Decision for the manuscript "HairSplitter: haplotype assembly from long, noisy reads"

Both reviewers indentify substantial merits of the submitted work. In particular, Reviewer 1 has some minor reservations regarding terminology and missing explanations. Reviewer 2, instead, asks for a better discussion about correctness and completeness of the method.

This preprint merits a revision to adress the comments of the reviewers. Apart for some minor editorial modifications (like captions, terminology, and citation fixes), two major things are required in a revised version of the manuscript:

- Discuss what part is novel and what components are re-used from the state of the art (Reviewer 1).

- Discuss also the aspect of assembly contituity and not only that of completeness (Reviewer 2). Perhaps, some more experiments illustrating the trade-off between contiguity, correctness, and completeness offered by HairSplitter could be presented.
by Giulio Ermanno Pibiri, 22 Apr 2024 07:53

To adress the comments of the reviewers, HairSplitter was improved and the examples provided in the paper were re-run with its current version (1.9.4). The main change introduced was the suppression of the 5 SNP-threshold, but we also found and corrected several bugs in the unzipping process thanks to feedback by users of our program. All the figures have been updated accordingly.

Manuscript: https://doi.org/10.1101/2024.02.13.580067
version: 1
Review by anonymous reviewer 1, 27 Mar 2024 10:26

SUMMARY

The authors provide a software HairSplitter to separate (phase) assemblies into strain-haplotypes using a strain-oblivious assembly and ONT or PacBio HiFi reads as input. As highly similar strains have been shown to have very different functional roles, software for accurate strain-specific assembly is needed. Several tools already exist for this (reported by the authors). However, the authors' software substantially improves over other state-of-the-art methods for noisier ONT reads, while performing reasonably well on PacBio HiFi reads.

I find the paper easy to follow and digest (with the exceptions listed in significant comments 1 and 2). The methods, while being succinctly presented, are well described. In particular, the variant calling procedure is simple and elegantly described.

Furthermore, the experiments are well performed against state-of-the-art using both simulated and biological data. The results are clearly presented and easy to follow. HairSplitter's limitations, such as requiring spanning at least five polymorphic loci, are adequately discussed. Two future directions of this work are also described and seem reasonable.

I only have minor-type comments of various significance.

SIGNIFICANT

1. Terminology: The authors use "a new read clustering algorithm" in the abstract, but clustering is not used in the paper. Instead, terms like 'read separation' is used in the figure 1, and 'read binning' is used as a section header. Are they all referring to the same thing? Also, 'scaffolding' (step e in Figure 1) is missing in the text. Is scaffolding the 'duplication' procedure described in the reassembly section? Please address this and use the same terms whenever possible.

"Read clustering", "read separation" and "read binning" indeed all refered to the same process - we harmonized the terminology by using "read binning" throughout the article. Scaffolding referred indeed to the duplication procedure described in the assembly section. To make things clearer and coherent with the GraphUnzip paper (https://www.biorxiv.org/content/10.1101/2021.01.29.428779v1.full), we renamed this process "unzipping" throughout the manuscript (including Figure 1) - as it is slightly different from what is usually called scaffolding. The paragraph describing the unzipping has been given a title separately from the "reassembly" section, as it is described as a separate step in the "overview of the pipeline" and Figure 1.

2. I was left wondering what methods were novel in this paper and what was re-used. The authors mention that the variant calling procedure is based on an already-explored idea (Z Feng et al., 2021). However, the methods in Z Feng et al., 2021 are quite dense to read. The authors should describe in more detail the similarities/differences to the approach (Z Feng et al., 2021). Also, if the read clustering(/binning) is novel, it could be emphasized not only in the abstract but also in the text.

The paper of Z Feng et al. states that "intuitively, assuming that sequencing errors are independent, the same combination of sequencing errors at multiple loci is unlikely to repeatedly occur together on multiple reads." This is also the core of our reasoning. However, the methods are completely different. Z Feng et al. evaluate all positions separately by estimating the bayesian probability that it is a SNP given all other positions. Our paper first detect groups of positions heuristically and then runs a statistical test on the groups (and not on the individual positions) to check wether this group can statistically be explained by sequencing errors or not. The algorithms are subsequently completely different. The first approach is more thorough (since the entirety of the pileup is taken into account at each position), but this forces Feng et al. (2021) to make approximations to make the problem tractable, which we believe is the reason iGDA does not perform as well as HairSplitter.

