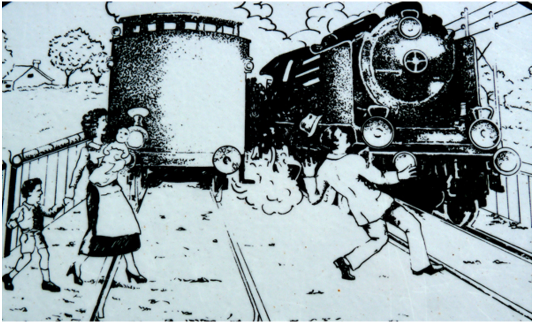
# On the Linearization of Scaffolds sharing Repeated Contigs

Mathias Weller, **Annie Chateau**, Tom Davot, Rodolphe Giroudeau

17 novembre 2017

# Disclaimer

Be careful : An NP-complete problem could hide another one.
This talk is about the **third** one you'll meet.



# Introduction

Sequencing is a technology used to infer genomic information out of DNA material.

- it produces short words, called *reads*, which have to be *assembled* to (try to) reconstruct the whole genomic sequence.
- Assembly can be modelized as an NP-complete problem (Shortest Superstring) [1]

1. Do you follow ? This is the first problem.

# INTRODUCTION

Strategies for assembly :

- Greedy
- Overlap-Layout-Consensus
- De Bruijn graphs

# Introduction

Results : sets of *contigs* of various sizes, disconnected

Far from the whole sequence…

Assembly doesn't take into account some pairing information on the reads.

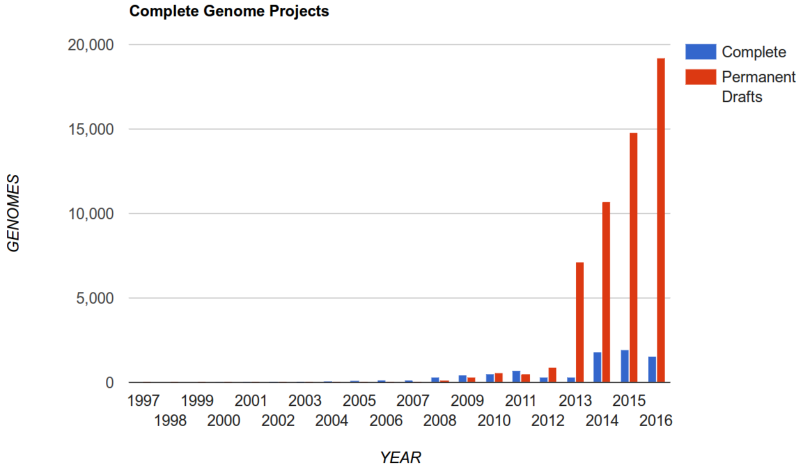
⇒ we need an additional step to use these information and (try to) produce chromosome-long sequences.

# Introduction

Why ?

- Observe genome-scale phenomena
- Improve reference genomes quality
- Lot of genomes have a "draft" status in databases

# Introduction




https://gold.jgi.doe.gov/statistics

# THE SCAFFOLDING PROBLEM [2]

To determine relative order and orientation of contigs, we need :

- Informations between contigs
  - pairing data (common, easy, cheap)
  - phylogenetic data (needs well assembled close species)
  - long reads (full of errors, expensive)
  - ...
- A weight on these information
  - number of pairs of reads
  - probabilistic measure
  - coverage depth
  - ...

2. yes, this is the second one, be patient

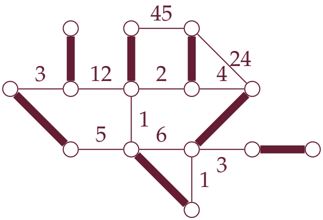
# The Scaffolding Problem

Data are modeled as a graph $G = (V, E)$ :

- **Vertices :** contigs extremities
- **Edges :**
  - between both extremities of a given contig (contig edge)
  - between extremities of distinct contigs (inter-contigs edge)

Weight function : $w : E \to \mathbb{R}$.

