Novel components at the periphery of long read genome assembly tools

A bioinformatics thesis

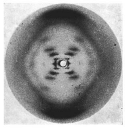
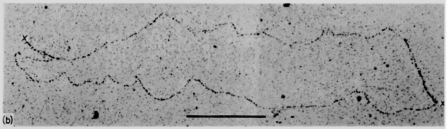
Pierre Marijon
Directeurs: Jean-Stéphane Varré, Rayan Chikhi

2 december 2019

Équipe BONSAI, Inria, University of Lille

Introduction

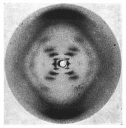
Go back to bases

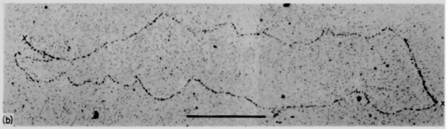



X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

[1] [Franklin and Gosling, 1953]
[2] [Cairns, 1963]

Go back to bases



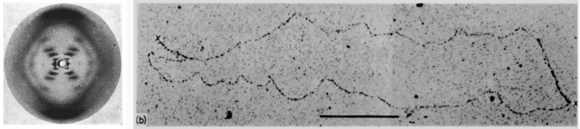


X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

DNA is the carrier of genetic information, having access to this information allows us to:

[1] [Franklin and Gosling, 1953]
[2] [Cairns, 1963]

Go back to bases



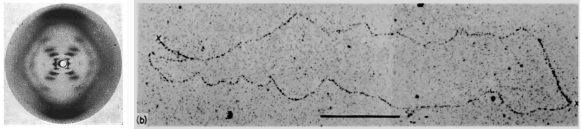
X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

DNA is the carrier of genetic information, having access to this information allows us to:

- understand the origin of genetic diseases

[1] [Franklin and Gosling, 1953]
[2] [Cairns, 1963]

Go back to bases



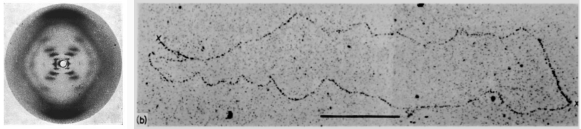
X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

DNA is the carrier of genetic information, having access to this information allows us to:

- understand the origin of genetic diseases
- reconstruct steps of the evolution

[1] [Franklin and Gosling, 1953]
[2] [Cairns, 1963]

Go back to bases



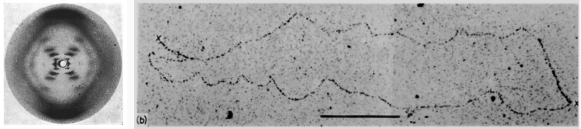
X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

DNA is the carrier of genetic information, having access to this information allows us to:

- understand the origin of genetic diseases
- reconstruct steps of the evolution
- identify species

[1][Franklin and Gosling, 1953]
[2][Cairns, 1963]

Go back to bases



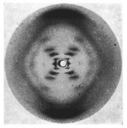
X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

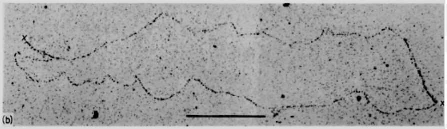
DNA is the carrier of genetic information, having access to this information allows us to:

- understand the origin of genetic diseases
- reconstruct steps of the evolution
- identify species
- observe the structure of the population

[1][Franklin and Gosling, 1953]
[2][Cairns, 1963]

Go back to bases





X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

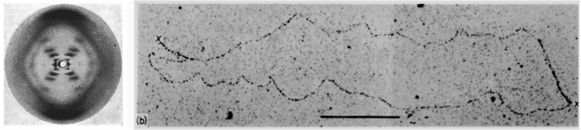
DNA is the carrier of genetic information, having access to this information allows us to:

- understand the origin of genetic diseases
- reconstruct steps of the evolution
- identify species
- observe the structure of the population

Many biological phenomena can be seen from a genomic perspective

[1] [Franklin and Gosling, 1953]
[2] [Cairns, 1963]

Go back to bases



X-ray diffraction of DNA[1] & Autoradiography of *E. coli* chromosome[2]

DNA is the carrier of genetic information, having access to this information allows us to:

- understand the origin of genetic diseases
- reconstruct steps of the evolution
- identify species
- observe the structure of the population

Many biological phenomena can be seen from a genomic perspective

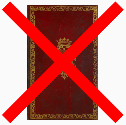
How we can read this information ?

[1][Franklin and Gosling, 1953]
[2][Cairns, 1963]

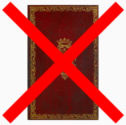
Reading and assembling DNA: a crazy monk analogy

Reading and assembling DNA: a crazy monk analogy

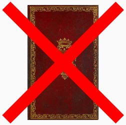
Reading and assembling DNA: a crazy monk analogy



Reading and assembling DNA: a crazy monk analogy

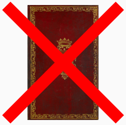


Reading and assembling DNA: a crazy monk analogy



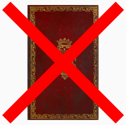
nostra, pAr inceptos himenaeos
nostra, per inceptos
conubia nostra, per inceptos
diam pharetra vitae. Class
placerat leo leo, in feugiat diam
vitae. Clas aptent taciti sociosqu ad
per inceptos
per inceptos
leo leEEEo
per inos
per inceptos
Suspendisse placerat leo leo
sociosqu ad litora torquent per conubia

Reading and assembling DNA: a crazy monk analogy



nostra, pAr inceptos himenaeos
nostra, per inceptos
conubia nostra, per inceptos
diam pharetra vitae. Class
placerat leo leo, in feugiat diam
vitae. Clas aptent taciti sociosqu ad
per inceptos
per inos
per inceptos
per inceptos
leo leEEEo
Suspendisse placerat leo leo
sociosqu ad litora torquent per conubia

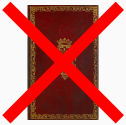
Reading and assembling DNA: a crazy monk analogy



nostra, pAr inceptos himenaeos
nostra, per inceptos
conubia nostra, per inceptos
diam pharetra vitae. Class
placerat leo leo, in feugiat diam
vitae. Clas aptent taciti sociosqu ad
per inceptos
per inceptos
leo leEEEo
per inos
per inceptos
Suspendisse placerat leo leo
sociosqu ad litora torquent per conubia

Suspendisse placerat leo leo, in feugiat diam pharetra vitae. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos

Reading and assembling DNA: a crazy monk analogy



nostra, pAr inceptos himenaeos
nostra, per inceptos
conubia nostra, per inceptos
diam pharetra vitae. Class
placerat leo leo, in feugiat diam
vitae. Clas aptent taciti sociosqu ad
per inceptos
per inceptos
leo leEEEo
per inos
per inceptos
Suspendisse placerat leo leo
sociosqu ad litora torquent per conubia

Class aptent taciti sociosqu ad litora torquent pAr conubia nostra,

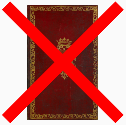
per inceptos per inceptos himenaeos.

Suspendisse placerat leo leo,

leo leEEo in feugiat diam pharetra vitae.

Reading and assembling DNA: a crazy monk analogy

Biologist



Genome

Sequencer

nostra, pAr inceptos himenaeos
nostra, per inceptos
conubia nostra, per inceptos
diam pharetra vitae. Class
placerat leo leo, in feugiat diam
vitae. Clas aptent taciti sociosqu ad
per inceptos
per inos
per inceptos
per inceptos
leo leEEEo
Suspendisse placerat leo leo
sociosqu ad litora torquent per conubia

Class aptent taciti sociosqu ad litora torquent pAr conubia nostra,

per inceptos per inceptos himenaeos.

