Hairsplitter: Separating noisy long reads into an unknown number of haplotypes
Roland Faure, Jean-François Flot, Dominique Lavenier

To cite this version:
Roland Faure, Jean-François Flot, Dominique Lavenier. Hairsplitter: Separating noisy long reads into an unknown number of haplotypes. Genome Informatics 2022, Sep 2022, London / Virtual, United Kingdom. pp.1-1. hal-03817928

HAL Id: hal-03817928
https://hal.science/hal-03817928
Submitted on 17 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
**Hairsplitter: Separating noisy long reads into an unknown number of haplotypes**

Roland Faure\textsuperscript{1, 2}, Jean-François Flot\textsuperscript{1}, Dominique Lavenier\textsuperscript{2}

1. Service Evolution Biologique et Ecologie, ULB, Brussels, Belgium 2. Univ. Rennes, Inria RBA, CNRS UMR 6074, Rennes, France

### Problem: assembling similar sequences

- **Almost-repeated region**
- **Sequencing noisy fragments, called reads, are produced**
- **Assembly: reads are aligned and the sequence reconstructed**

**PROBLEM!**

Similar regions get collapsed in a single sequence

### State of the art

**Input:** All reads, the draft assembly and the number of groups we seek

**Output:** Reads split into groups

**Existing software:** WhatsHap (polyphase), Phasebook, Falcon-Unzip, HapCut, HapDup...

**But:** All software need to know the number of groups beforehand! We don’t always know that → metagenomes, genomic repeats, insertions...

→ **Hairsplitter** splits reads in an agnostic number of groups

### Algorithm

1. **Select ‘divergent’ positions**
   - Most frequent base
   - Diverging base

2. Detect SNPs patterns that occur more often than expected from random sequencing errors

3. **3 groups of reads emerge from the patterns**

### Results

- **Hairsplitter** splits a contig into an agnostic number of groups
- **Hairsplitter** can safely be applied to all contigs of the graph to split the contigs that need to be split
- **Hairsplitter** could be used to improve the contiguity of assemblies by overcoming almost-repeated regions

### Conclusion & Perspectives

- **Hairsplitter** splits a contig into an agnostic number of groups
- **Hairsplitter** can safely be applied to all contigs of the graph to split the contigs that need to be split
- **Hairsplitter** could be used to improve the contiguity of assemblies by overcoming almost-repeated regions

### References