# THE SCAFFOLDING PROBLEM



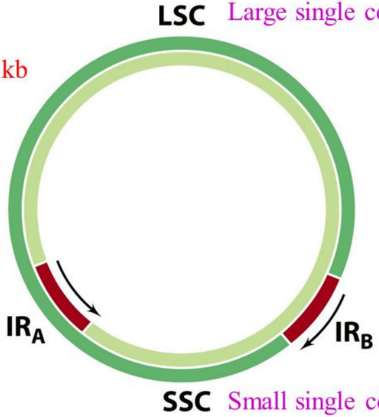

Graph with $2n$ vertices and a perfect matching (contig edges)

We use additionnal parameters to model the desired chromosomic structure : $\sigma_p$ linear chromosomes and $\sigma_c$ circular chromosomes

# WHAT ABOUT "SHARED CONTIGS"?

Typical chloroplast genome – 120 to 160 kb

Large single copy region

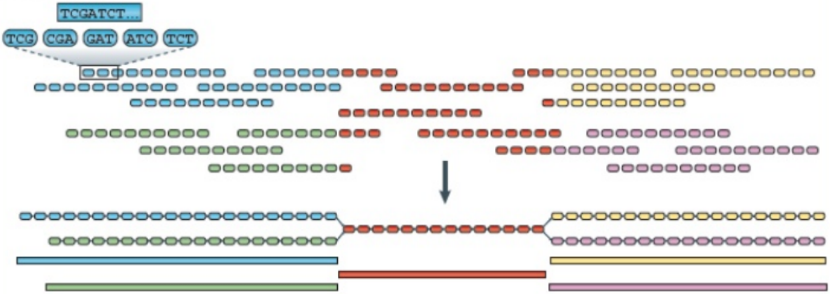



Inverted repeats – 6 to 76 kb

Small single copy region

Figure 15-10 Brock Biology of Microorganisms 11/e

# WHAT ABOUT "SHARED CONTIGS"?



# The Scaffolding problem with multiplicities

**Input :** $G = (V, E)$, $w : E \rightarrow \mathbb{N}$, $M^*$ perfect matching, $\sigma_p, \sigma_c, k \in \mathbb{N}$, $m : E \rightarrow \mathbb{N}$

**Query :** Does it exist a set $S$ of $\sigma_p$ alternating open walks and $\sigma_c$ alternating closed walks covering $G$ such that $w(S) \geq k$ and satisfying the maximal mutliplicity constraint ?

# THE SCAFFOLDING PROBLEM WITH MULTIPLICITIES

Guess what ?[3]

3. If you read footnotes, you already know

# THE SCAFFOLDING PROBLEM WITH MULTIPLICITIES

Guess what ? [3]

It is NP-complete !

3. If you read footnotes, you already know

# That's another story but...

We have algorithms and exact methods to solve this problem efficiently on real instances.

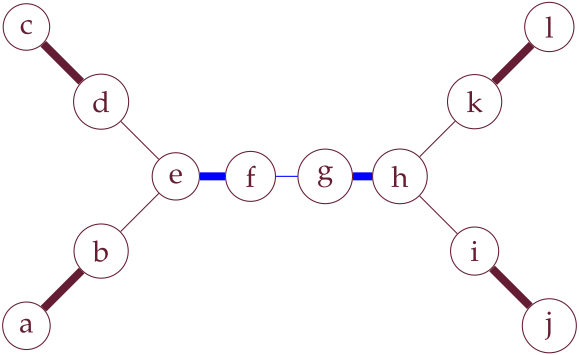
It scales : 1h30 to scaffold a mosquitoe genome...

Take repeats into account...

...everything seems to be perfect, but ...

# Solution graph

Juste a little problem : the solution appears as a graph, not as a collection of cycles and paths. Multiples edges are not ”attributed” to a particular path



# LINEARIZATION OF SOLUTION GRAPH

Biologists like linear (or circular) genomes.

**First solution :** Convince biologists that a solution graph is cool.

It may take a while and lots of efforts...

**Second solution :** Transform the graph into sequences, without creating chimera

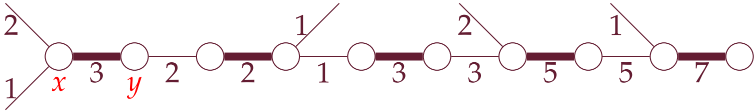
# Notations

**Definition**

Let $p$ be an alternating $u$-$v$-path in a solution graph. If all edges of $p$ have the same multiplicity $\mu$ (that is, $m(e) = \mu$ for all $e \in p$), then $p$ is called *$\mu$-uniform* (or simply *uniform* if $\mu$ is unknown). Further, if $p$ is $\mu$-uniform and each of $u$ and $v$ is incident with a non-matching edge of multiplicity strictly less than $\mu$, then $p$ is called "ambiguous".

# Notations

Ambiguous path :



# The ideal case

**Theorem**

*Let G be a solution graph. Then, G is made up of a unique multiset of alternating walks if and only if G does not contain ambiguous paths.*

# Strategies to get rid of ambiguous paths?

Ignore. Chose an arbitrary multiset of walks making up $G$. Weight is preserved, but risk to produce a chimeric sequence.