Suspendisse placerat leo leo,

leo leEEo in feugiat diam pharetra vitae.

Assembly tools

My contribution

PhD main concern : improving result of assembly tools without modifying existing assembly tools

We focus on:

[3] [Marijon et al., 2019b]
[4] [Marijon et al., 2019a]

My contribution

PhD main concern : improving result of assembly tools without modifying existing assembly tools

We focus on:

- improving input of assembly [3]

[3][Marijon et al., 2019b]
[4][Marijon et al., 2019a]

My contribution

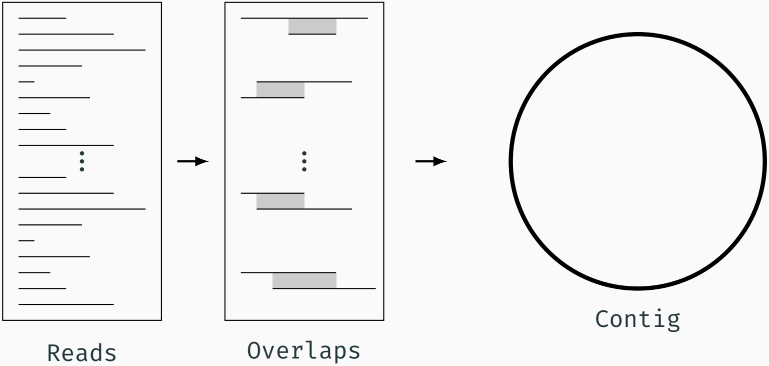
PhD main concern : improving result of assembly tools without modifying existing assembly tools

We focus on:

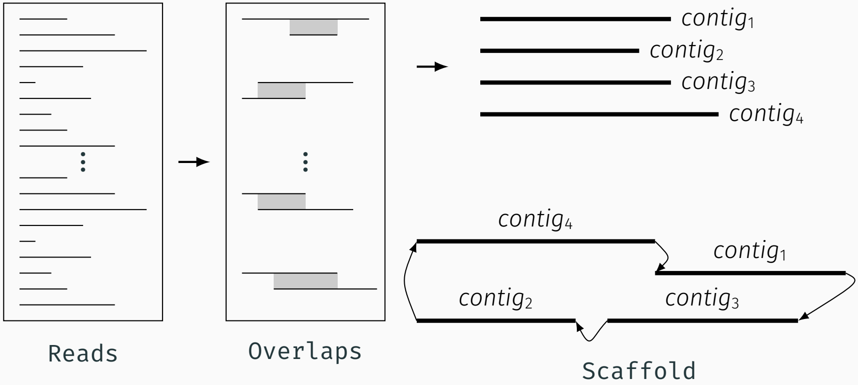
- improving input of assembly [3]
- trying to understand why assembly is fragemented and if we can solve this fragmentation [4]

[3] [Marijon et al., 2019b]
[4] [Marijon et al., 2019a]

Glossary

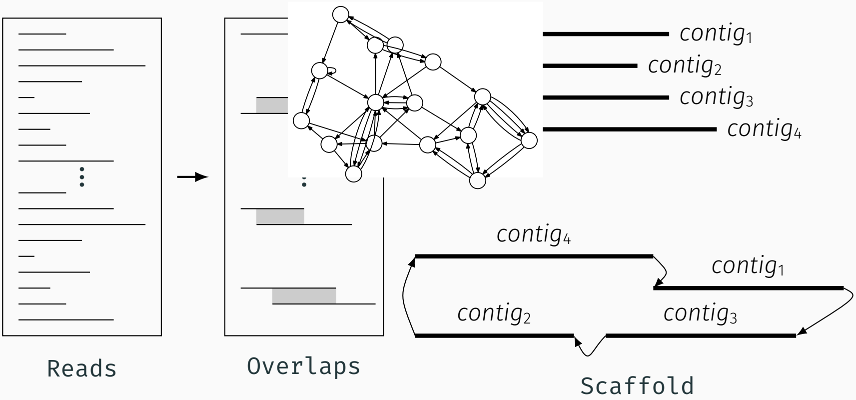


Glossary



Reads
Overlaps
$contig_1$
$contig_2$
$contig_3$
$contig_4$
$contig_4$
$contig_1$
$contig_2$
$contig_3$
Scaffold

Glossary



Reads
Overlaps
contig1
contig2
contig3
contig4
contig4
contig1
contig2
contig3
Scaffold

Assembly problem isn't solved

Number of contigs	2nd Gen.	3rd Gen.	# chromosome
Gorilla gorilla gorilla			24 x 2
Schistosoma japonicum			8 x 2
Escherichia coli			1
Ambystoma mexicanum			14 x 2

[5] [Scally et al., 2012]
[6] [Gordon et al., 2016]
[7] [Schistosoma japonicum Genome Sequencing and Functional Analysis Consortium, 2009]
[8] [Luo et al., 2019]
[9] GenBank Id 6313798
[10] [Maio et al., 2019]
[11] [Keinath et al., 2015]
[12] [Smith et al., 2019]

Assembly problem isn't solved

Number of contigs	2nd Gen.	3rd Gen.	# chromosome
Gorilla gorilla gorilla	461,501 [5]		24 x 2
Schistosoma japonicum	95,269 [7]		8 x 2
Escherichia coli	1 [9]		1
Ambystoma mexicanum	1,479,440 [11]		14 x 2

[5] [Scally et al., 2012]
[6] [Gordon et al., 2016]
[7] [Schistosoma japonicum Genome Sequencing and Functional Analysis Consortium, 2009]
[8] [Luo et al., 2019]
[9] GenBank Id 6313798
[10] [Maio et al., 2019]
[11] [Keinath et al., 2015]
[12] [Smith et al., 2019]

Assembly problem isn't solved

Number of contigs	2nd Gen.	3rd Gen.	# chromosome
Gorilla gorilla gorilla	461,501 [5]	170,105 [6]	24 x 2
Schistosoma japonicum	95,269 [7]	2,108[8]	8 x 2
Escherichia coli	1 [9]	1 [10]	1
Ambystoma mexicanum	1,479,440 [11]	891,205 [12]	14 x 2

[5] [Scally et al., 2012]
[6] [Gordon et al., 2016]
[7] [Schistosoma japonicum Genome Sequencing and Functional Analysis Consortium, 2009]
[8] [Luo et al., 2019]
[9] GenBank Id 6313798
[10] [Maio et al., 2019]
[11] [Keinath et al., 2015]
[12] [Smith et al., 2019]

Assembly problem isn't solved

Number of contigs	2nd Gen.	3rd Gen.	# chromosome
Gorilla gorilla gorilla	461,501 [5]	170,105 [6]	24 x 2
Schistosoma japonicum	95,269 [7]	2,108[8]	8 x 2
Escherichia coli	1 [9]	1 [10]	1
Ambystoma mexicanum	1,479,440 [11]	891,205 [12]	14 x 2

[5] [Scally et al., 2012]
[6] [Gordon et al., 2016]
[7] [Schistosoma japonicum Genome Sequencing and Functional Analysis Consortium, 2009]
[8] [Luo et al., 2019]
[9] GenBank Id 6313798
[10] [Maio et al., 2019]
[11] [Keinath et al., 2015]
[12] [Smith et al., 2019]

