

# HairSplitter: assembling long reads in an unknown number of haplotypes

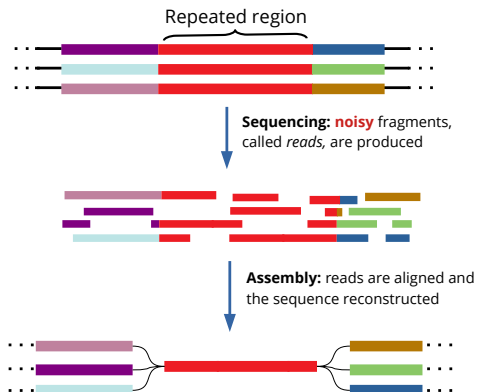
Roland Faure<sup>1,2</sup>, Jean-François Flot<sup>1</sup>, Dominique Lavenier<sup>2</sup>

<sup>1</sup>Université libre de Bruxelles (ULB)

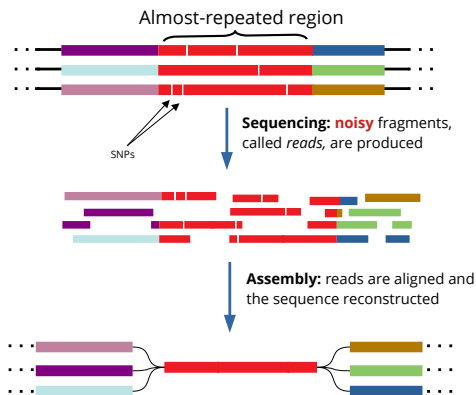
<sup>2</sup>Université de Rennes, IRISA

November 2022

## Genome assembly

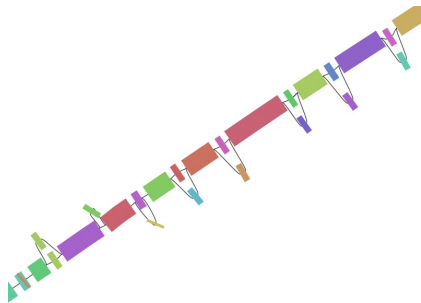


## Genome assembly: similar regions get collapsed



- ▶ When divergence is small compared to the error rate of reads, SNPs are discarded as errors

## Genome assembly: similar regions get collapsed



Screenshot of the Flye assembly of diploid *Adineta vaga*

- ▶ Loss of heterozygous information!

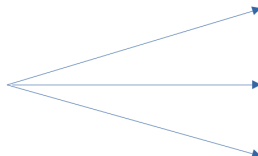
## Obtaining uncollapsed assemblies

- ▶ We want to recover the lost diversity

## Obtaining uncollapsed assemblies

- ▶ We want to recover the lost diversity
- ▶ State of the art (long reads): phase the collapsed contigs using HapDup, WhatsHap, HapCut, H-PopG...

```
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACGGTGTCCCTATGGAGCG--TCGCGACCGTA
AACTGTGTCCTATAGAGCGATAACGCGACCGTA
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACTGTGACCCCTATAGAGCGATAACGCGACCGTA
AACTGTGTCCT-TAGAGCGATTTCGCAAGCGTA
AACTGCGTCCCTATAGAGCGATAACGCGACCGTA
```



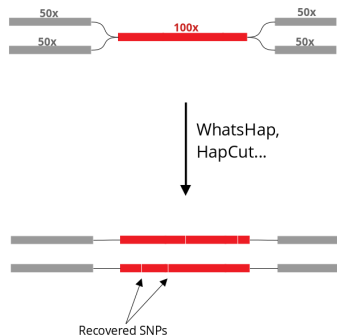
```
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACTGTGTCCT-TAGAGCGATTTCGCAAGCGTA
AACGGTGTCCCTATGGAGCG--TCGCGACCGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACTGCGTCCCTATAGAGCGATAACGCGACCGTA
AACTGTGACCCCTATAGAGCGATAACGCGACCGTA
AACTGTGTCCTATAGAGCGATAACGCGACCGTA
```

**Input:** All reads, the draft assembly and the number of groups  
**Output:** Reads split into groups

Let's recover the lost diversity!



