Peer Community In Mathematical & Computational Biology

RESEARCH ARTICLE

Open Access
Open Peer-Review
Open Data
Open Code

HairSplitter: haplotype assembly from long, noisy reads

Roland Faure [1,2], Dominique Lavenier[1] & Jean-François Flot[2,3]

[1] Univ. Rennes, INRIA RBA, CNRS UMR 6074, Rennes, France
[2] Service Evolution Biologique et Ecologie, Université libre de Bruxelles (ULB), Brussels, Belgium
[3] Interuniversity Institute of Bioinformatics in Brussels – (IB)[2], Brussels, Belgium

Cite as: xxx

Correspondence: roland.faure@irisa.fr

Recommender: FirstName FamilyName

Reviewers: FirstName FamilyName and two anonymous reviewers

This version of the article has not yet been peer-reviewed by *Peer Community In Mathematical and Computational Biology* (https://doi.org/xxx/xxx)

Abstract

Motivation: Long-read assemblers face challenges in discerning closely related viral or bacterial strains, often collapsing similar strains in a single sequence. This limitation has been hampering metagenome analysis, where diverse strains may harbor crucial functional distinctions.

Results: We introduce a novel software, HairSplitter, designed to retrieve strains from a strain-oblivious assembly and long reads. The method uses a custom variant calling process to operate with erroneous long reads and introduces a new read clustering algorithm to recover an a priori unknown number of strains. On noisy long reads, HairSplitter can recover more strains while being faster than state-of-the-art tools, both in the viral and the bacterial case.

Availability: HairSplitter is freely available on GitHub at github.com/RolandFaure/HairSplitter.

Contact: roland.faure@irisa.fr

Keywords: Metagenomes; Metaviromes; Haplotyping; Genome assembly; Strain separation

Introduction

Microbiomes play a crucial roles in many ecosystems, such as soils or human guts, in turn impacting human health (Conlon and Bird, 2014) and soil fertility (Coban et al., 2022). Microbiomes typically contain sets of organisms with highly similar genomes, the sequences of which are called haplotypes (short for "haploid genotypes" (Ceppellini et al., 1967)). Distinguishing these lineages is an important challenge, as small genomic differences between haplotypes can lead to significant phenotypic changes. For instance, some strains of *Escherichia coli* can be pathogenic or commensal while having an Average Nucleotide Identity (ANI) (Konstantinidis and Tiedje, 2005) of more than 98.5% (Frank et al., 2011). A few mutations also became famous for altering significantly the infectiousness of some coronaviruses lineages (Magazine et al., 2022).

De novo sequencing and assembling is a central method to characterize microbial communities. Unlike previous methods, it allows to analyse the composition of a metagenome without culturing the strains, enabling a wide range of analyses (Ward, 2006). While existing genome assemblers proficiently reconstruct genomes of abundant species, they struggle to distinguish viral or bacterial haplotypes. The main difficulty for assemblers lies in the unknown number of haplotypes in a sample and their uneven coverage (Ghurye et al., 2016).

Many tools have been developed to overcome this problem in the context of short-read assemblies, such as OPERA-MS (Bertrand et al., 2019), Constrains (C Luo et al., 2015), STRONG (Quince et al., 2020), StrainXpress (Kang et al., 2022) and VStrains (R Luo and Lin, 2023). However, these methods are not designed for long-read sequencing and do not exploit the long-range information contained in long reads.

Long reads with low error rate, such as PacBio HiFi reads, may in the long term be a solution to finely distinguish strains. Specialized software such as hifiasm (Cheng et al., 2021), metamDBG (Benoit et al., 2024) and stRainy (Kazantseva et al., 2023) have proved successful at distinguishing strains. However, HiFi remains expensive and comes with serious limitations regarding the quantity of DNA needed for sequencing.

Several methods have been implemented to deal with haplotype separation for long reads with high error rates. While the viral and bacterial haplotype assembly problems are identical in their formulation, the characteristics of the input data vary significantly: the genomes are generally much shorter and much more deeply sequenced in the viral case. This has led to the emergence of software specialized in either of the two problems. In the context of bacterial strain separation, Vicedomini et al., 2021 showed that mainstream assemblers such as metaFlye (Kolmogorov et al., 2020) and Canu (Koren et al., 2017) failed to distinguish close bacterial haplotypes and proposed a new tool, called Strainberry, to reconstruct strains. In the context of viral strain separation, Strainline (X Luo et al., 2022) and HaploDMF (Cai et al., 2022) were presented to tackle specifically the viral haplotype reconstruction problem and need very high depth of sequencing to work. The method iGDA (Z Feng et al., 2021) was proposed as a general approach to phase minor variants while handling high error rates and could theoretically assemble both bacterial and viral haplotypes. The main shortcomings of all of these methods is that they struggle to recover haplotypes of low abundance. Additionally, most of these tools are very computationally intensive.

We present HairSplitter, an efficient pipeline for separating haplotypes in the viral and bacterial context using potentially error-prone long reads. HairSplitter first calls variants using a custom process to distinguish actual variants from alignment or sequencing artefacts, clusters the reads into an unspecified number of haplotypes, creates the new separated contigs and finally untangles the assembly graph. HairSplitter can be used for either metaviromes or bacterial metagenomes.

Methods

Overview of the pipeline

The HairSplitter pipeline is depicted on Figure 1 and comprises five steps: 1) correcting the assembly, 2) calling variants on each contig, 3) separating the reads by haplotype on each contig, 4) reassembling the strain-specific contigs and 5) scaffolding.

Assembly correction

Initially, the reads are aligned to the assembly using minigraph (Li et al., 2020, and the assembly is subsequently examined for breakpoints. Breakpoints represent locations in a contig where a significant number of reads stop aligning, typically signalling that all the collapsed strains in the contig do not contain the entire sequence of the contig. These contigs are potential sources of misassemblies, thus HairSplitter breaks the contigs at these breakpoints. Additionally, links are added in the graph between ends of contigs when there is sufficient read support. The process is illustrated in Figure 1a.

The refined assembly resulting from this process is used throughout the subsequent stages of the pipeline.

Mathematical model behind variant calling

To sort reads into haplotypes, the intuitive method of clustering reads based on the similarity of their full sequence proves ineffective due to the dominance of sequencing and alignment errors, obscuring strain differences. HairSplitter first identifies variants positions, pinpointing loci where strains exhibit actual differences. The reads are then separated based only on these loci. We did not find any variant caller suitable for our specific challenge - calling variants with noisy long reads in a metagenomic context including potentially low-abundance strains while maintaining high computational efficiency. Thus, we devised our own variant calling procedure, based on an idea already explored in (Z Feng et al., 2021).