Assembly outline

Sequencing

Assembly

Scaffolding
& Evaluation

Assembly outline

Sequencing

Pre-assembly

- Overlapping
- Scrubbing

Assembly

Scaffolding
& Evaluation

Assembly outline

Sequencing

Pre-assembly

- Overlapping
- Scrubbing

Assembly

Post-assembly

Scaffolding & Evaluation

Assembly outline

Sequencing

Pre-assembly

- Overlapping
- Scrubbing

Assembly

Post-assembly

Scaffolding
& Evaluation

Pre-Assembly: `fpa` and `yacrd`

Sequencing

Pre-assembly
- Overlapping
- Scrubbing

Assembly

Post-assembly

Evaluation & Scaffolding

Overlap definition

```
(R1) ACTGAGATGGACTTAGA
               |||||||
          (R2) ACTTAGAGAGGATAGGATA
```

Overlap definition

```
(R1) ACTGAGATGGACTTAGA
               |||||||
          (R2) ACTTAGAGAGGATAGGATA
```

```
(R1) ACTGAGATGGACTTAGA
               ||| | |
          (R3) ACT-ACACATGGTAGTAGAA
```

Overlap definition

```
(R1) ACTGAGATGGACTTAGA
              |||||||
          (R2) ACTTAGAGAGGATAGGATA
```

```
(R1) ACTGAGATGGACTTAGA
             |||  |  |
          (R3) ACT-ACACATGGTAGTAGAA
```

Some third generation overlaping tools: `daligner` [Myers, 2014], `MHAP` [Koren et al., 2017], `Minimap2` [Li, 2016a, Li, 2018].

Some overlaps are too short to be useful

Shaun Jackman October 4, 2018
@sjackman

I have a 1.2 TB PAF.gz file of minimap2 all-vs-all alignments of 18 flowcells of Oxford Nanopore reads.

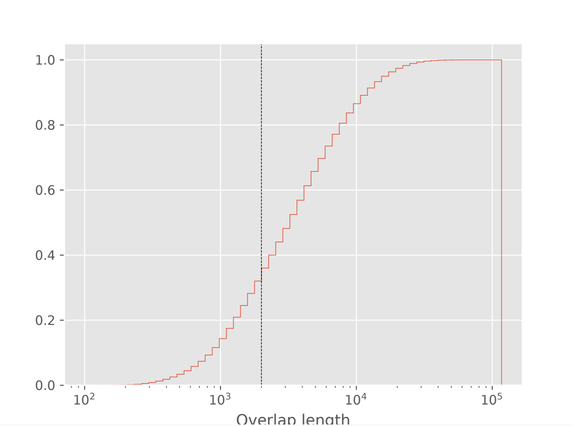
Some overlaps are too short to be useful

In a typical assembly pipeline (`Minimap2`/`Miniasm` [13]), overlap lengths look like this:

[13] [Li, 2016b]

Some overlaps are too short to be useful

In a typical assembly pipeline (`Minimap2`/`Miniasm` [13]), overlap lengths look like this:

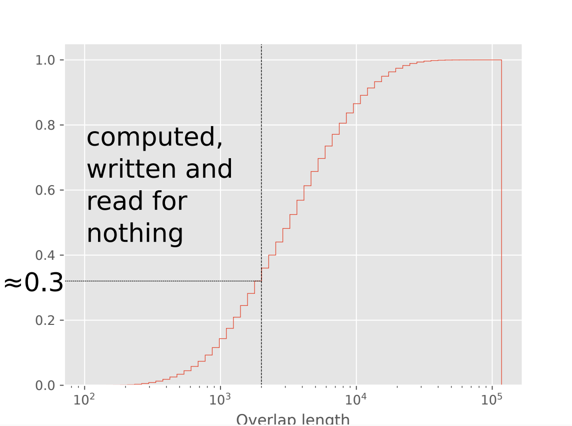


Overlap found by `Minimap2` on dataset SRR8494940 *E. coli* Nanopore 340x

[13] [Li, 2016b]

Some overlaps are too short to be useful

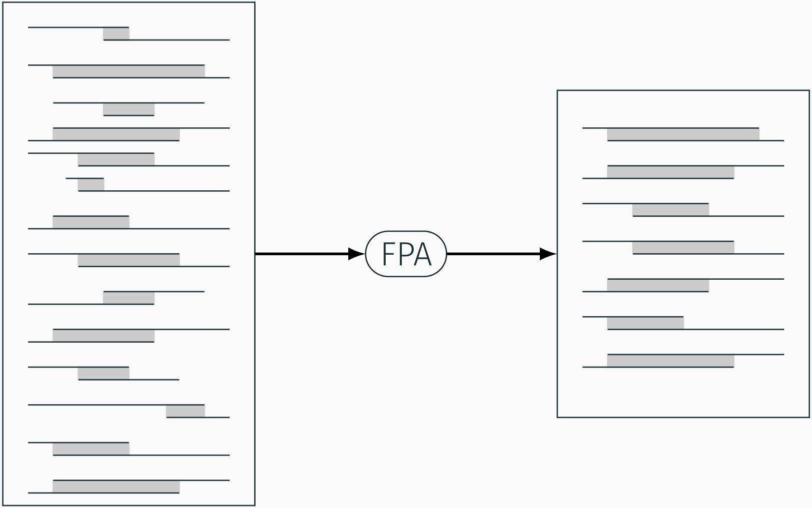
In a typical assembly pipeline (`Minimap2`/`Miniasm` [13]), overlap lengths look like this:



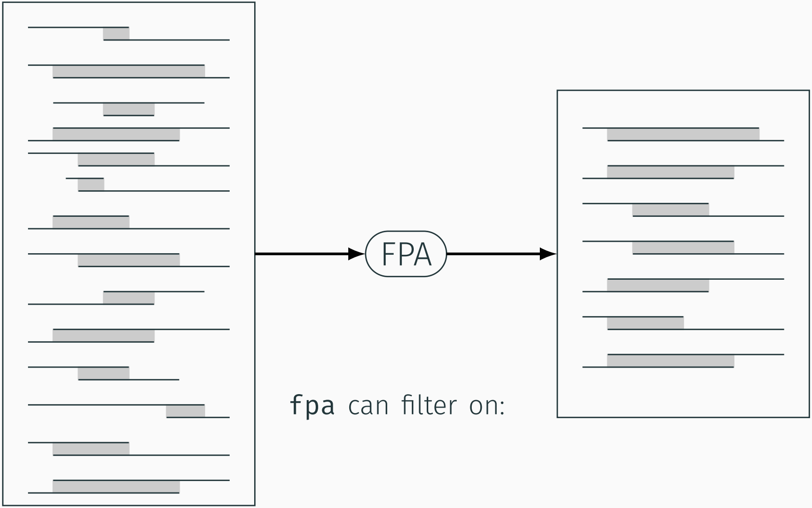
Overlap found by `Minimap2` on dataset SRR8494940 *E. coli* Nanopore 340x

[13] [Li, 2016b]

fpa: Filter Pairwise Alignment

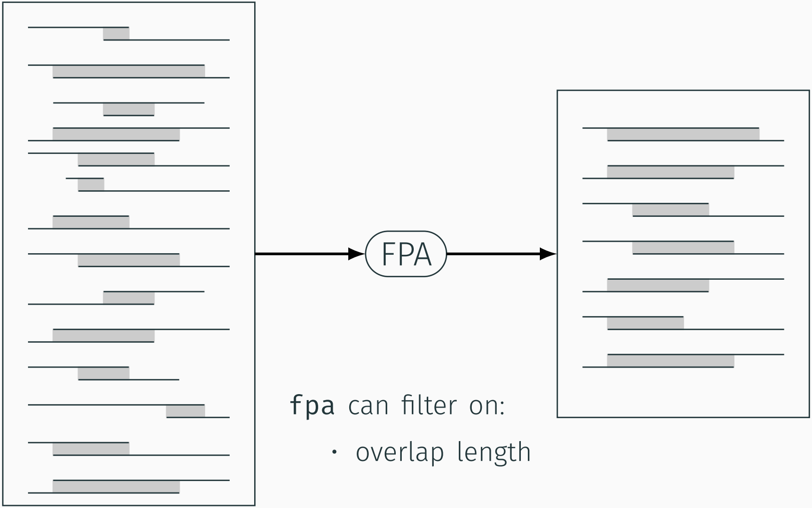


fpa: Filter Pairwise Alignment



fpa can filter on:

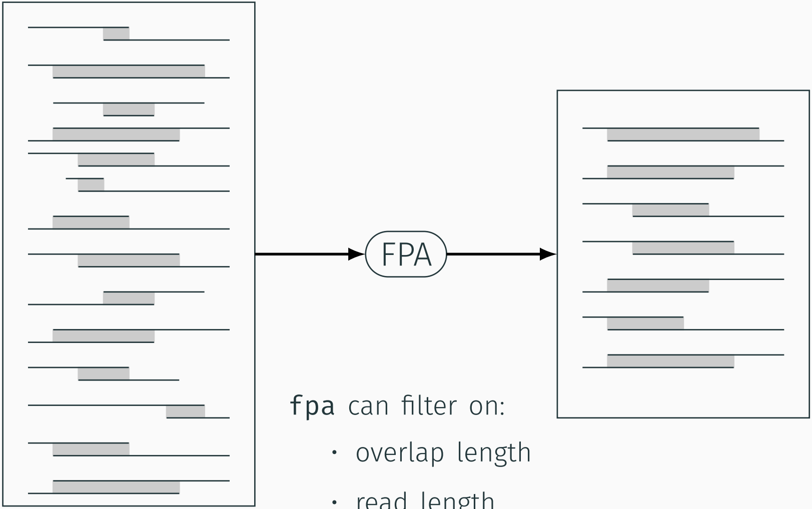
fpa: Filter Pairwise Alignment



fpa can filter on:

- overlap length

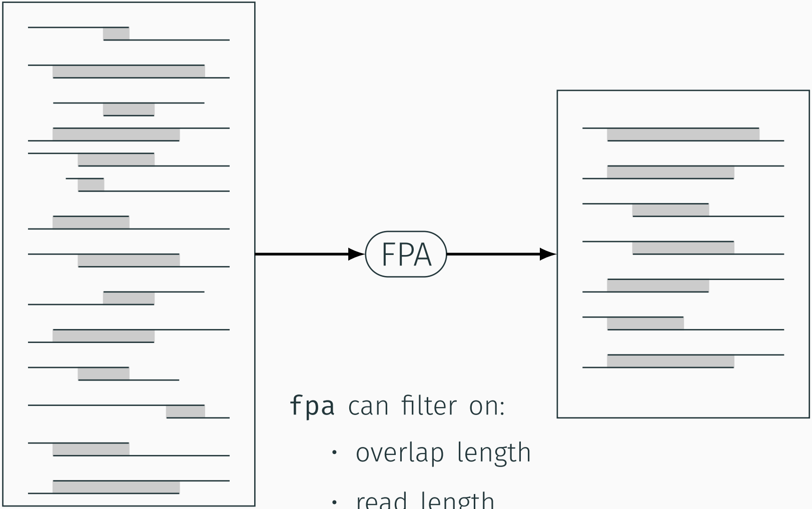
fpa: Filter Pairwise Alignment



fpa can filter on:

- overlap length
- read length

`fpa`: Filter Pairwise Alignment



`fpa` can filter on:

- overlap length
- read length
- overlap type

`fpa` effect on assembly

To study `fpa` effect on downstream analysis we compare two assembly pipelines:

- `Minimap2` → `Miniasm`
- `Minimap2` → `fpa` → `Miniasm`

On two dataset:

- *H. sapiens* chr 1, Nanopore, 30x [14]
- *E. coli*, Nanopore, 50x [15]

[14] [Jain et al., 2018]
[15] [Maio et al., 2019]

fpa effect on assembly

Dataset	*H. sapiens* chr 1		*E. coli*	
Pipeline	w/o `fpa`	`fpa`	w/o `fpa`	`fpa`
Time (s)	3593	3386	30	31
PAF size	32G	9.5G	141M	82M
# contigs	168	150	5	5
contiguity[16]	407821	438055	1450762	1246808

[16] for experts it's NGA50

fpa effect on assembly

Dataset	*H. sapiens* chr 1		*E. coli*	
Pipeline	w/o fpa	fpa	w/o fpa	fpa
Time (s)	3593	≈ 0.9x	30	≈ 1x
PAF size	32G	≈ 0.3x	141M	≈ 0.6x
# contigs	168	≈ 0.9x	5	= 1
contiguity[16]	407821	≈ 1.1x	1450762	≈ 0.9x

[16] for experts it's NGA50

Sequencing

Pre-assembly

- Overlapping
- Scrubbing

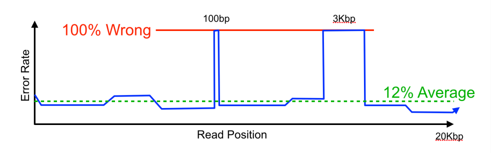
Assembly

Post-assembly

Evaluation & Scaffolding

Error type in third generation reads

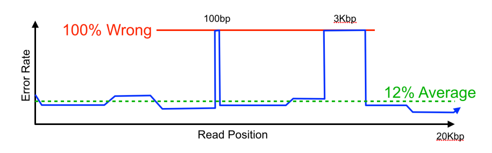
Errors are not homogeneously distributed along the read [17]



[17] [Myers, 2015]
[18] [Wick and Holt, 2019]

Error type in third generation reads

Errors are not homogeneously distributed along the read [17]

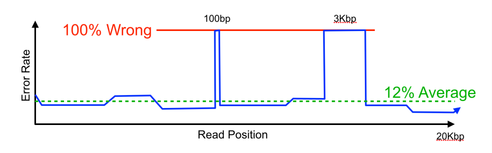


Glitches read [18]

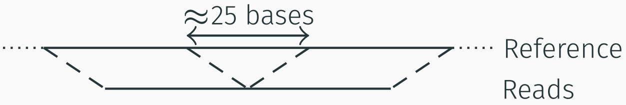
[17] [Myers, 2015]
[18] [Wick and Holt, 2019]

Error type in third generation reads

Errors are not homogeneously distributed along the read [17]



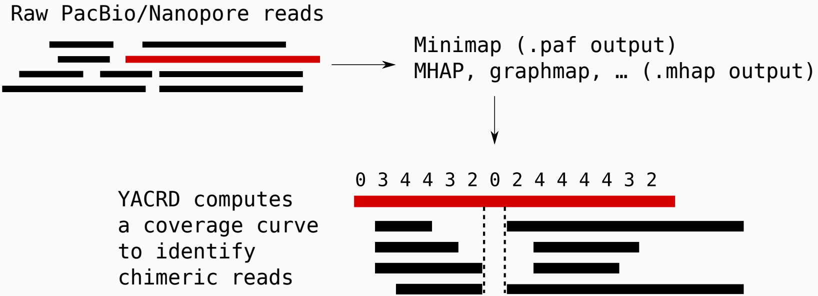
Glitches read [18]



Chimeric read [18]

[17] [Myers, 2015]
[18] [Wick and Holt, 2019]

yacrd: Yet Another Chimeric Read Detector



`yacrd` effect on assembly

To study the effect of `yacrd` we run it on two datasets:

- *H. sapiens* chr 1, Nanopore, 30x [19]
- *E. coli*, Nanopore, 50x [20]

And we run `Minimap2` → `Miniasm` assembly

We compare `yacrd` against two other scrubbing tools:

- `DASCRUBBER` [21]
- `MiniScrub` [22]

---

[19] [Jain et al., 2018]
[20] [Maio et al., 2019]
[21] [Myers, 2017]
[22] [LaPierre et al., 2018]

`yacrd`: Result on reads

Dataset	Scrubber	Error rate	# chimeric reads
H. sapiens chr1	raw	21.05	25888
	`yacrd`	19.01	5216
	`DASCRUBBER`	16.86	1640
E. coli	raw	15.63	351
	`yacrd`	14.34	64
	`DASCRUBBER`	13.07	50
	`MiniScrub`	11.51	58

yacrd: Result on assembly

We present the ratio against the assembly with raw reads

Dataset	Scrubber	contig	contiguity[23]	misassemblies
H. sapiens chr1	yacrd	2x	4x	0.25x
	DASCRUBBER	2x	4x	0.1x
E. coli	yacrd	1x	2x	0.6x
	DASCRUBBER	1x	2x	0.6x
	MiniScrub	9x	0.4x	0.8x

[23] still NGA50

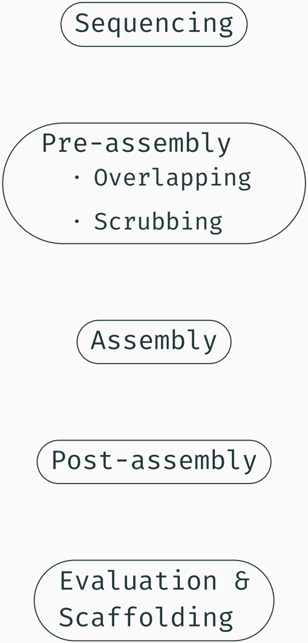
yacrd: Result on assembly

We present the ratio against the assembly with raw reads

Dataset	Scrubber	contig	contiguity[23]	misassemblies
H. sapiens chr1	yacrd	2x	4x	0.25x
	DASCRUBBER	2x	4x	0.1x
E. coli	yacrd	1x	2x	0.6x
	DASCRUBBER	1x	2x	0.6x
	MiniScrub	9x	0.4x	0.8x

Dataset	yacrd	DASCRUBBER	Raw read assembly
H. sapiens chr1	27 mins	3 days 2 hours	$\approx$ 1 hours
E. coli	33 mins	1 days 20 hours	$\approx$ 30 mins

[23] still NGA50


Sequencing
Pre-assembly
· Overlapping
· Scrubbing
Assembly
Post-assembly
Evaluation & Scaffolding

Sequencing

Pre-assembly
- Overlapping
- Scrubbing

Assembly

Post-assembly

Evaluation & Scaffolding

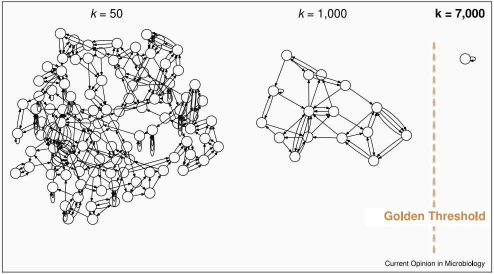
Post-Assembly: `KNOT` Knowledge Network Overlap exTraction

Bacterial *de novo* assembly problem, solved ?

Assembly of 3rd generation sequencing data

- high error rate in reads
- but solves almost all genomic repetitions

Assembly of the *E. coli* genome[24]:



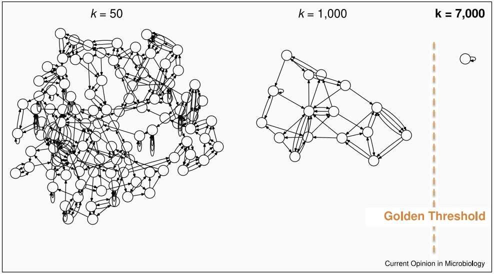
[24] One chromosome, one contig [Koren and Phillippy, 2015]

Bacterial *de novo* assembly problem, solved ?

Assembly of 3rd generation sequencing data

- high error rate in reads
- but solves almost all genomic repetitions

Assembly of the *E. coli* genome[24]:



But in reality ...

[24] One chromosome, one contig [Koren and Phillippy, 2015]

Assembly is solved for many bacteria but not for all

NCTC: 3000 bacteria cultures sequenced with PacBio (read length ≈ 10-20kb), and assembled with HGAP[25]

599 / 1735 (34 %) assemblies are not single-contig (as of Feb 2019)

Species	Strain	Sample	Runs	Automated Assembly	Manual Assembly	Manual Assembly Chromosome Contig Number	Manual Assembly Plasmid Contig Number	Manual Assembly Unidentified Contig Number
Achromobacter xylosoxidans	NCTC10807	ERS451415	ERR550491 ERR550506 ERR550507	Pending	EMBL	1	0	0
Budvicia aquatica	NCTC12282	ERS462988	ERR581162	Pending	EMBL	2	0	0
Campylobacter jejuni	NCTC11351	ERS445056	ERR550473 ERR550476	Pending	EMBL	1	0	0
Cedecea neteri	NCTC12120	ERS462978	ERR581152 ERR581168 ERR597265	Pending	EMBL	7	1	0
Citrobacter amalonaticus	NCTC10805	ERS485850	ERR601566 ERR601575	Pending	EMBL	1	2	0
Citrobacter freundii	NCTC9750	ERS485849	ERR601559 ERR601565	Pending	EMBL	1	0	0
Citrobacter koseri	NCTC10849	ERS473430	ERR581173	Pending	EMBL	1	1	0
Corynebacterium diphtheriae	NCTC11397	ERS451417	ERR550510	Pending	EMBL	1	0	0
Cronobacter sakazakii	NCTC11467	ERS462977	ERR581151 ERR581167	Pending	EMBL	4	3	0

[25] [Chin et al., 2013]

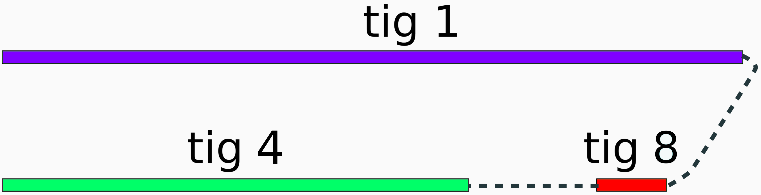
A synthetic example

- **Dataset**: *Terriglobus roseus* synthetic pacbio, 20x coverage (LongISLND[26])
- **Assembly tools**: Canu [27]

[26][Lau et al., 2016]
[27][Koren et al., 2017]

A synthetic example

- **Dataset**: *Terriglobus roseus* synthetic pacbio, 20x coverage (LongISLND[26])
- **Assembly tools**: Canu [27]



Can we recover missing edges between contigs?