To clarify the article, we moved the reference to Feng et al. (2021) down to the 3rd paragraph of "Mathematical model behind variant calling", which we rephrased to:
"The key lies in taking several loci into account simultaneously, an idea already explored in Feng et al. (2021) and leveraging the assumption that alignment artifacts occur randomly in the pileup while genomic variant are expected to be correlated along the alignment. Consequently, pileups at polymorphic loci are expected to exhibit strong correlation, contrary to pileups at non-polymorphic loci. HairSplitter introduces a new statistical approach and a new algorithm to exploit this observation and detect even rare strains, as illustrated below."

MINOR

- It should be mentioned what the input and output formats of HairSplitter are at some point early in the paper (e.g., GFA/fasta assembly format and reads in fastq?).

In "overview of the pipeline", we added the sentence: "HairSplitter takes as input an assembly (in fasta format) or an assembly graph (in gfa format) as well as sequencing reads (fasta/q), and produces a new assembly (fasta and gfa)."

- The authors' statistical derivation for variant calling is elegant, both in the approach and in its simple presentation, allowing the reader to quickly absorb the approach. In addition to significant comment 2, I had some minor comments on it:
- Do the authors differentiate between indels and substitutions? What about indels spanning more than one position in the pileup?

Indels and substitutions are considered in the same way, the indel character being treated as a 5th base. This was clarified in the sentence "HairSplitter then traverses the pileup, determining, for each position, the majority allele and the main alternative allele (either a base or an indel)." Long indels are treated as multiple adjacent loci, which does not necessitate any special exceptions.

- It would be preferable if the authors denoted the error (currently e) to any other letter (e.g., p?). The inattentive reader could confuse e^{ab} with the exponential function.

The letter denoting the error has been changed to \epsilon.

- The authors mention that "clusters with more than five positions are deemed robust," but their model would be able to "handle" smaller b if the assumptions were valid. I have worked with similar approaches, and in my experience, it is the assumption that 'errors are independent' that typically does not fit biological data (especially indels in homopolymer regions). It is, in my opinion, totally fine from a modeling perspective to make the simplification of independent errors, but it would help if the authors could describe their reason for this lower threshold in more detail beyond "to avoid the inadvertent selection of artifact-prone positions." Is the authors' experience the same as mine with indels and homopolymers?

Before seeing this comment, we had tried using only the statistical test described above to select clusters and it had failed. After reading this comment, we went back to see exactly what did not work. It turns out that it was only a question of subtleties in the implementation. We reworked the code and successfully suppressed this five-positions limit. This improved quite significantly the completeness of the assembly of the low-divergence dataset ME8067/K12 dataset (0.07 % divergence), for which the metaQuast completeness increased from 64 % to 86 %. The hypothesis that errors occur on random reads thus seems to hold reasonably well for the tested datasets.

The paragraph was rewritten: "After all positions have been considered, clusters are tested using the statistical model described above and only clusters with a p-value below 0.001 are kept. The corresponding positions are outputted as polymorphic sites."

- I believe 'MetaQUAST completeness' (Figures) and "Genome Fraction %" (Suppl. Tables) are the same data presented twice under different names. I suggest using consistent naming (or removing duplicated data presentation).

The legend of the tables has been changed to "completeness"

- Supplementary Table 2: Is there a missing comma in Hairsplitter's #mismatches per 100kbp? 31944 seems high. It is potentially a copy-paste error since the NGA50 is also 31944 for that experiment.

This has been corrected. All values concerning the output of HairSplitter were updated anyway since the version of HairSplitter has changed.

- Legend of Figure 3 should include "V.fluvialis" dataset: "27-mer completeness, MetaQUAST completeness and run-time of different software on the V.fluvialis and the three Zymo-GMS datasets."

This was changed.

- I had to download a 5.9Gb file for information on how the tools were run (commands and parameters). It could be added to supplementary data or available in some other form separate from the data.

The Zenodo archive was uploaded, this time with the reads and assemblies in a different file from the command lines.

Review by Dmitry Antipov, 21 Apr 2024 22:49

Questions:
Does the title clearly reflect the content of the article?
Yes
Does the abstract present the main findings of the study?
Possibly, Not completely convinced (will explain later)
Are the research questions/hypotheses/predictions clearly presented?
Yes
Does the introduction build on relevant research in the field?
Yes
Are the methods and analyses sufficiently detailed to allow replication by other researchers?
Yes
Are the methods and statistical analyses appropriate and well described?
Yes, with minor questions
Are the results described and interpreted correctly?
Not convinced, will explain later
Have the authors appropriately emphasized the strengths and limitations of their study/theory/methods/argument?
Questionable

Are the conclusions adequately supported by the results (without overstating the implications of the findings)?
Questionable

Major issues:
As for me, in the assembly-related paper the most important part is the results section. And here I see a clear problem on focusing on only completeness, with contiguity issues completely ignored in the main text. Supplementary table 3 shows that for some of the strains NA50 of HairSplitter is dramatically lower(6K vs 180K!) than for competitiors or even the original Flye assembly. I'm not sure whether such fragmented assembly, without haplotype labels, makes lots of sence for downstream analysis.
This is quite possible that tradeoff between contiguity, correctness and completeness favors HairSplitter, but this definitely should be better shown and discussed in the main text. As for now I'm not convinced.