The naivest procedure to identify polymorphic loci consists in going through the pileup of the reads on the assembly and identifying loci where at least a proportion p of reads have an alternative allele. However, this approach falls short when using error-prone reads. For instance, in the case of a strain representing only 1% of the total of the reads, p needs to be less than 0.01 to detect variant positions corresponding to this strain, resulting in the selection of many artefactual positions if the reads have an error rate $> 1\%$.

The key lies in taking several loci into account simultaneously. HairSplitter leverages the assumption that alignment artifacts occur randomly in the pileup, whereas genomic variant are expected to be correlated along the alignment. Consequently, pileups at polymorphic loci exhibit strong correlation, contrary to pileups at non-polymorphic loci. This allows HairSplitter to detect even rare strains, as illustrated below.

Consider a complete pileup of n reads over m positions, which we will model as a matrix of letters. Let us assume that errors occur independently on all reads and at all positions with a probability $\leq e$ and that all errors on a given column are identical (worst-case scenario). We aim to estimate the probability that there exist a reads that share errors at b different loci. In other words, the probability that there exist a submatrix of size $a * b$ containing only errors in the pileup, defined by selecting a rows (reads) and b columns (loci).

There exist $\binom{n}{a}\binom{m}{b}$ submatrices of size $a * b$. Each of these submatrix has probability lower than e^{ab} to contain only errors. Therefore, given that the expectation is linear (DeGroot and Schervish, 2002), the expectation E of the number of submatrices of size $a * b$ containing only errors in the pileup is lower than $\binom{n}{a}\binom{m}{b} * e^{ab}$. Now, to obtain the probability that there exist no submatrix of size $a * b$ containing only errors, we can use Markov's inequality, according to which the probability that a positive random variable be higher than 1 is always smaller than the expectation of this variable (DeGroot and Schervish, 2002). Here, it tells us that the

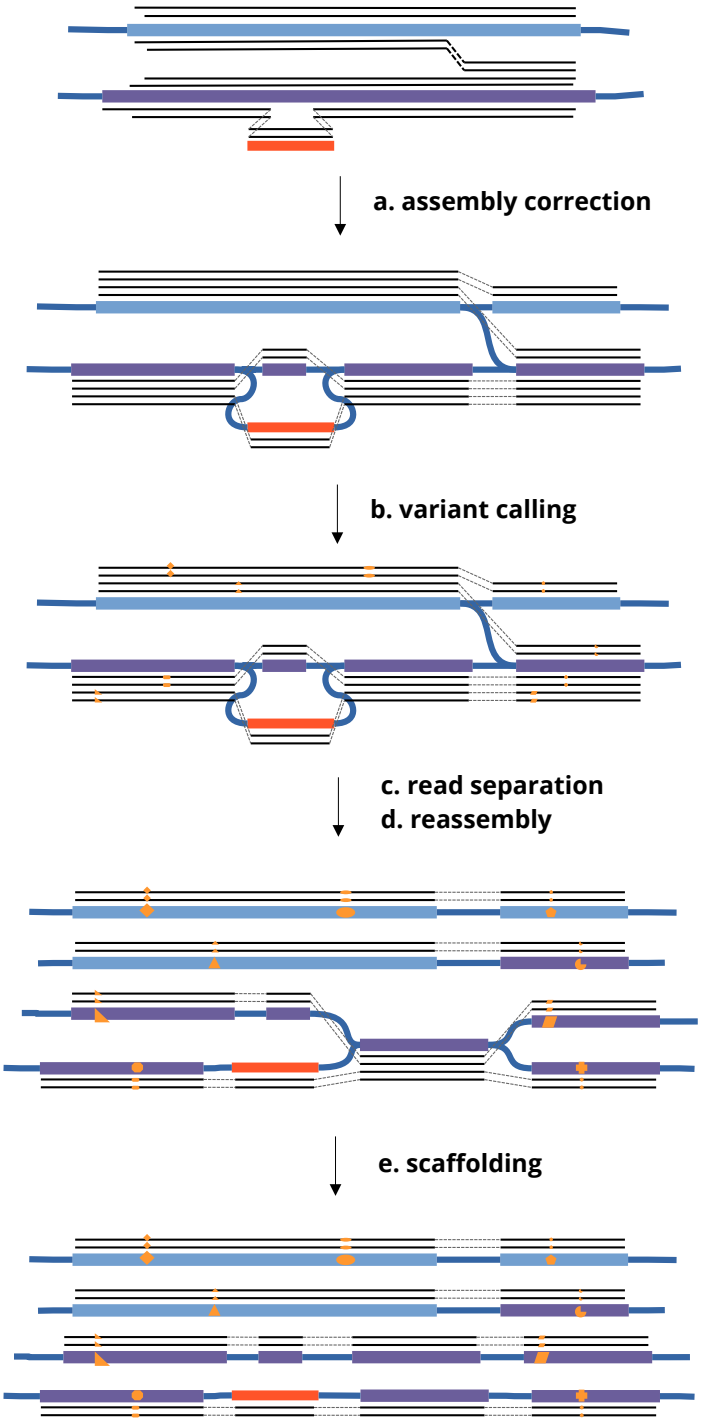

Figure 1. Illustration of the five steps of the HairSplitter pipeline. Colored rectangles represent contigs, thick blue lines are links in the assembly graph and black lines represent the reads aligned on the assembly. Orange shapes on reads and contigs indicate variant positions compared to the original sequence.

```
ref AACAAGATAGACCAGATAGACACAGATTGGCGTTTAGGAACAGATGACAGATACGCA

r1  AACAAGATAGA-CAGATAGACACAGATTGGCGTTTAGGAACAGATGACAGATA-GCA
r2  AACAAGATAGAC-AGATAGCACAGGATTGGCGTTTAGGAACAGATGATAGATAC--A
r3  AACAAGATAGA-CAGATAGACACAGATTGGCGTTTAGTAACAGATGACAGATAGCCA
r4  AACCAGATAGAC-AGATAGACACATATTGGCGTTTAGGAACATTTGACAGATA-GCA
r5  AACCAGATAGA-CAGATAGGCACATATTGGCGTTTAGGAACAGTTGACAGA--CGCA
r6  AACCAGATAGAC-AGATAGACACATATTGGCGTTTAGGATCAGTTGACAGATA-GCA
```

Figure 2. In this pileup of reads, does the submatrix of variants highlighted in red vouch for the presence of two strains? The probability that there exist 3 reads having alternative allele at 3 loci if we estimate $e = 0.1$ is less than 0.02: the variants are thus likely not independent and probably underline the presence of at least two different strains.

probability that there exist a submatrix containing only errors is smaller than E. In other terms, the probability that there exist somewhere in the pileup a reads sharing errors at b different loci is lower than $\binom{n}{a}\binom{m}{b} * e^{ab}$.