[26] [Lau et al., 2016]
[27] [Koren et al., 2017]

A synthetic example

An assembly graph can be defined as :

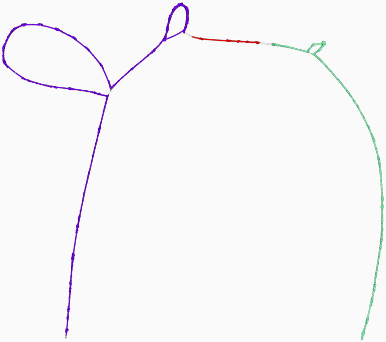
- nodes → reads
- edges → overlaps

[28][Li, 2018]

A synthetic example

An assembly graph can be defined as :

- nodes → reads
- edges → overlaps



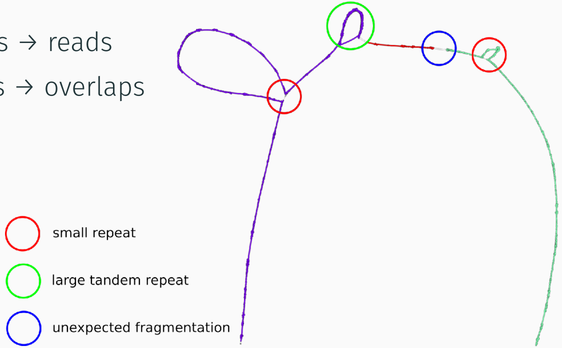
Overlap graph (constructed by `Minimap2` [28]), reads are colored by `Canu` contig.

[28] [Li, 2018]

A synthetic example

An assembly graph can be defined as :

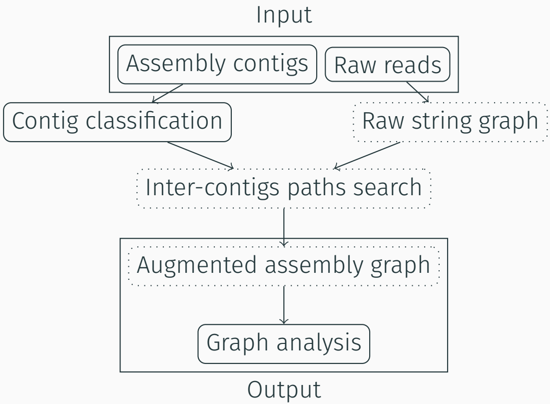
- nodes → reads
- edges → overlaps



Overlap graph (constructed by `Minimap2` [28]), reads are colored by `Canu` contig.

[28] [Li, 2018]

KNOT


Input
Assembly contigs
Raw reads
Contig classification
Raw string graph
Inter-contigs paths search
Augmented assembly graph
Graph analysis
Output

Definition of an Augmented Assembly Graph

The AAG is an undirected, weighted graph:

- nodes: contigs extremities
- edges:
  - between extremities of a contig (weight = 0),
  - paths found between contigs (weight = path length in bases)

Definition of an Augmented Assembly Graph

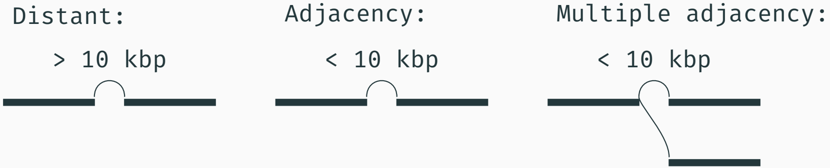
The AAG is an undirected, weighted graph:

- nodes: contigs extremities
- edges:
    - between extremities of a contig (weight = 0),
    - paths found between contigs (weight = path length in bases)

Plain links are paths compatible with true order of contigs, dotted links are other paths.

Graph analysis

We classify paths based on their length (in base pairs):



In prokaryotes, most repetitions are < 10 kbp [29]

[29] [Treangen et al., 2009]

Test on 38 datasets from NCTC3000

We selected 38 datasets from NCTC3000, where `Canu`, `Miniasm` and `Hinge` didn't produce the expected number of chromosomes (*i.e. unsolved assemblies*).

- 19 datasets were *manually solved* by NCTC
- 17 remained fragmented
- 2 with no assembly attempt by NCTC

Result

Across 38 datasets:

Mean number of	
Canu contigs	4.32
Edges in AAG	32.67
Theoretical max. edges in AAG	41.83
Distant edges	28.64
Adjacency edges	4.02
Missing adjacency in:	
Canu contigs graph	4.94
AAG, adjacency edges	2.70

Result

Across 38 datasets:

Mean number of	
Canu contigs	4.32
Edges in AAG	32.67
Theoretical max. edges in AAG	41.83
Distant edges	28.64
Adjacency edges	4.02
Missing adjacency in:	
Canu contigs graph	4.94
AAG, adjacency edges	2.70

Almost half of the missing paths in contigs graph are recovered.

Hamilton walk

AAG's are generally complete graphs. We can enumerate all their Hamilton walks.

The weight of a walk is the sum of all edge weights.

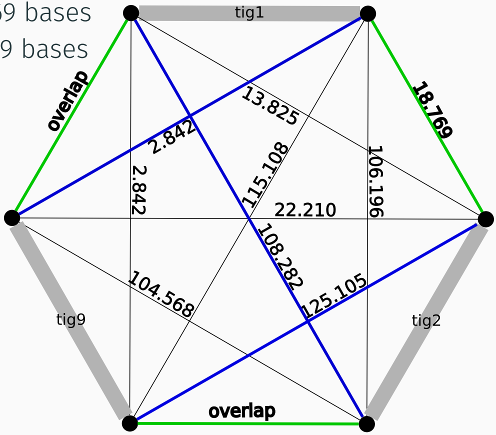
Hamilton walk

AAG's are generally complete graphs. We can enumerate all their Hamilton walks.

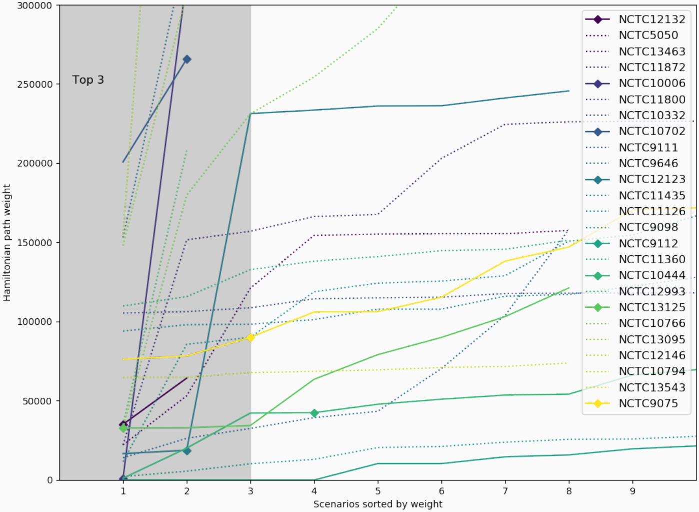
The weight of a walk is the sum of all edge weights.

Supposedly: We assume that **lowest-weight walk** is the true genome.