The unzipping procedure has been extensively revised and debugged. The module responsible for the unzipping did not introduce errors but was uselessly cautious in many situations. After debugging (and other changes prompted by the comments of reviewer 1), all the test were re-run with HairSplitter v1.9.4. This showed a great increase in contiguity: for example, the NGA50 of the Y5 HairSplitter assembly went from 6K to 254K. The contiguity problem has thus been greatly mitigated. Nevertheless, we introduced a comment on the slightly worse contiguity compared to Strainberry "Particularly with Nanopore data, HairSplitter produced the most complete assemblies, though less contiguous than those produced by Strainbery."

Assembly correction stage name is likely misleading.
If significant amount of reads stops aligning, this may be still not a misassembly if another reads align. Consider two strains, ABC and AB'C
It's quite possible for the assembler to report it as one large contig ABC and additional contig B'. HairSplitter may split ABC into three separate contigs A, B, C since reads from the second strain stop aligning at B. Both {A,B,B',C} & {ABC, B'} are correct, there are no misassemblies or any other error in any of those representations.
This stage itself still makes sence in this case, since HairSplitter has it's own reassembly step, and it would be hard to connect B' with "part" of ABC without such splitting.
Also, some assemblers (not sure about Flye though) can output not only final contigs but unitigs or nodes of the underlying graph - is it what is really required from this step?

Because minigraph was used, possibly reads are aligned not to the assembly(contigs) but to the assembly _graph_? This would reduce my concerns about this step, although this step should be still evaluated separately, i.e. those breakpoints can be compared with quast-reported misassemblies.

To adress these pertinent comments, we rephrased the paragraph, renamed this step and added an evaluation of this process. The new version is:
"Completion of the assembly graph
To work well, HairSplitter needs as input an assembly graph on which all non-chimeric reads align from end to end, which we define as a ``complete'' assembly graph. If the assembly was not provided as a graph, it is turned into an incomplete graph with no edges. Collapsed assembly graphs are also often incomplete because of contigs that have been detached from their neighbors and of collapsed structural variation between strains.

Aligning reads on an incomplete graph translates as locations where a significant number of reads stop aligning, which we call breakpoints. Breakpoints can occur in the middle or the end of contigs. To complete the initial assembly graph, the reads are aligned on the graph using minigraph. The assembly is subsequently examined for breakpoints and 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. An evaluation of this step in terms of misassemblies and contiguity is provided in Supplementary Table 5.

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

Minor issues:

Line 23: metaMDBG is not HiFi only but work on the ONT too. It would be really benefitial to include this tool in all the comparisons.

When we tested metaMDBG on our datasets it recovered very poorly the strain diversity (e.g. on the Zymo HiFi dataset the metaQuast completeness was below 50%). This is because metaMDBG explicitely tries to collapse similar strains (in the metaMDBG article: "This ‘local progressive abundance filter’ removes complex errors, inter-genomic repeats and strain variability (Fig. 1c)"). It thus does not seem fair to compare it against other methods regarding strain recovery. Thus, we removed the reference to metaMDBG.

Line 28: To the best of my knowledge, ONT has even bigger limitations regarding the quantity of DNA than hifi. Anyway, some reference is required here.

It is true that ultra-low input protocols exist for HiFi sequencing. The field is changing fast and we could not find publication comparing objectively the pros and cons of each method in terms of input DNA vs. sequencing biases. Hence, to avoid entering a complex discussion, we decided to change this paragraph and underline the ease of use and low price of Nanopore sequencing:

“Long reads with extremely low error rate, such as PacBio HiFi reads, have been used to distinguish finely strains with the help of specialized software such as hifiasm and stRainy. However, this challenge has not been yet successfully tackled in the case of noisier reads such as ``regular'' PacBio data or Oxford Nanopore Technology (ONT) reads, the latter of which can be obtained very rapidly on cheap sequencers that are small enough to be carried into the field”

Line 83: closing bracket missing
This has been corrected

Lines 107-112: It is not clear whether described clusterisation is limited by contigs' ends or not (I suppose it is), clarification needed.
It is indeed. We propose: “To generate the pileup, all reads are aligned to the contigs of the assembly using minimap2 using default settings. HairSplitter then traverses the pileup of each contig and determines, for each position, the majority allele and the main alternative allele (either a base or a special "deletion" character)”

Line 107: Minimap2 has different settings presets for hifi and ont (-x option) Were they actually meant as default? Or no options other than input/output and threads were used?