Now, let us consider a pileup with $n = 1000$ reads across $m = 5000$ positions and $e = 0.1$. The probability that there exist $a = 10$ reads sharing errors at $b = 10$ different loci is lower than $\binom{n}{a}\binom{m}{b} * e^{ab} = 9.10^{-44}$. Therefore, if the error rate is of 10% or less and the pileup indicates 10 reads (1% coverage) sharing an alternative allele at 10 loci (divergence of 0.2%), we can confidently assume that these are not errors, suggesting these reads originate from the same strain, and the loci are polymorphic sites.

Despite its simplified nature, this model underscores the statistical power gained by examining multiple loci simultaneously, enabling the detection of low-abundance, highly similar strains even in the presence of very noisy long reads. The idea behind the model is illustrated in Figure 2.

Variant calling

The approach to identifying polymorphic loci capitalizes on the statistical power underlined above. Specifically, HairSplitter aims to identify clusters of positions featuring alternative alleles on the same reads.

To generate the pileup, all reads are aligned to the assembly using minimap2 (Li, 2018) using default settings. HairSplitter then traverses the pileup, determining, for each position, the majority allele and the main alternative allele. Only positions with a minimum of five reads carrying alternative alleles are considered potential polymorphic sites to ensure statistical robustness (cf. model above). HairSplitter compares each new position to previously observed positions. If the set of reads with alternative alleles at this position and at a previously encountered position share more than 90% reads, the new position is clustered with the old one.

After all positions have been considered, clusters with more than five positions are deemed robust, and the corresponding positions are outputted as polymorphic sites. Once again, the threshold is imposed to avoid the inadvertent selection of artifact-prone positions.

Read binning

The contig is divided into windows with a default size of w (2000 bases by default). Reads are binned by haplotypes sequentially on the windows of a contig. Only reads spanning the entirety of the window are considered for binning. To cluster reads, HairSplitter operates on the premise that reads originating from the same haplotype should be identical at all polymorphic loci. Nevertheless, inherent sequencing and variant-calling errors might introduce unintended discrepancies among reads from a single haplotype. To address this, HairSplitter adopts a three-step strategy.

dataset	species	# strains	strain coverages	ANI divergence	sequencing technology
HBV-2	hepatitis B	2	4000x, 9900x	10%	Nanopore R.9.4.1
Norovirus-7	Norovirus	7	50, 350, 450, 700, 900, 1150, 1400x	1-3.9 %	Nanopore R.9.4.1
V. fluvialis	*Vagococcus fluvialis*	5	90x, 136x, 172x, 182x, 206x	0.01-1.51%	Nanopore R9.4.1
Zymo-GMS Q9	*Escherichia coli*	5	90x, 90x, 90x, 90x, 90x	0.37-1.51%	Nanopore R9.4.1
Zymo-GMS Q20	*Escherichia coli*	5	25x, 25x, 25x, 25x, 25x	0.37-1.51%	Nanopore R10.4.1
Zymo-GMS HiFi	*Escherichia coli*	5	41x, 41x, 41x, 41x, 41x	0.37-1.51%	PacBio HiFi

Table 1. Characteristics of the different datasets used for benchmarking on real data.

Step one is to correct errors at polymorphic loci. HairSplitter corrects the errors at polymorphic loci by performing a k-nearest-neighbour imputation (Fix and Hodges, 1989), with $k = 5$. The distance between two reads is defined as the number of different alleles at polymorphic positions. Each base of the pileup is considered and changed to the most frequent base among the k nearest neighbours on all reads and all positions until convergence.

Step two is to form clusters of reads, clustering reads together if and only if they exhibit no differences at any polymorphic loci.

In the third step, a last check is run to rescue small clusters that can arise from errors in Step 1. HairSplitter constructs a graph linking each read to its k closest neighbours, including links between all pairs of reads differing on one position or less. The graph is then clustered using the Chinese Whispers algorithm (Biemann, 2006), initialising the clustering with the clusters obtained in the second step. The Chinese Whispers algorithm always converge toward a stable solution, i.e. a clustering where all reads are in the same group as at least half of their neighbors. There exist many stable clusterings but the algorithm is likely to converge to a solution close to the initialization: the clusters obtained in the second step are unlikely to be significantly altered.

Reassembly

Across all windows on every contig, the original sequence undergoes repolishing using the haplotype-specific groups of reads previously identified. The polishing can be executed with either Racon (Fang and Wang, 2022) or Medaka (*Medaka* 2018), with the latter being more precise but considerably slower in our experience. By default, HairSplitter uses Medaka exclusively for short genomes ($\leq$ 1 Mb).

The resulting assembly comprises contigs of length w that can easily be stitched into longer contigs. For this purpose, a straightforward algorithm is employed, derived from Faure et al.(Faure et al., 2021) and depicted in Figure 1e. Let us call a contig exhibiting multiple outgoing links with other contigs at one end a “knot”. Knots generally represent collapsed contigs. All reads are initially aligned on the assembly graph. Subsequently, knots are iteratively assessed. If more than three reads traverse a neighbor of the knot (called A), then traverse the knot, and traverse another neighbor at the opposite end of the knot (called B), the knot is duplicated to create a new contig which will have as unique neighbors A and B. The links from A and B to the original knot are deleted, preserving only the links to the copy of the contig. This process is repeated until no further knots can be duplicated.

Results

Datasets

The datasets used in this article are described in Table 1. The accession numbers of the data on public repositories can be found in section* “Reproducibility and data availablility".

Bacterial datasets

We used the Zymobiotics Gut Microbiome Standard (abbreviated to Zymo-GMS) and a *Vagococcus fluvialis* dataset (Rodriguez Jimenez et al., 2022) to compare the performance of different algorithms designed to separate bacterial haplotypes in a metagenomic context. Zymo-GMS is a mixture of bacteria, archaea and yeast, 21 different strains in total, dosed to mimic the composition of the human gut microbiome. These 21 strains include five *Escherichia coli* strains, which we used to evaluate the strain-separation ability of various programs. Three Zymo-GMS sequencing were used, respectively from a Nanopore R9.4.1 run, a Nanopore 10.4.1 run and a PacBio HiFi run. The *Vagococcus fluvialis* dataset consists of a mix of five *Vagococcus fluvialis* strains that were sequenced together using barcoded reads, each barcode corresponding to a strain. We did not use the barcode information for the assemblies, reserving them for validation. Among the five strains, three had an ANI over 99.99%. metaFlye is used to assemble the reads, as it yielded better assemblies compared to Canu according to Vicedomini et al. (Vicedomini et al., 2021).