## Let's recover the lost diversity!

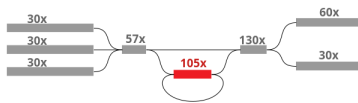




## Let's recover the lost diversity!



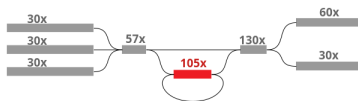
WhatsHap,  
HapCut...



## Let's recover the lost diversity!

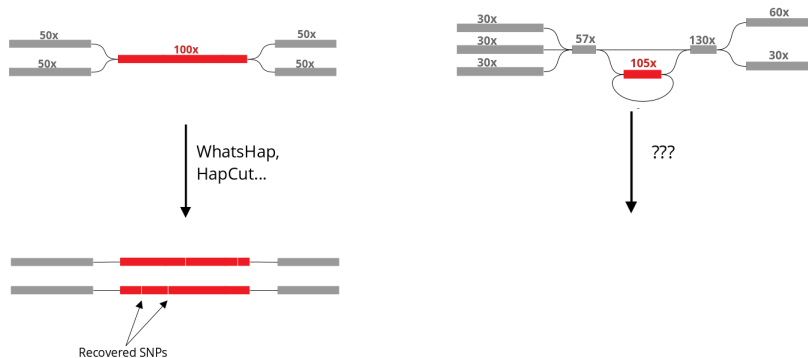


WhatsHap,  
HapCut...



???

## Let's recover the lost diversity!



- ▶ In many cases the copy-number of a contig is unknown: polyploid genomes, repeats, metagenomic assemblies...

# HairSplitter

```
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACGGTGTCCCTATGGAGCG--TCGCGACCGTA
AACTGTGTCCTATAGAGCGATAACGCGACCGTA
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACTGTGACCCCTATAGAGCGATAACGCGACCGTA
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACTGCGTCCCTATAGAGCGATAACGCGACCGTA
```



**Input:** All reads, the draft assembly and the number of groups  
**Output:** Reads split into groups

```
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
AACTGTGTCCT-TAGAGCGATTTCGCGACGTA
```

```
AACGGTGTCCCTATGGAGCG--TCGCGACCGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
```

```
AACTGCGTCCCTATAGAGCGATAACGCGACCGTA
AACTGTGACCCCTATAGAGCGATAACGCGACCGTA
AACTGTGTCCTATAGAGCGATAACGCGACCGTA
```

# HairSplitter

```
AACTGTGTCCT-TAGAGCGATTTCGCGAGCGTA  
AACGGTGTCCCTATGGAGCG--TCGCGACCGTA  
AACTGTGTCCTATAGAGCGATACGCGACCGTA  
AACTGTGTCCT-TAGAGCGATTTCGCGAGCGTA  
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA  
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA  
AACTGTGACCCCTATAGAGCGATACGCGACCGTA  
AACTGTGTCCT-TAGAGCGATTTCGCAAGCGTA  
AACTGCGTCCCTATAGAGCGATACGCGACCGTA
```



**Input:** All reads, the draft assembly  
and the number of groups  
**Output:** Reads split into groups

```
AACTGTGTCCT-TAGAGCGATTTCGCGAGCGTA  
AACTGTGTCCT-TAGAGCGATTTCGCGAGCGTA  
AACTGTGTCCT-TAGAGCGATTTCGCAAGCGTA
```

```
AACGGTGTCCCTATGGAGCG--TCGCGACCGTA  
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA  
AACGGTGTCCCTATAGAGCGATTTCGCGACCGTA
```

```
AACTGCGTCCCTATAGAGCGATACGCGACCGTA  
AACTGTGACCCCTATAGAGCGATACGCGACCGTA  
AACTGTGTCCTATAGAGCGATACGCGACCGTA
```

- ▶ *Hairsplitter*: a person who argues about differences that are too small to be important - *Britannica.com*