## Strategies to get rid of ambiguous paths?

Ignore. Chose an arbitrary multiset of walks making up $G$. Weight is preserved, but risk to produce a chimeric sequence.

Clever. Chose walks that optimize some global criterion (i.e. N50). Again, risk to produce chimeric sequences.

# Strategies to get rid of ambiguous paths?

Ignore. Chose an arbitrary multiset of walks making up $G$. Weight is preserved, but risk to produce a chimeric sequence.

Clever. Chose walks that optimize some global criterion (i.e. N50). Again, risk to produce chimeric sequences.

Brutal. Isolate ambiguous paths by removing all non-matching edges incident to their extremities. Weight could drastically be lowered.

# Strategies to get rid of ambiguous paths?

Ignore. Chose an arbitrary multiset of walks making up $G$. Weight is preserved, but risk to produce a chimeric sequence.

Clever. Chose walks that optimize some global criterion (i.e. N50). Again, risk to produce chimeric sequences.

Brutal. Isolate ambiguous paths by removing all non-matching edges incident to their extremities. Weight could drastically be lowered.

Semi-brutal. Choose a proper set of endpoints of ambiguous path and remove all non-matching edges incident to it. Optimize a criteria.

# Strategies to get rid of ambiguous paths?

Ignore. Chose an arbitrary multiset of walks making up $G$. Weight is preserved, but risk to produce a chimeric sequence.

Clever. Chose walks that optimize some global criterion (i.e. N50). Again, risk to produce chimeric sequences.

Brutal. Isolate ambiguous paths by removing all non-matching edges incident to their extremities. Weight could drastically be lowered.

Semi-brutal. Choose a proper set of endpoints of ambiguous path and remove all non-matching edges incident to it. Optimize a criteria.

# Semi-brutal cut [4]

**Input :** a solution graph $(G, M^*, w, m)$ and some $k \in \mathbb{N}$

**Query :** Is there a set $X$ of extremities of ambiguous paths in $G$ such that removing all non-contig edges incident to vertices of $X$ destroys all ambiguous paths and the score of $X$ is at most $k$ ?

4. Here it is !

# Scoring function for SBC

**Cut score.** Pay one per side of an ambiguous path that is cut : $\text{score}(X) := |X|$.

**Path score.** Pay one for each multiplicity that is cut : $\text{score}(X) := \sum\{m(uv) \mid uv \in E \setminus M^* \wedge uv \cap X \neq \emptyset\}$.

**Weight score.** Pay the total cost of edges that are cut : $\text{score}(X) := \sum\{m(uv) \cdot w(uv) \mid uv \in E \setminus M^* \wedge uv \cap X \neq \emptyset\}$.

# Complexity of SBC

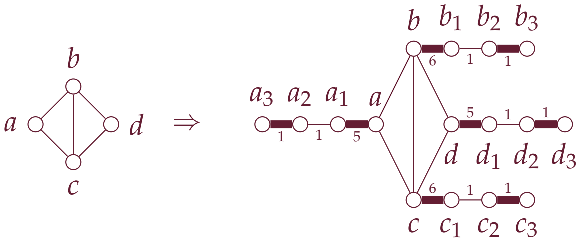
# Complexity

**Theorem**
*It is $\mathcal{NP}$-hard to decide whether all ambiguous paths in a solution graph can be destroyed by removing the non-matching edges incident to at most $k$ endpoints.*

**Theorem**
*It is $\mathcal{NP}$-hard to decide whether a solution graph without ambiguous paths can be obtained by removing at most $k$ non-matching edges.*

# (IDEA OF) PROOFS

For Cut-Score : reduction from Vertex Cover




## Corollary

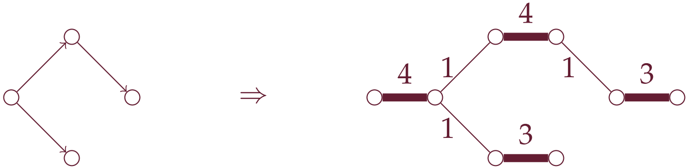
*SBC with cut-score cannot be solved in* $2^{o(n)}$ *time unless ETH fails, and cannot be approximated within a ratio of* 1.3606 *(resp. better than factor* 2*) unless* $\mathcal{P} = \mathcal{NP}$ *(resp. UGC fails).*