In addition, we simulated datasets to assess the impact of the number of strains, coverage and divergence on the assemblies. These experiments were directly inspired by the protocol of Vicedomini et al. (Vicedomini et al., 2021). The genomes of ten strains of *Escherichia coli* were downloaded from the SRA, namely 12009 (GCA_000010745.1), IAI1 (GCA_000026265.1), F11 (GCA_018734065.1), S88 (GCA_000026285.2), Sakai (GCA_003028755.1), SE15 (GCA_000010485.1), *Shigella flexneri* (GCF_000006925.2), UMN026 (GCA_000026325.2), HS (GCA_ 000017765.1), and K12 (GCF_009832885.1). These strains were chosen to be representative of the diversity of *E. coli.* We simulated Nanopore sequencing using Badread (R Wick, 2019) with the setting "Nanopore2023" to simulate 50x of R10.4.1 reads. Between 2 and 10 strains were mixed to assess how many strains the software could separate. From the 10-strain mix, the 12009 strain was downsampled to 30x, 20x, 10x and 5x to assess the impact of the coverage on strain separation. Finally, to assess the impact of the divergence of sequences on strain separation, 50x of reads were simulated for strain K12 and for strains of decreasing divergence with K12; assemblies of K12 with each of these strain was evaluated for separation.

Viral datasets

Two datasets were used to benchmark the performance of the programs tested at separating viral haplotypes, a 2-strain hepatitis B Virus (HBV) mix from (McNaughton et al., 2019) and an in-silico mix of the sequencing of seven strains of Norovirus from Cai et al. (Flint et al., 2021). These datasets were directly taken from the paper of HaploDMF (Cai et al., 2022). The reference genomes to run reference-based tools were taken as the reference genome in the GenBank database, GCF_000861825.2 for HBV and MW661279.1 for Norovirus.

Performance evaluation

We used MetaQUAST (Mikheenko et al., 2015) to measure assembly features such as assembly length, NG50, misassemblies, mismatches, indels and completeness. MetaQUAST was run with the –unique-mapping and –reuse-combined-alignments options to prevent a sequence, whether a contig or part of it, from being mapped to multiple distinct reference locations.

To assess if strains are well represented, the most important metric is the completeness of the resulting assembly. We chose to assess MetaQUAST completeness but also 27-mer completeness. MetaQUAST completeness measures the percentage of the solution on which the assembly aligns, while 27-mer completeness measures the percentage of the 27-mers of the solution that are effectively found in the assembly. Collapsed homozygous contigs typically impact negatively MetaQUAST completeness but not 27-mer completeness.

Evaluated software

In addition of HairSplitter, we chose to evaluate the software stRainy (Kazantseva et al., 2023) and Strainberry (Vicedomini et al., 2021), which have been introduced specifically as bacterial strain separation methods,

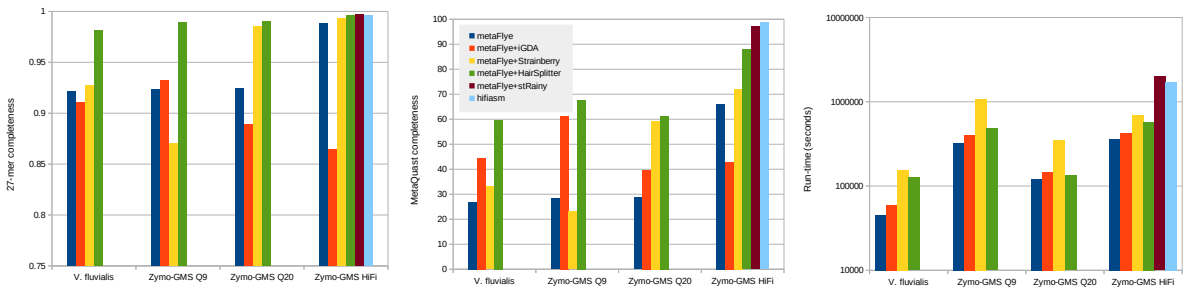

Figure 3. 27-mer completeness, MetaQUAST completeness and run-time of different software on the three Zymo-GMS dataset. The run-times are the run-times of the full assembly pipeline (assembly+strain separation) and are represented in log scale.

hifiasm-meta (X Feng et al., 2022), which is the most popular assembler for direct HiFi assembly, Strainline (X Luo et al., 2022) and HaploDMF (Cai et al., 2022), which have been introduced as viral strain separation methods and finally iGDA (Z Feng et al., 2021), which can perform both.

We have tried using all these software on all datasets. Strainline and HaploDMF failed to run in reasonable time on non-viral datasets and were automatically killed after 15 days of processing. Strainline failed to perform strain separation on the HBV-2 dataset within its allowed RAM limit of 50G, probably because of the high coverage. We tried downsampling the dataset but the problem remained.

The reference-based virus phasing tools were run with the same reference genome as in (Cai et al., 2022), MT622522.1 for hepatitis B and MW661279.1 for Norovirus.

Benchmarking evaluation

Bacterial haplotypes

The benchmark results on the Zymo-GMS and *V. fluvialis* datasets are illustrated in Figure 3 and detailed in Supplementary Table 2. HairSplitter performed better separation of the conspecific strains compared to the original metaFlye assemblies, delivering more comprehensive and accurate assemblies than Strainberry and iGDA. Particularly with Nanopore data, HairSplitter produced the most complete assemblies. On HiFi reads, all assemblies depicted a high duplication ratio but no evidence of erroneous contigs. This could be caused by the presence of small intra-specific variation unrepresented in the reference and would merit further investigation. The direct assembly of reads with hifiasm seems to yield the best results in terms of completeness and contiguity. In several assemblies, 27-mer completeness remains high while MetaQUAST completeness is notably lower. This discrepancy arises when repeated genomic regions are not duplicated to their correct multiplicity. Typically, the three almost identical *V. fluvialis* strains were assembled as one.

The completeness of assemblies in the simulated benchmark is presented in Figure 4, with a detailed evaluation in Supplementary Table 3. The evaluation of iGDA is not depicted because iGDA inexplicably decreased the completeness of the original metaFlye assemblies. Simulations indicated that HairSplitter significantly outperformed Strainberry, particularly in scenarios involving a high number of strains in the metagenome. The relatively high completeness of the 8-strains Strainberry assembly can be attributed to its high duplication ratio. The completeness of HairSplitter assemblies decreased with the depth of coverage and similarity of the strains, especially below 20x coverage and 1% divergence. Interestingly, the decline in MetaQUAST completeness with coverage and divergence was more pronounced than the decline in 27-mer completeness, highlighting HairSplitter's effectiveness in separating divergent regions and its difficulties in duplicating homozygous regions. This corresponds to the results observed in the Zymo-GMS datasets, where many pairwise divergences of strains were $< 1\%$.