# Splitting contigs: algorithm

```
ref AACTGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCGGAAGCTGAAGTGT
r1  AACTGTGTCCCT-TAGAGCGATTTCGCGAGCGTATCTCGGAAGCTGAAGTGT
r2  AACGGTGTCCATATGGAGCG--TCGCGACCGTATCTCGAAAGCAGAAGTGT
r3  AACTGTGTCCCTATAGAGCGATACGCGACCGTACCTCGGAAGCTGAA-TGT
r4  AACTGTGTCCAT-TAGAGCGATTTCGCGAGCGTATCTCGGAAGCTGAAGTGT
r5  AACGGTGTCCATATAGAGCGATTTCGCGACCGTACCTCGAAAGCTGAAGTGT
r6  AACGGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCGAAAGCAGAAGTGT
r7  AACTGTGACCCTATAGAGCGATACGCGACCGTACCTCGGAAGCAGAA-TGT
r8  AACTGTGTCCAT-TAGAGCGATTTCGCAAGCGTACCTCGGAAGCTGAAGTGT
r9  AACTGCGTCCCTATAGAGCGATACGCGACCGTACCTCGGAAGCAGAA-TGT
```

- ▶ Reads are aligned on the (collapsed) contig

## Splitting contigs: algorithm

```

ref  AACTGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCGGAAGCTGAAGTGT
r1   AACTGTGTCCCT--TAGAGCGATTTCGCGAGCGTATCTTCGGAAGCTGAAGTGT
r2   AACGGGTGTCCATATGGAGCG--TCGCGACCGTATCTTCGAAAGCAAAGTGT
r3   AACTGTGTCCCTATAGAGCGATACGCGACCGTACCTCGGAAGCTGAA--TGT
r4   AACTGTGTCCA--TAGAGCGATTTCGCGAGCGTACTTCGGAAGCTGAAGTGT
r5   AACGGTGTCCATATAGAGCGATTTCGCGACCGTACCTCCAAAGCTGAAGTGT
r6   AACGGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCCAAAGCAAAGTGT
r7   AACTGTGTACCCTATAGAGCGATACGCGACCGTACCTCGGAAGCAAA--TGT
r8   AACTGTGTCCA--TAGAGCGATTTCGAGCGTACCTCGGAAGCTGAAGTGT
r9   AACTCGTCCCTATAGAGCGATACGCGACCGTACCTCGGAAGCAAA--TGT
    
```

- Positions with high divergence are selected

# Splitting contigs: algorithm

```

ref  AACTGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCGGAAGCTGAAGTGT
r1   AACTGTGTCCCT--TAGAGCGATTTCGCGACCGTATCTCTCGGAAGCTGAAGTGT
r2   AACGGTGTCCATAATGGAGCG--TCGCGACCGTATCTCTCGAAGCTGAAGTGT
r3   AACTGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCGGAAGCTGAA--TGT
r4   AACTGTGTCCATA--TAGAGCGATTTCGCGACCGTATCTCTCGGAAGCTGAAGTGT
r5   AACGGTGTCCATAATAGAGCGATTTCGCGACCGTACCTCGAAGCTGAAGTGT
r6   AACGGTGTCCCTATAGAGCGATTTCGCGACCGTACCTCGAAGCTGAAGTGT
r7   AACTGTGTACCCATAAGAGCGATTTCGCGACCGTACCTCGGAAGCTGAAG--TGT
r8   AACTGTGTCCATA--TAGAGCGATTTCGAGCGTACCTCGGAAGCTGAAGTGT
r9   AACTTCGTCCCTATAGAGCGATTTCGCGACCGTACCTCGGAAGCTGAAG--TGT
  
```



(TGTGGTTT): {r2,r5,r6}, {r1,r3,r4,r7,r8,r9}  
 (CACACCAC): {r2,r4,r5,r8}, {r1,r3,r6,r7,r9}  
 (-AA-AAA-A): {r1,r4,r8}, {r2,r3,r5,r6,r7,r9}  
 (TTATTTATA): {r3,r7,r9}, {r1,r2,r4,r5,r6,r8}  
 (GCCGCCCGC): {r1,r4,r8}, {r2,r3,r5,r6,r7,r9}