# (IDEA OF) PROOFS

For Path-Score : reduction from Transitivity Deletion

**Input :** A triangle-free directed acyclic graph $(V, A)$ and $k \geq 0$.

**Question :** Is there an $A' \subseteq A$ with $|A'| \leq k$ and $(V, A \setminus A')$ is transitive ?

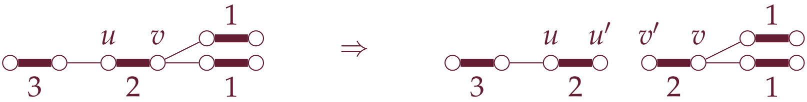



For Weight-Score : Path-Score is a special case of Weight-Score

# Polynomial cases

## Rule

*Let $uv \in M^*$ be a contig edge that does not occur in ambiguous paths and let $u$ and $v$ have degree at least two. Then, remove $uv$, add new vertices $u'$ and $v'$ and add the contig edges $uv'$ and $vu'$ with multiplicity $m(uv)$.*



# Polynomial cases

**Trees :** dynamic programming

$c(x)$ = cost of a solution below $x$ in which all non-contigs incident with $x$ are cut

$\bar{c}(x)$ = cost of any other solution below $x$.

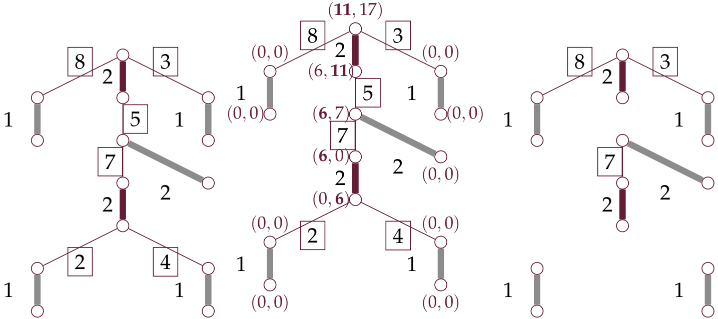
If $x$ is a leaf of $G$, $c(x) = \bar{c}(x) = 0$.

For any non-leaf $x$, we set

$$c(x) = \sum_{y \in Children(x)} min(\bar{c}(y), c(y)) + \sum_{y \in Children(x) \setminus \{M^*(x)\}} w_{xy}$$

$$\bar{c}(x) = \begin{cases} c(M^*(x)) & \text{if } M^*\text{(x) is below} x \\ 0 & \text{otherwise} \end{cases} + \sum_{y \in Children(x) \setminus \{M^*(x)\}} min(\bar{c}(y), c(y) + w_{xy})$$

# Polynomial cases



# Polynomial cases

**Max degree two :** collection of cycles and paths.

ILP formulation yields totally unimodular matrices

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}$$

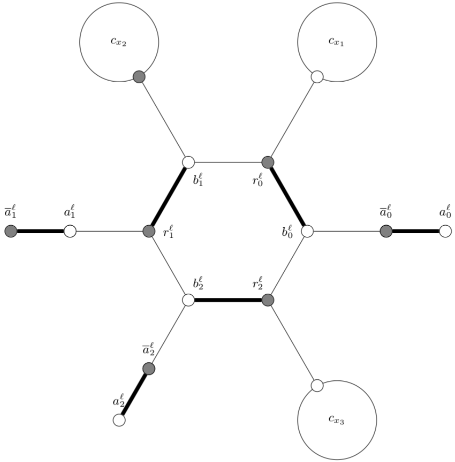
# Towards the frontier

**Theorem**

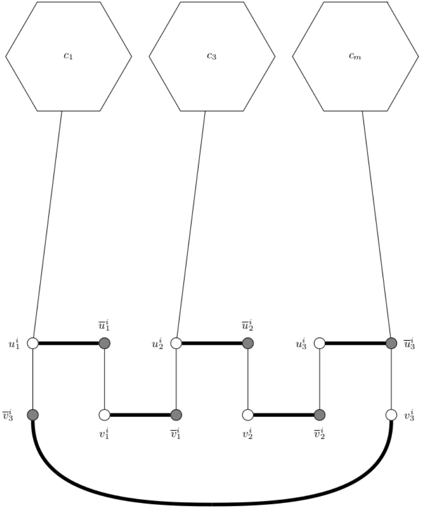
*The problem* SBC *for Cut-Score is* $\mathcal{NP}$*-complete, even if the graph is bipartite, planar, has maximum degree three and has its multiplicities in the set* $\{1, 2\}$.

# Towards the frontier

**Idea of proof :** reduction from 3-SAT



# TOWARDS THE FRONTIER



# Conclusion

There is a lot of work left :

- Approximation
- Define and test heuristics
- Test ILP
- Lower bounds of complexity