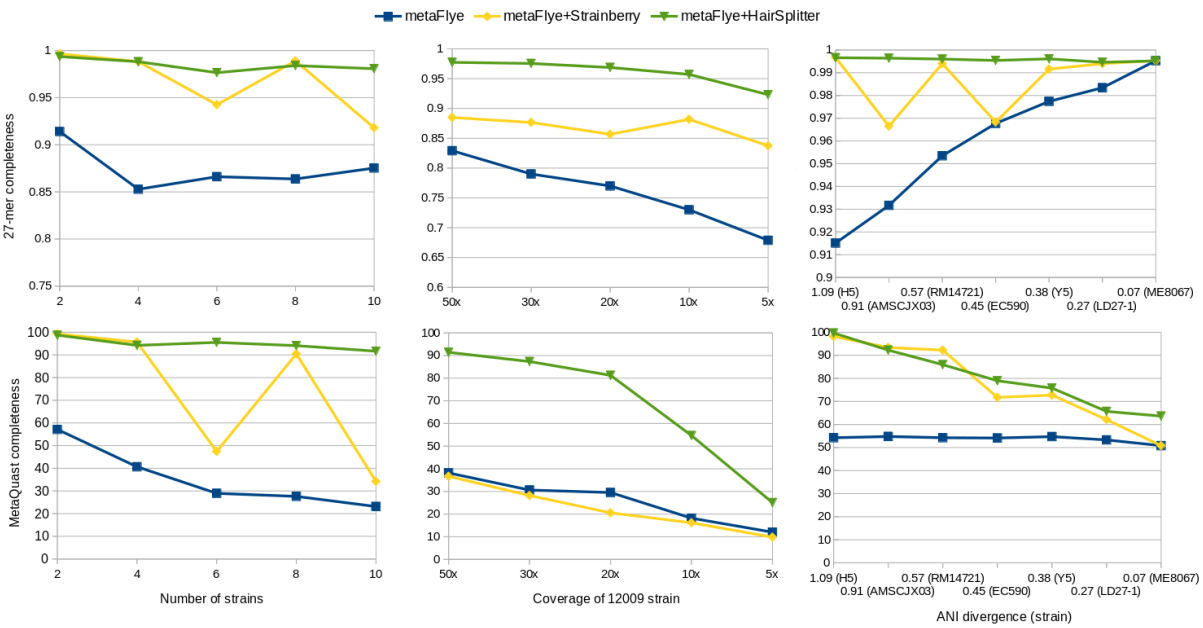

Figure 4. MetaQUAST completeness of assemblies of simulated metagenomes of *E. coli*. On the left, mix of 2 to 10 strains sequenced with 50x coverage were assembled. In the middle, strain 12009 was downsampled in the 10-strains metagenome and completeness of the 12009 strain is measured. On the right, reads of strains of decreasing divergence were mixed with K-12 reads and assembled.

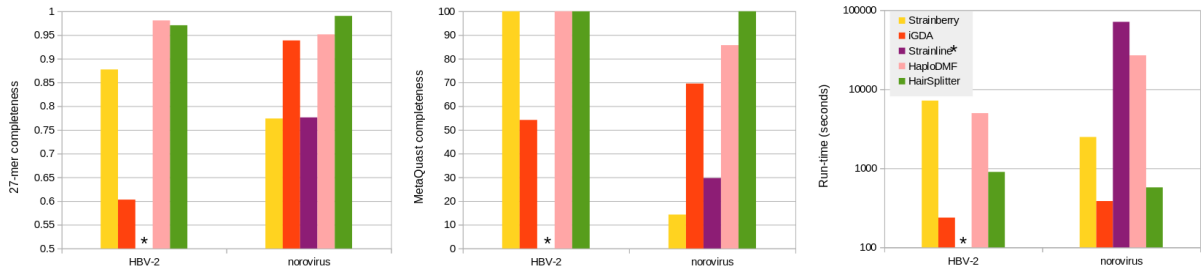

Figure 5. 27-mer completeness, MetaQUAST completeness and run-time of different software on the two viral datasets. Note that the run-time is shown in log scale. The Strainline assembly of HBV-2 is not shown because Strainline could not finish on this dataset.

Viral haplotypes

The completeness results of the benchmark on the viral datasets are depicted Figure 5 and more complete evaluation of assemblies are available in Supplementary Table 5.

HaploDMF and HairSplitter managed to separate completely the HBV strains according to MetaQUAST. iGDA failed to recover the strains, while Strainberry outputted four different haplotypes instead of two (see supplementary Table 5). We checked that HaploDMF and HairSplitter separated the reads adequately, thus the slight differences in 27-mers completeness stem from polishing errors.

HairSplitter stood out as the sole software capable of successfully recovering all seven strains in the Norovirus mix, even capturing the least abundant strain comprising only 1% of the mix. To assess the sensitivity limits of HairSplitter in the viral context, we conducted two additional experiments within the Norovirus mix. In the first experiment, we decreased the relative abundance of the rarest strain to 0.1%, while maintaining 50x coverage by uniformly increasing the coverage of the other strains. Remarkably, HairSplitter still achieved complete recovery (99.99% MetaQUAST completeness) of the rarest strain. The limited amount of data prevented us to further reduce the strain's relative abundance. In the second experiment, we uniformly diminished the coverage of all strains. The rarest strain was entirely recovered (99.99% MetaQUAST completeness) when covered at $\geq$40x, only the most divergent part of the virus was recovered (26.4% MetaQUAST completeness) at

coverage 20x and 30x, and the strain was not recovered at all at 10x coverage. The primary determinant of HairSplitter's sensitivity thus seems to be absolute coverage rather than the strain's relative coverage.

Discussion

In this manuscript, we introduce HairSplitter, a pipeline to assemble haplotypes separately using an input assembly and long reads. The pipeline includes two main novelties, an assembly correction program and a read separation procedure. HairSplitter proved useful when dealing with noisy data ($\geq$ 1% error rate), whereas specialised software such as hifiasm seemed to be better indicated to assemble HiFi reads. We show that HairSplitter can effectively separate several similar strains in both bacterial and viral contexts. Compared to the state of the art, HairSplitter can deal with a high number of strains and with low relative abundances, while maintaining a low computational cost.

HairSplitter faces a significant limitation when strains are highly similar, resulting in two distinct challenges. Firstly, the tool requires reads to span at least five polymorphic loci for effective separation into haplotype groups, posing difficulties in highly similar regions. Secondly, even with optimal read separation, homozygous regions present a challenge, as reads cannot be binned into haplotype groups within these regions. Consequently, full strain recovery necessitates the duplication of homozygous regions to their correct multiplicity. Although an immediate solution to the first problem is not apparent, the results obtained using HiFi data suggest progress could be made concerning homozygous region duplication. Indeed, stRainy achieved a 27-mer completeness comparable to HairSplitter, suggesting that the separation of reads was not significantly finer. However, stRainy outperformed HairSplitter in MetaQUAST completeness, indicating more effective duplication of homozygous regions in the assembly process.