(TTCTCCCC): {r1,r2,r4}, {r3,r5,r6,r7,r8,r9}  
 (GAGGAAGGG): {r2,r5,r6}, {r1,r3,r4,r7,r8,r9}  
 (TATTTAATA): {r2,r6,r7,r9}, {r1,r3,r4,r5,r8}  
 (GG-GGA-G-): {r3,r7,r9}, {r1,r2,r4,r5,r8}

- ▶ Each position partitions the reads in two groups



## Splitting contigs: algorithm

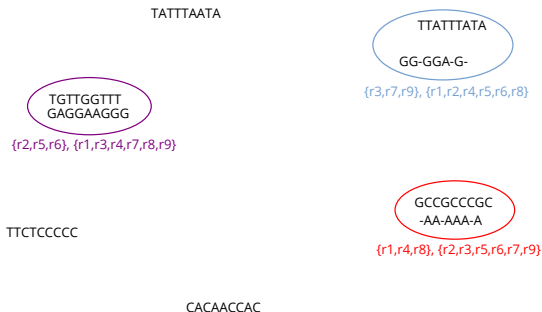
- ▶ We can use the *V-measure* ( $V$ ) between two partitions to define a distance  $d$  between two positions:

$$d(TGTTGGTTT, GAGGAAGGG) = V\left(\begin{matrix} \{r2, r5, r6\} \{r1, r3, r4, r7, r8, r9\} \\ \{r2, r5, r6\} \{r1, r3, r4, r7, r8, r9\} \end{matrix}\right) = 0$$

$$d(TGTTGGTTT, CACAACCAC) = V\left(\begin{matrix} \{r2, r5, r6\} \{r1, r3, r4, r7, r8, r9\} \\ \{r2, r4, r5, r8\} \{r1, r3, r6, r7, r9\} \end{matrix}\right) = 0.16$$

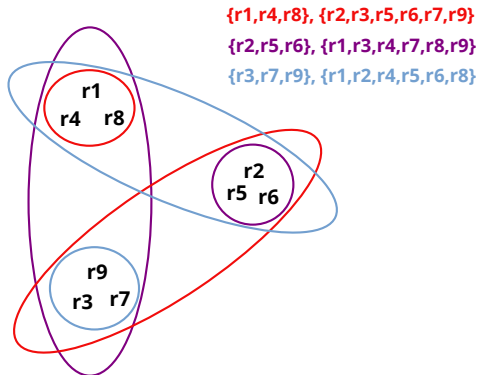
- ▶ Use this distance to cluster positions

## Splitting contigs: algorithm



- ▶ Recurring partitions correspond to SNPs
- ▶ Isolated positions correspond to error-prone positions (the probability of two random partitions being close decreases exponentially with coverage )

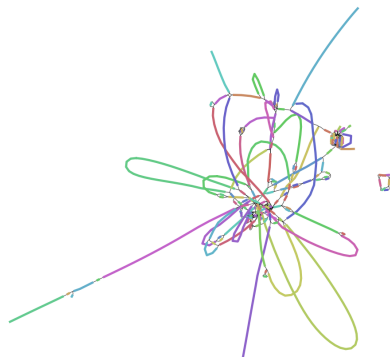
## Splitting contigs: algorithm



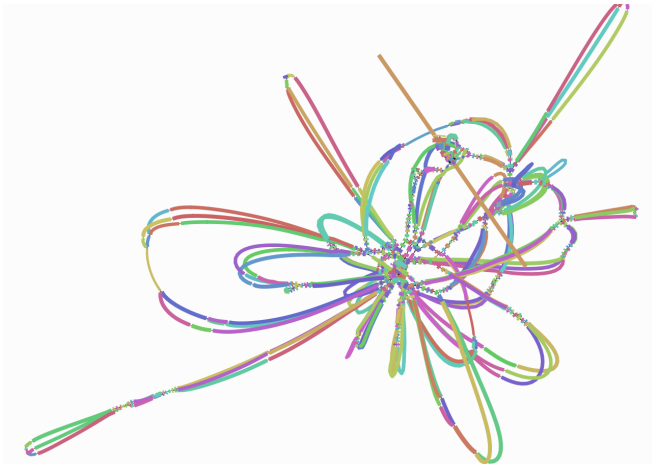
- ▶ Reads are separated, then the 3 versions of the contig are created using Racon