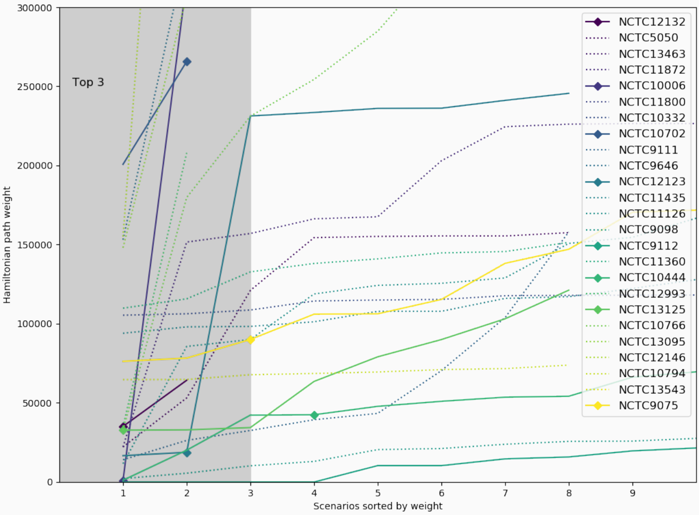
- Green walk weight: 18,769 bases
- Blue walk weight: 136,229 bases



Hamilton walk



Hamilton walk



Generally, the true contig ordering is a low-weight Hamiltonian walk

Conclusion

Summary: `yacrd` and `fpa`

`fpa` allows users to reduce the memory impact of overlap files without impact on assembly and was used:

[30] https://github.com/ekg/yeast-pangenome
[31] https://github.com/natir/yacrd/issues/30

Summary: `yacrd` and `fpa`

`fpa` allows users to reduce the memory impact of overlap files without impact on assembly and was used:

- in a genome graph pipeline generation [30] to keep only very long overlap
- `KNOT` pipeline to convert overlap into overlap graph

[30] https://github.com/ekg/yeast-pangenome
[31] https://github.com/natir/yacrd/issues/30

Summary: `yacrd` and `fpa`

`fpa` allows users to reduce the memory impact of overlap files without impact on assembly and was used:

- in a genome graph pipeline generation [30] to keep only very long overlap
- `KNOT` pipeline to convert overlap into overlap graph

`yacrd` improves `Miniasm` and `Wtdbg2` quality with a limited effect on assembly pipeline computation time and was used:

[30] https://github.com/ekg/yeast-pangenome
[31] https://github.com/natir/yacrd/issues/30

Summary: yacrd and fpa

fpa allows users to reduce the memory impact of overlap files without impact on assembly and was used:

- in a genome graph pipeline generation [30] to keep only very long overlap
- KNOT pipeline to convert overlap into overlap graph

yacrd improves Miniasm and Wtdbg2 quality with a limited effect on assembly pipeline computation time and was used:

- to remove chimera in a long read metagenome characterization pipeline [Cuscó et al., 2018]
- to improve some flye assembly[31]

[30] https://github.com/ekg/yeast-pangenome
[31] https://github.com/natir/yacrd/issues/30

Summary: `yacrd` and `fpa`

`fpa` allows users to reduce the memory impact of overlap files without impact on assembly and was used:

- in a genome graph pipeline generation [30] to keep only very long overlap
- `KNOT` pipeline to convert overlap into overlap graph

`yacrd` improves `Miniasm` and `Wtdbg2` quality with a limited effect on assembly pipeline computation time and was used:

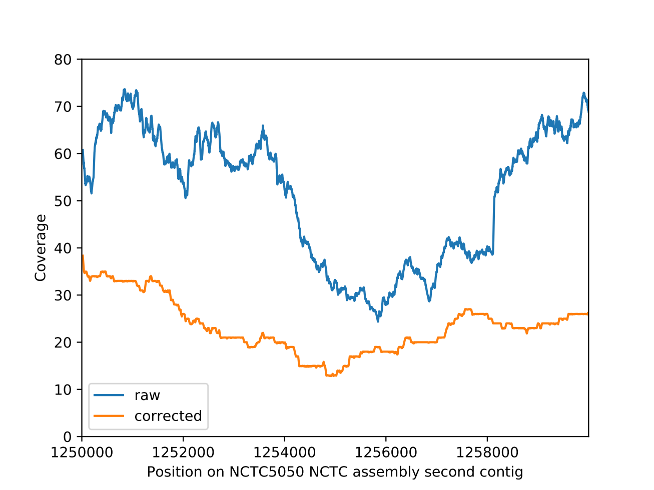
- to remove chimera in a long read metagenome characterization pipeline [Cuscó et al., 2018]
- to improve some `flye` assembly[31]

I'm still not satisfied

[30] https://github.com/ekg/yeast-pangenome
[31] https://github.com/natir/yacrd/issues/30

Summary: yacrd and fpa

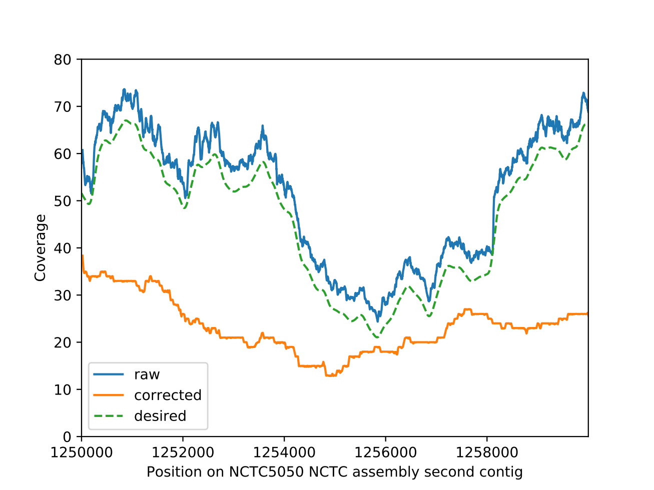
Scrubbing or correcting reads can create a coverage gap



Correction performed by the Canu correction module

Summary: `yacrd` and `fpa`

Scrubbing or correcting reads can create a coverage gap



Correction performed by the Canu correction module

Summary: KNOT

The KNOT AAG help to understand and improve assembly without any new information.

- Bacterial assembly is not solved for all datasets
- Build and analyse **Augmented Assembly Graph** can help

Future:

- Reduce the computation time
- Get more users

Open questions:

- Behavior of the AAG on heterozygote dataset
- How to adapt to multichromosomal species

Outlook

Publications:

- Graph analysis of fragmented long-read bacterial genome assemblies `doi: 10.1093/bioinformatics/btz219`
- yacrd and fpa: upstream tools for long-read genome assembly `doi: 10.1101/674036`

Blog posts:

- State-of-the-art long reads overlapper-compare
- How to reduce the impact of your PAF file on your disk by 95%
- Misassemblies in noisy assemblies

Software:

- KNOT `https://github.com/natir/knot/`
- `yacrd` `https://github.com/natir/yacrd/`
- `fpa` `https://github.com/natir/fpa/`

Perspectives

"With modern fast sequencing techniques and suitable computer programs it is now possible to sequence whole genomes with-out the need of restriction maps."*

* Adapted from R. Chikhi talk, CGSI 2019**

** Adapted from A. Phillippy's talk, RECOMB-Seq'19 [32]

[32] [Staden, 1979]

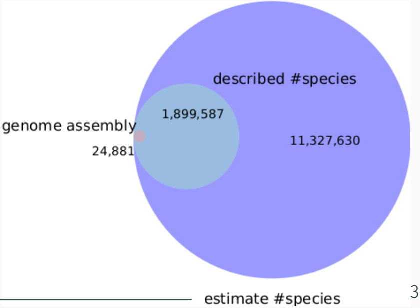
[33] data extract from ebi database and [Chapman, 2009]

Perspectives

"With modern fast sequencing techniques and suitable computer programs it is now possible to sequence whole genomes with-out the need of restriction maps."*

* Adapted from R. Chikhi talk, CGSI 2019**

** Adapted from A. Phillippy's talk, RECOMB-Seq'19 [32]



[33]

[32][Staden, 1979]
[33]data extract from ebi database and [Chapman, 2009]

Acknowledgements

First, members of my jury and:

- Jean-Stéphane Varré & Rayan Chikhi
- The BONSAI team
- All staff members of:
  - CRISTAL laboratory
  - Inria Lille Nord Europe center
  - University of Lille

Finally, my friends and familly.

References i

Cairns, J. (1963).
The bacterial chromosome and its manner of replication as seen by autoradiography.
Journal of Molecular Biology, 6(3):208–IN5.