A direction for future work would also be to generalize the assembly correction module. The idea of the module is to make sure all reads align end-to-end onto the assembly graph. We believe such a module could be useful to improve many assemblies. However, the version implemented for now in HairSplitter is very basic and does not perform well in repeated, complicated regions of the graph. A more sophisticated module could involve local reassembly and iterative graph correction.

Since HairSplitter is already successful at separating both bacterial and viral haplotypes, we expect to be able to extend this work naturally towards the phasing of polyploid organisms, including highly heterozygous non-model organisms, which remains an open problem (Guiglielmoni et al., 2021). For this particular case, some extra information could be leveraged to improve the HairSplitter pipeline, such as the fact that all haplotypes are expected to be equally abundant and that the total number of haplotype is usually known.

Reproducibility and data availablility

The HairSplitter code can be found on github at https://github.com/rolandfaure/hairsplitter.

The experiments were run with Flye 2.9.2-b1786, hifiasm HairSplitter v1.6.10, HaploDMF commit a07d082c3, Strainline commit 8d26341, iGDA commit 54ecec9, Strainberry v1.1, stRainy commit 34573cd, hifiasm-meta v0.3-r063.2, minimap2 v2.26-r1175 and Quast v5.2.0.

HBV sequencing reads can be found under accession number ERR3253560 in SRA. The seven Norovirus sets of reads can be found under accession numbers SRR13951181, SRR13951181, SRR13951186, SRR13951185, SRR13951184, SRR13951165 and SRR13951160. The *Vagococcus fluvialis* data are accessible under project PRJNA755170 in SRA. The Zymo-GMS sequencing data can be found under accession numbers SRR17913200, SRR17913199 and SRR13128013.

All the assemblies, simulated data and command lines used are available on Zenodo, DOI 10.5281/zenodo.10495033, https://zenodo.org/records/10495033.

Acknowledgments

We thank Ulysse Faure for his mathematical help. Alexandros Vasilikopoulos, Andrew Woodruff and Alessandro Derzelle tested HairSplitter and kindly helped debugging.

We acknowledge the GenOuest bioinformatics core facility (https://www.genouest.org) for providing the computing infrastructure. The programs Tablet (Milne et al., 2013) and Bandage (RR Wick et al., 2015) were used to visualize data while developing HairSplitter.

For the purpose of Open Access, a CC-BY public copyright licence has been applied by the authors to the present document and will be applied to all subsequent versions up to the Author Accepted Manuscript arising from this submission

Fundings

This work was funded by a Ph.D. AMX grant.

Conflict of interest disclosure

The authors declare that they comply with the PCI rule of having no financial conflicts of interest in relation to the content of the article. The authors declare the following non-financial conflict of interest: Jean-François Flot is a recommender of PCI Genomics.

References

Benoit G, S Raguideau, R James, A Phillippy, R Chikhi, and C Quince (Jan. 2024). High-quality metagenome assembly from long accurate reads with metaMDBG. *Nature Biotechnology*, 1–6. https://doi.org/10.1038/s41587-023-01983-6.

Bertrand D, J Shaw, M Kalathiyappan, AHQ Ng, MS Kumar, C Li, M Dvornicic, JP Soldo, JY Koh, C Tong, OT Ng, T Barkham, B Young, K Marimuthu, KR Chng, M Sikic, and N Nagarajan (Aug. 2019). Hybrid metagenomic assembly enables high-resolution analysis of resistance determinants and mobile elements in human microbiomes. en. *Nature Biotechnology* 37, 937–944. ISSN: 1087-0156, 1546-1696. https://doi.org/10.1038/s41587-019-0191-2.

Biemann C (July 2006). Chinese whispers: An efficient graph clustering algorithm and its application to natural language processing problems. *Proceedings of TextGraphs*, 73–80.

Cai D, J Shang, and Y Sun (Oct. 2022). HaploDMF: viral Haplotype reconstruction from long reads via Deep Matrix Factorization. *Bioinformatics* 38. https://doi.org/10.1093/bioinformatics/btac708.

Ceppellini R, E Curtoni, P Mattiuz, V Miggiano, G Scudeller, and A Serra (1967). Genetics of leukocyte antigens: a family study of segregation and linkage. In: *Report of Histocompatibility testing 1967*. Ed. by Curtoni E.S. Mattiuz P.L. TR.

Cheng H, G Concepcion, X Feng, H Zhang, and H Li (Feb. 2021). Haplotype-resolved de novo assembly using phased assembly graphs with hifiasm. *Nature Methods* 18, 1–6. https://doi.org/10.1038/s41592-020-01056-5.

Coban O, G Deyn, and M Ploeg (Mar. 2022). Soil microbiota as game-changers in restoration of degraded lands. *Science* 375, abe0725. https://doi.org/10.1126/science.abe0725.

Conlon M and A Bird (Dec. 2014). The Impact of Diet and Lifestyle on Gut Microbiota and Human Health. *Nutrients* 7, 17–44. https://doi.org/10.3390/nu7010017.

DeGroot M and M Schervish (Jan. 2002). *Probability and Statistics*. Pearson. ISBN: ISBN 978-0-321-50046-5.

Fang L and K Wang (June 2022). Polishing high-quality genome assemblies. en. *Nature Methods* 19, 649–650. ISSN: 1548-7091, 1548-7105. https://doi.org/10.1038/s41592-022-01515-1.

Faure R, N Guiglielmoni, and JF Flot (Feb. 2021). GraphUnzip: unzipping assembly graphs with long reads and Hi-C. *bioRxiv*. https://doi.org/10.1101/2021.01.29.428779.

Feng X, H Cheng, D Portik, and H Li (June 2022). Metagenome assembly of high-fidelity long reads with hifiasm-meta. *Nature Methods* 19, 1–4. https://doi.org/10.1038/s41592-022-01478-3.

Feng Z, J Clemente, B Wong, and E Schadt (May 2021). Detecting and phasing minor single-nucleotide variants from long-read sequencing data. *Nature Communications* 12, 3032. https://doi.org/10.1038/s41467-021-23289-4.

Fix E and JL Hodges (1989). Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties. *International Statistical Review* 57, 238–247. ISSN: 03067734, 17515823.

Flint A, S Reaume, J Harlow, E Hoover, K Weedmark, and N Nasheri (Sept. 2021). Genomic Analysis of Human Noroviruses Using Combined Illumina-Nanopore Data. *Virus Evolution* 7. https://doi.org/10.1093/ve/veab079.

Frank C, D Werber, JP Cramer, M Askar, M Faber, M an der Heiden, H Bernard, A Fruth, R Prager, A Spode, M Wadl, A Zoufaly, S Jordan, MJ Kemper, P Follin, L Müller, LA King, B Rosner, U Buchholz, K Stark, and G Krause (2011). Epidemic Profile of Shiga-Toxin–Producing *Escherichia coli* O104:H4 Outbreak in Germany. *New England Journal of Medicine* 365, 1771–1780. https://doi.org/10.1056/NEJMoa1106483.