## Dataset

- ▶ *In silico* mix of 3 strains of *Saccharomyces cerevisiae*
- ▶ Nanopore reads with  $> 5\%$  error rate
- ▶ Assembled with Flye  
--keep-haplotypes
- ▶ We get a collapsed assembly



# Result!



## Result!

- ▶ Our “solution genome” = the 3 genomes assembled separately

	Size of assembly (Mbp)	Missing k-mers	Missing haplotype-specific k-mers
Solution	35.8	-	-
Before Hairsplitter	12.7	29.4 %	55.3 %
After Hairsplitter	33.9	8.5 %	16 %

## Comparison with WhatsHap polyphase

- ▶ edge\_132 of the assembly

	proportion of unassigned reads	proportion of mis-assigned reads
HairSplitter	0%	4.5%
Whatshap-polyphase	51%	8.2%

- ▶ HairSplitter much faster because does not need variant calling

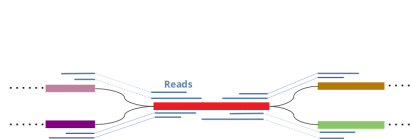
## Splitting the assembly

- ▶ First phase all contigs

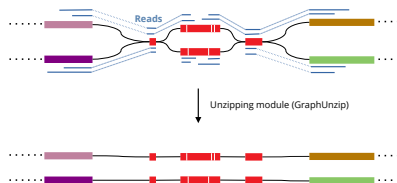


## Splitting the assembly

- ▶ First phase all contigs
- ▶ Then improve the contiguity of the assembly!



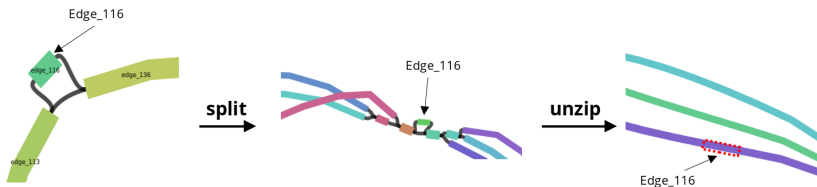
Before HairSplitter



After HairSplitter

# Result!

- ▶ Example on the triploid yeast



- ▶ N50 went from 373 to 465kbp (assembled separately: 767kbp)

## Pros and cons of HairSplitter

### Limitations of HairSplitter:

- ▶ Not *very* fast: it re-polishes the whole assembly
- ▶ Limited in the number of haplotypes

### Strengths of HairSplitter:

- ▶ Very modular, can be used with any assembler
- ▶ Naive: makes no assumption on ploidy, parameter-free
- ▶ Safe: won't artificially duplicate contigs

## Take-home message

- ▶ HairSplitter **splits collapsed assemblies** from “draft” assemblies obtained by any means
- ▶ HairSplitter can **recover haplotypes** and **distinguish repeated elements**
- ▶ Only needs **sequencing reads**, potentially error-prone

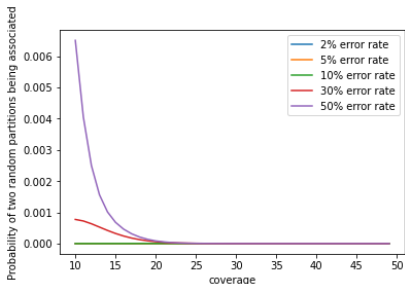
## Take-home message

- ▶ HairSplitter **splits collapsed assemblies** from “draft” assemblies obtained by any means
- ▶ HairSplitter can **recover haplotypes** and **distinguish repeated elements**
- ▶ Only needs **sequencing reads**, potentially error-prone
- ▶ Not really available yet ([github.com/RolandFaure/HairSplitter](https://github.com/RolandFaure/HairSplitter))

## Acknowledgements

- ▶ Dominique Lavenier and Jean-François Flot for their supervision
- ▶ The EEB-EBE and GenScale teams

## Similarity between random partitions



Probability of two positions being clustered together by HS, in function of the error rate at these position

- ▶ With high coverage, the erroneous positions won't cluster