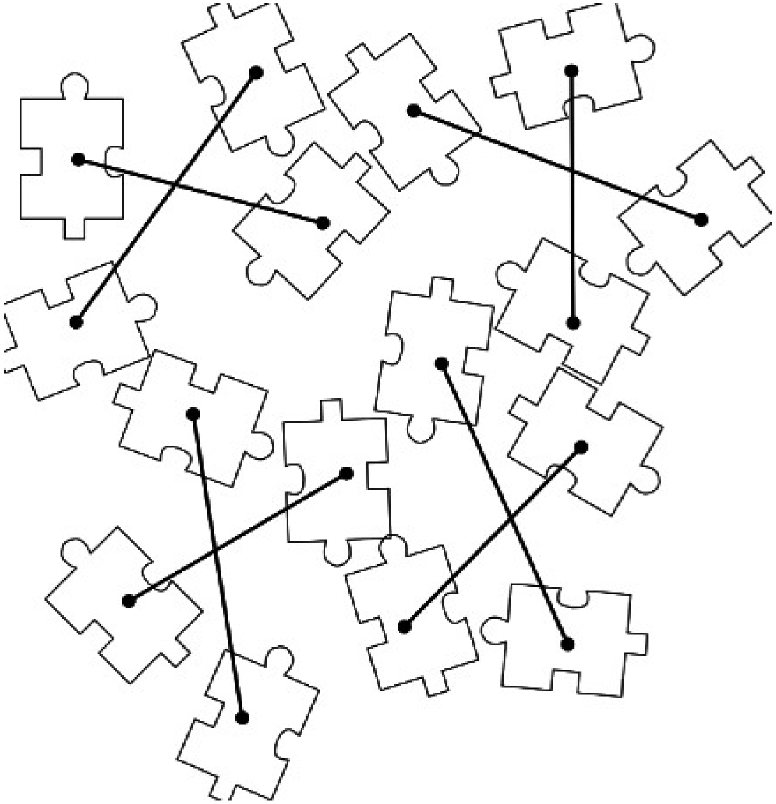
(using extra pairing information)

Scaffolding problem: Find an ordering (absolute coordinate) of contigs.

Not satisfactory: why should we start from contigs?

Paired-end assembly

Intuitively close to the paired jigsaw problem:

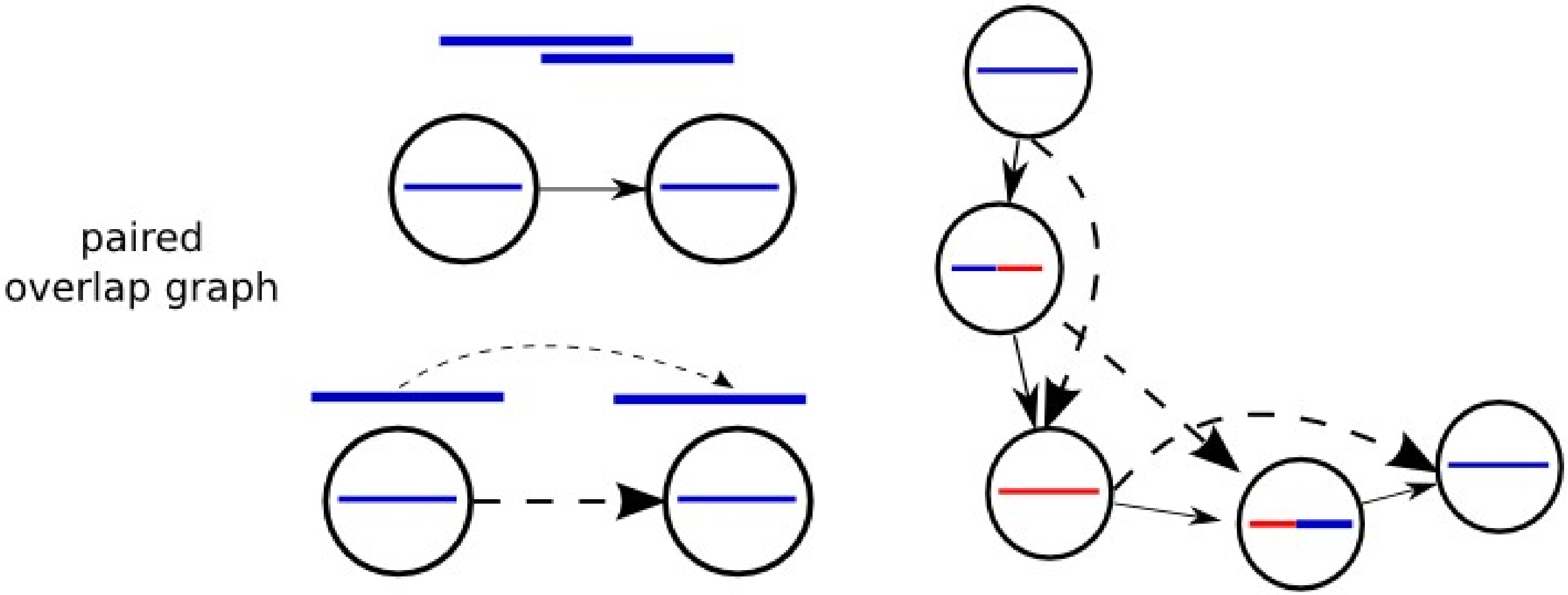


Equivalent paired assembly problem:
add pairs as special edges in the graph
impose the pairing constraint on path.

Paired AP, paired SCS :
MaxSNP-hard

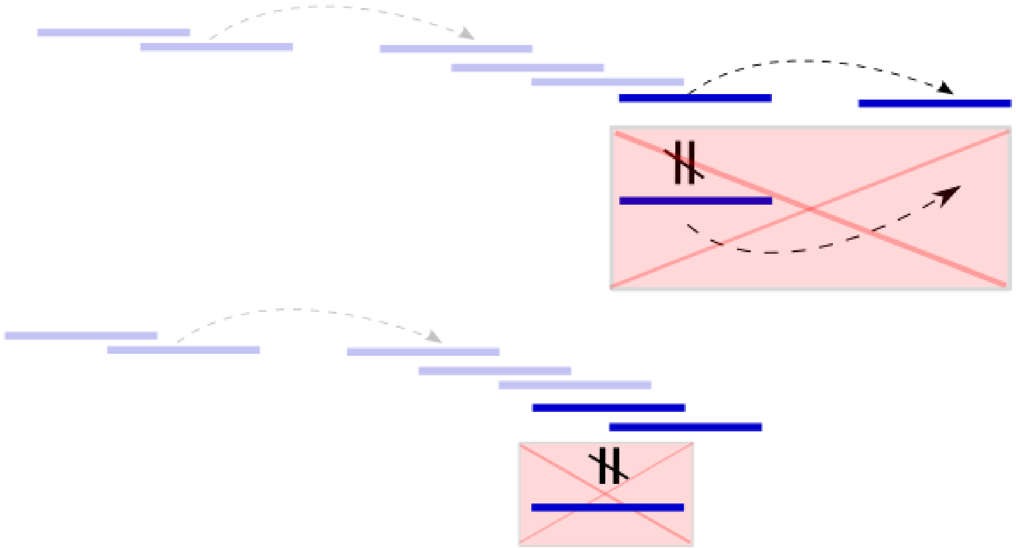
On-going work

paired overlap graphs



Greedy heuristic:

Find all non-overlapping maximal-length paths where (in-degree of visited edges = 1)



Observation:

these paths spell valid scaffolds.

contigs are included

Perspectives

In which cases can we do polynomial-time assembly?

- $r < 2k$?
- can we exploit pairing + repeats $> 2r-k$?

Can we get ε-approximations in some cases?

Thank you for your attention!