Chapman, A. (2009).
Numbers of Living Species in Australia and the World 2nd edn.

Chin, C.-S., Alexander, D. H., Marks, P., Klammer, A. A., Drake, J., Heiner, C., Clum, A., Copeland, A., Huddleston, J., Eichler, E. E., Turner, S. W., and Korlach, J. (2013).
Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data.
Nature Methods, 10(6):563–569.

References ii

Cuscó, A., Catozzi, C., Viñes, J., Sanchez, A., and Francino, O. (2018).
Microbiota profiling with long amplicons using nanopore sequencing: full-length 16s rRNA gene and whole rrn operon.
F1000Research, 7:1755.

Franklin, R. E. and Gosling, R. G. (1953).
Molecular configuration in sodium thymonucleate.
Nature, 171(4356):740–741.

Gordon, D., Huddleston, J., Chaisson, M. J. P., Hill, C. M., Kronenberg, Z. N., Munson, K. M., Malig, M., Raja, A., Fiddes, I., Hillier, L. W., Dunn, C., Baker, C., Armstrong, J., Diekhans, M., Paten, B., Shendure, J., Wilson, R. K., Haussler, D., Chin, C.-S., and Eichler, E. E. (2016).
Long-read sequence assembly of the gorilla genome.
Science, 352(6281):aae0344–aae0344.

Jain, M., Koren, S., Miga, K. H., Quick, J., Rand, A. C., Sasani, T. A., Tyson, J. R., Beggs, A. D., Dilthey, A. T., Fiddes, I. T., Malla, S., Marriott, H., Nieto, T., O'Grady, J., Olsen, H. E., Pedersen, B. S., Rhie, A., Richardson, H., Quinlan, A. R., Snutch, T. P., Tee, L., Paten, B., Phillippy, A. M., Simpson, J. T., Loman, N. J., and Loose, M. (2018).
Nanopore sequencing and assembly of a human genome with ultra-long reads.
Nature Biotechnology, 36(4):338–345.

Keinath, M. C., Timoshevskiy, V. A., Timoshevskaya, N. Y., Tsonis, P. A., Voss, S. R., and Smith, J. J. (2015).
Initial characterization of the large genome of the salamander ambystoma mexicanum using shotgun and laser capture chromosome sequencing.
Scientific Reports, 5(1).

Koren, S. and Phillippy, A. M. (2015).
One chromosome, one contig: complete microbial genomes from long-read sequencing and assembly.
Current Opinion in Microbiology, 23:110–120.

Koren, S., Walenz, B. P., Berlin, K., Miller, J. R., Bergman, N. H., and Phillippy, A. M. (2017).
Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation.
Genome Research, 27(5):722–736.

LaPierre, N., Egan, R., Wang, W., and Wang, Z. (2018).
MiniScrub: de novo long read scrubbing using approximate alignment and deep learning.
bioRxiv.

Lau, B. et al. (2016).
LongISLND:in silicosequencing of lengthy and noisy datatypes.
Bioinformatics, 32(24):3829–3832.

Li, H. (2016a).
Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences.
Bioinformatics, 32(14):2103–2110.

Li, H. (2016b).
`Minimap2` and `Miniasm`: Fast mapping and *de novo* assembly for noisy long sequences.
Bioinformatics, 32(14):2103–2110.

Li, H. (2018).
Minimap2: pairwise alignment for nucleotide sequences.
Bioinformatics, 34(18):3094–3100.

Luo, F., Yin, M., Mo, X., Sun, C., Wu, Q., Zhu, B., Xiang, M., Wang, J., Wang, Y., Li, J., Zhang, T., Xu, B., Zheng, H., Feng, Z., and Hu, W. (2019).
An improved genome assembly of the fluke schistosoma japonicum.
PLOS Neglected Tropical Diseases, 13(8):e0007612.

Maio, N. D., Shaw, L. P., Hubbard, A., George, S., Sanderson, N., Swann, J., Wick, R., AbuOun, M., Stubberfield, E., Hoosdally, S. J., Crook, D. W., Peto, T. E. A., Sheppard, A. E., Bailey, M. J., Read, D. S., Anjum, M. F., Walker, A. S., and and, N. S. (2019).
Comparison of long-read sequencing technologies in the hybrid assembly of complex bacterial genomes.
bioRxiv.

- Marijon, P., Chikhi, R., and Varré, J.-S. (2019a).
  Graph analysis of fragmented long-read bacterial genome assemblies.
  Bioinformatics.
- Marijon, P., Chikhi, R., and Varré, J.-S. (2019b).
  yacrd and fpa: upstream tools for long-read genome assembly.
  bioRxiv.
- Myers, G. (2014).
  Daligner: Fast and sensitive detection of all pairwise local alignments.
  `https://dazzlerblog.wordpress.com/2014/07/10/dalign-fast-and-sensitive-detection-of-all-pairwise-local-alignme`

Myers, G. (2015).
Intrinsic quality values.
https://dazzlerblog.wordpress.com/2015/11/06/intrinsic-quality-values/.

Myers, G. (2017).
Scrubbing reads for better assembly.
https://dazzlerblog.wordpress.com/2017/04/22/1344/.

Scally, A., Dutheil, J. Y., Hillier, L. W., Jordan, G. E., Goodhead, I., Herrero, J., Hobolth, A., Lappalainen, T., Mailund, T., Marques-Bonet, T., McCarthy, S., Montgomery, S. H., Schwalie, P. C., Tang, Y. A., Ward, M. C., Xue, Y., Yngvadottir, B., Alkan, C., Andersen, L. N., Ayub, Q., Ball, E. V., Beal, K., Bradley, B. J., Chen, Y., Clee, C. M., Fitzgerald, S., Graves, T. A., Gu, Y., Heath, P., Heger, A., Karakoc, E., Kolb-Kokocinski, A., Laird, G. K., Lunter, G., Meader, S., Mort, M.,

Mullikin, J. C., Munch, K., O'Connor, T. D., Phillips, A. D., Prado-Martinez, J., Rogers, A. S., Sajjadian, S., Schmidt, D., Shaw, K., Simpson, J. T., Stenson, P. D., Turner, D. J., Vigilant, L., Vilella, A. J., Whitener, W., Zhu, B., Cooper, D. N., de Jong, P., Dermitzakis, E. T., Eichler, E. E., Flicek, P., Goldman, N., Mundy, N. I., Ning, Z., Odom, D. T., Ponting, C. P., Quail, M. A., Ryder, O. A., Searle, S. M., Warren, W. C., Wilson, R. K., Schierup, M. H., Rogers, J., Tyler-Smith, C., and Durbin, R. (2012).
Insights into hominid evolution from the gorilla genome sequence.
Nature, 483(7388):169–175.

Schistosoma japonicum Genome Sequencing and Functional Analysis Consortium (2009).
The schistosoma japonicum genome reveals features of host–parasite interplay.
Nature, 460(7253):345–351.

Smith, J. J., Timoshevskaya, N., Timoshevskiy, V. A., Keinath, M. C., Hardy, D., and Voss, S. R. (2019).
A chromosome-scale assembly of the axolotl genome.
Genome Research, 29(2):317–324.

Staden, R. (1979).
A strategy of DNA sequencing employing computer programs.
Nucleic Acids Research, 6(7):2601–2610.

Treangen, T. J., Abraham, A.-L., Touchon, M., and Rocha, E. P. (2009).
Genesis, effects and fates of repeats in prokaryotic genomes.
FEMS Microbiology Reviews, 33(3):539–571.

Wick, R. and Holt, K. E. (2019).
rrwick/Long-read-assembler-comparison: Initial release.