Ghurye J, V Cepeda-Espinoza, and M Pop (Sept. 2016). Metagenomic Assembly: Overview, Challenges and Applications. *The Yale Journal of Biology and Medicine* 89, 353–362.

Guiglielmoni N, A Houtain, A Derzelle, K Doninck, and JF Flot (June 2021). Overcoming uncollapsed haplotypes in long-read assemblies of non-model organisms. *BMC Bioinformatics* 22. https://doi.org/10.1186/s12859-021-04118-3.

Kang X, X Luo, and A Schönhuth (Sept. 2022). StrainXpress: strain aware metagenome assembly from short reads. en. *Nucleic Acids Research* 50, e101–e101. ISSN: 0305-1048, 1362-4962. https://doi.org/10.1093/nar/gkac543.

Kazantseva E, A Donmez, M Pop, and M Kolmogorov (Feb. 2023). *stRainy: assembly-based metagenomic strain phasing using long reads*. en. preprint. Bioinformatics. https://doi.org/10.1101/2023.01.31.526521.

Kolmogorov M, DM Bickhart, B Behsaz, A Gurevich, M Rayko, SB Shin, K Kuhn, J Yuan, E Polevikov, TPL Smith, and PA Pevzner (Nov. 2020). metaFlye: scalable long-read metagenome assembly using repeat graphs. en. *Nature Methods* 17, 1103–1110. ISSN: 1548-7091, 1548-7105. https://doi.org/10.1038/s41592-020-00971-x.

Konstantinidis K and J Tiedje (Mar. 2005). Genomic insights that advance the species definition for prokaryotes. *Proceedings of the National Academy of Sciences of the United States of America* 102, 2567–72. https://doi.org/10.1073/pnas.0409727102.

Koren S, BP Walenz, K Berlin, JR Miller, NH Bergman, and AM Phillippy (May 2017). Canu: scalable and accurate long-read assembly via adaptive *k* -mer weighting and repeat separation. en. *Genome Research* 27, 722–736. ISSN: 1088-9051, 1549-5469. https://doi.org/10.1101/gr.215087.116.

Li H (Sept. 2018). Minimap2: pairwise alignment for nucleotide sequences. en. *Bioinformatics* 34. Ed. by Birol I, 3094–3100. ISSN: 1367-4803, 1367-4811. https://doi.org/10.1093/bioinformatics/bty191.

Li H, X Feng, and C Chu (Oct. 2020). The design and construction of reference pangenome graphs with minigraph. *Genome Biology* 21, 265. https://doi.org/10.1186/s13059-020-02168-z.

Luo C, R Knight, H Siljander, M Knip, R Xavier, and D Gevers (Sept. 2015). ConStrains identifies microbial strains in metagenomic datasets. *Nature biotechnology* 33. https://doi.org/10.1038/nbt.3319.

Luo R and Y Lin (2023). VStrains: De Novo Reconstruction of Viral Strains via Iterative Path Extraction from Assembly Graphs. In: *Research in Computational Molecular Biology*. Ed. by Tang H. Cham: Springer Nature Switzerland, pp. 3–20. ISBN: 978-3-031-29119-7.

Luo X, X Kang, and A Schönhuth (Jan. 2022). Strainline: full-length de novo viral haplotype reconstruction from noisy long reads. *Genome Biology* 23. https://doi.org/10.1186/s13059-021-02587-6.

Magazine N, T Zhang, Y Wu, M McGee, G Veggiani, and W Huang (Mar. 2022). Mutations and Evolution of the SARS-CoV-2 Spike Protein. *Viruses* 14, 640. https://doi.org/10.3390/v14030640.

McNaughton A, H Roberts, D Bonsall, Md Cesare, J Mokaya, S Lumley, T Golubchik, P Piazza, J Martin, C Lara, A Brown, M Ansari, R Bowden, E Barnes, and P Matthews (May 2019). Illumina and Nanopore methods for whole genome sequencing of hepatitis B virus (HBV). *Scientific Reports* 9. https://doi.org/10.1038/s41598-019-43524-9.

Medaka (2018). github.com/nanoporetech/medaka.

Mikheenko A, V Saveliev, and A Gurevich (Nov. 2015). MetaQUAST: Evaluation of metagenome assemblies. *Bioinformatics* 32, btv697. https://doi.org/10.1093/bioinformatics/btv697.

Milne I, G Stephen, M Bayer, PJA Cock, L Pritchard, L Cardle, PD Shaw, and D Marshall (Mar. 2013). Using Tablet for visual exploration of second-generation sequencing data. en. *Briefings in Bioinformatics* 14, 193–202. ISSN: 1467-5463, 1477-4054. https://doi.org/10.1093/bib/bbs012.

Quince C, S Nurk, S Raguideau, R James, OS Soyer, JK Summers, A Limasset, AM Eren, R Chikhi, and AE Darling (Sept. 2020). *Metagenomics Strain Resolution on Assembly Graphs*. en. preprint. Bioinformatics. https://doi.org/10.1101/2020.09.06.284828.

Rodriguez Jimenez A, N Guiglielmoni, L Goetghebuer, E Dechamps, I George, and JF Flot (Aug. 2022). Comparative genome analysis of *Vagococcus fluvialis* reveals abundance of mobile genetic elements in sponge-isolated strains. *BMC Genomics* 23. https://doi.org/10.1186/s12864-022-08842-9.

Vicedomini R, C Quince, AE Darling, and R Chikhi (July 2021). Strainberry: automated strain separation in low-complexity metagenomes using long reads. en. *Nature Communications* 12, 4485. ISSN: 2041-1723. https://doi.org/10.1038/s41467-021-24515-9.

Ward N (Apr. 2006). New directions and interactions in metagenomics research. *FEMS microbiology ecology* 55, 331–8. https://doi.org/10.1111/j.1574-6941.2005.00055.x.

Wick R (Apr. 2019). Badread: simulation of error-prone long reads. *Journal of Open Source Software* 4, 1316. ISSN: 2475-9066. https://doi.org/10.21105/joss.01316.

Wick RR, MB Schultz, J Zobel, and KE Holt (Oct. 2015). Bandage: interactive visualization of *de novo* genome assemblies. en. *Bioinformatics* 31, 3350–3352. ISSN: 1367-4811, 1367-4803. https://doi.org/10.1093/bioinformatics/btv383.

Supplementary material

		Genome Fraction (%)	Duplication ratio	NGA50	#misassemblies	# mismatches per 100 kbp	# indels per 100 kbp
Vagococcus fluvialis	metaFlye	26.90	1.097	-	40	340.14	383.57
	metaFlye + iGDA	44.530	1.151	1313	2	115.57	391.70
	metaFlye + Strainberry	33.112	1.119	-	45	77.71	510.25
	metaFlye + HairSplitter	59.668	1.137	31944	73	31944	355.52
Zymo-GMS Q9*	metaFlye	28.365	1.048	-	66	286.86	38.80
	metaFlye + iGDA	61.293	1.489	12450	21	225.67	41.30
	metaFlye + Strainberry	23.166	1.072	-	45	191.51	51.65
	metaFlye + HairSplitter	67.642	1.131	15357	89	93.44	30.94
Zymo-GMS Q20*	metaFlye	28.742	1.051	-	62	300.25	34.41
	metaFlye + iGDA	39.650	1.118	-	8	181.55	28.23
	metaFlye + Strainberry	59.197	1.138	28421	66	193.55	42.42
	metaFlye + HairSplitter	61.291	1.033	5264	12	34.77	11.97
Zymo-GMS HiFi*	metaFlye	66.064	1.076	79832	39	92.55	6.65
	metaFlye + iGDA	42.996	1.515	17104	15	102.70	9.96
	metaFlye + Strainberry	72.016	1.142	53249	46	57.53	6.92
	metaFlye + HairSplitter	87.906	1.582	45942	56	62.30	10.53
	metaFlye + stRainy	97.078	1.737	41195	47	44.15	12.26
	hifiasm	98.732	1.911	288422	82	30.07	4.99

Table 2. metaQuast metrics of the bacterial assemblies obtained from experimental data.

		Genome Fraction (%)	Duplication ratio	NGA50	#misassemblies	# mismatches per 100 kbp	# indels per 100 kbp
# strains							
2	metaFlye	57.137	1.043	61559	22	216.18	216.53
	metaFlye + Strainberry	99.268	1.074	701492	11	24.83	64.73
	metaFlye + HairSplitter	98.696	1.063	194764	7	11.54	47.26
4	metaFlye	40.666	1.071	-	50	562.91	268.83
	metaFlye + Strainberry	95.631	1.148	251144	39	93.07	81.88
	metaFlye + HairSplitter	94.217	1.077	25068	6	14.21	46.39
6	metaFlye	28.949	1.087	-	63	585.20	264.89
	metaFlye + Strainberry	47.430	1.108	8151	90	289.19	92.23
	metaFlye + HairSplitter	95.505	1.099	37869	35	39.62	46.65
8	metaFlye	27.599	1.051	-	76	527.44	277.85
	metaFlye + Strainberry	90.438	1.533	83755	157	179.31	172.25
	metaFlye + HairSplitter	94.079	1.157	32914	71	48.11	53.62
10	metaFlye	23.130	1.036	-	79	469.27	277.71
	metaFlye + Strainberry	34.207	1.095	-	175	363.06	137.01
	metaFlye + HairSplitter	91.626	1.159	27528	78	51.29	47.10
coverage							
30x*	metaFlye	30.618	1.032	-	1	719.80	55.27
	metaFlye + Strainberry	28.188	1.085	-	2	347.21	127.06
	metaFlye + HairSplitter	87.322	1.134	28641	0	60.09	35.23
20x*	metaFlye	29.522	1.018	-	6	859.11	76.03
	metaFlye + Strainberry	20.562	1.037	-	3	274.86	102.01
	metaFlye + HairSplitter	81.232	1.138	28700	1	94.37	37.13
10x*	metaFlye	18.201	1.005	-	1	498.17	85.10
	metaFlye + Strainberry	16.178	1.007	-	1	347.06	181.70
	metaFlye + HairSplitter	54.665	1.111	11999	5	120.35	53.81
5x*	metaFlye	12.020	1.010	-	2	849.88	201.29
	metaFlye + Strainberry	9.807	1.013	-	2	422.61	273.49
	metaFlye + HairSplitter	25.052	1.042	-	2	311.52	166.26
divergence							
H5	metaFlye	54.246	1.002	19206	22	324.97	29.23
(1.09%)	metaFlye + Strainberry	98.157	1.001	652572	2	324.97	1.35
	metaFlye + HairSplitter	99.666	1.042	196697	2	1.33	7.74
AMSCJX03	metaFlye	54.783	1.001	18675	17	254.48	28.42
(0.91%)	metaFlye + Strainberry	93.390	1.003	279448	3	0.73	1.98
	metaFlye + HairSplitter	92.249	1.003	18001	0	3.09	4.17
RM74721	metaFlye	54.254	1.000	19360	19	132.60	11.73
(0.57%)	metaFlye + Strainberry	92.256	1.006	380826	1	2.97	8.57
	metaFlye + HairSplitter	85.936	1.006	10000	0	3.56	4.98
EC590	metaFlye	54.132	1.000	17337	10	117.79	12.85
(0.45%)	metaFlye + Strainberry	71.749	1.003	156627	10	9.29	1.72
	metaFlye + HairSplitter	78.957	1.005	6001	6	8.48	5.10
Y5	metaFlye	54.736	1.002	22104	21	63.85	9.38
(0.38%)	metaFlye + Strainberry	72.758	1.006	181387	13	8.64	3.28
	metaFlye + HairSplitter	75.774	1.009	6000	1	1.68	1.89
LD27-1	metaFlye	53.295	1.001	19411	8	43.83	5.33
(0.27%)	metaFlye + Strainberry	62.101	1.004	112749	13	7.41	3.53
	metaFlye + HairSplitter	65.674	1.007	4000	0	3.54	2.80
ME8067	metaFlye	50.820	1.000	47356	8	10.29	3.19
(0.07%)	metaFlye + Strainberry	50.820	1.000	47356	8	10.29	3.19
	metaFlye + HairSplitter	63.628	1.002	4000	0	1.57	1.02

Table 3. metaQuast metrics of the bacterial assemblies obtained from simulated Nanopore R10.4.1 data. * The metrics displayed for the downsampled datasets are the metrics computed with respect to the downsampled strain, and not with respect to the complete 10 strains.

		Genome Fraction (%)	Duplication ratio	NGA50	#misassemblies	# mismatches per 100 kbp	# indels per 100 kbp
HBV-2	Strainberry	99.984	2.174	4504	3	881.59	1562.50
	iGDA	54.174	1.001	1081	0	201.15	229.89
	Strainline						
	HaploDMF	99.984	1.000	3207	0	15.58	93.46
	HairSplitter	99.953	1.001	3209	0	46.72	109.02
norovirus	Strainberry	14.283	1.000	-	0	52.97	13.24
	iGDA	69.514	1.548	2838	0	112.55	15.83
	Strainline	29.659	5.787	7541	0	479.44	136.67
	HaploDMF	85.702	1.000	7549	0	165.60	26.50
	HairSplitter	100.000	1.038	7550	0	107.57	35.96

Table 4. metaQuast metrics of the viral assemblies.