HairSplitter now contains an option -x which gets passed on to minimap2. We suppressed ‘with default settings’.

Line 103: What about structural variations? Major ones likely were splitted in assembly correction step, but how are minor one processed?
No special case is made for small structural variations. They align on the reference contig as deletions, insertions and substitutions. Deletions and insertions are easy to deal with, as they are just handled as a fifth base in the pileup. Trickier variants like inversions will result in a complex series of substitutions, insertions and deletion. They are particularly hard to deal with, as a single sequencing error can alter significantly the alignment of two sequences that do not align well. This could be a consideration to improve HairSplitter, but in our tests we did not observe big problems because of this.

Line 136: is it the same k as above (5)? Anyway, further explanation of the chinese whispers algorithm would be helpful - i.e. can it only split some of the previous clusters or output something completely independent from that inital clusterisation?

Yes, the k is the same as above, even though it is not necessary. The idea is that these two steps should not add errors if there are more than 5 reads in the haplotype. The Chinese Whispers algorithm is now briefly described: ‘The Chinese Whispers algorithm iteratively assign reads to the most represented cluster among their neighbors until convergence.’ and ‘the clusters obtained in the second step are unlikely to be significantly altered, but very small clusters will likely be merged with other close cluster.’

It is highly unlikely that the Chinese whispers split existing groups, since at this step reads are grouped if and only if they are identical, thus will be linked in a clique in the Chinese Whisper graph.

Line 146: Racon ref is likely incorrect, I suppose it should be https://genome.cshlp.org/content/early/2017/01/18/gr.214270.116

Indeed, this has been corrected.

Line 181: To show the tools performance on the real data it can be benefitial to use estimated strains coverage from some of the already studied datasets and not just 30,20,10,5. metaMDBG uses a zymo dataset with uneven coverage between species/strains (in the contrast to used in HairSplitter), it can be benefitial to compare on it

metaFlye was already quite successful at distinguishing the highly divergent strains of the datasets used in the metaMDBG paper, thus HairSplitter would be of little use in such case (cf supplementary material of the metaMDBG paper), the only exception being the Zymobiomics Gut Mcrobiome Standard sequencing, which we used to benchmark HairSplitter. Analysing a long-read dataset of highly similar strains mixed in uneven proportions would be a very good test of the performance of HairSplitter indeed, but unfortunately we could not find any published long-read sequencing data obtained from such mock community.

Supplementary result tables: would be also nice to have total assembly length there

This has been added

Discussion: assembly/assembly graph alignment for assembly correction should clarified.

Now we define the procedure as assembly graph completion, which we believe is clearer.

Additional changes introduced during the revision step:
-The analysis of the results of the Zymobiomics HiFi dataset was updated. We investigated the reason of the high duplication ratio of the different assemblers. Our first hypothesis, that the dataset contained hidden strain variation, did not hold. Rather, the duplications stem from two factors: 1) the difficulty to duplicate identical regions to their exact multiplicity; 2) duplication errors in the original metaFlye assembly are "hidden" as mismatches in metaQuast because contigs are long and span entire repeated regions and flanking regions. stRainy, Strainberry and Hairsplitter cut the contigs into smaller pieces which then are identified correctly as duplications. This led us to rewrite the analysis of the HiFi results:

"On HiFi reads, the stRainy, hifiasm and HairSplitter assemblies depicted a high k-mer completeness. However, they showed either a high duplication ratio (for stRainy and hifiasm) or low metaQuast completeness (for HairSplitter) because none managed to duplicate repeated genomic regions to their correct multiplicities. This effect is also observed in several Nanopore assemblies, where 27-mer completeness remains high while MetaQUAST completeness is notably lower. Typically, the three almost identical *V. fluvialis* strains were assembled as one."

Since the superiority of hifiasm on this dataset over stRainy and HairSplitter is not established, the conclusion regarding HiFi reads has been changed: 'HairSplitter proved useful when dealing with noisy data (> 1% error rate), whereas its usefulness on HiFi reads compared to specialised software such as hifiasm or stRainy is debatable.'

-Because of the upgrade in HairSplitter version, the performance of HairSplitter on highly similar genomes improved dramatically. Consequently, we changed the paragraph describing the results of the simulated benchmark to "The completeness also decreased slightly with the divergence of the strains, though the metaQuast completeness remained high (84%) when assembling two strains with 0.07% divergence"

- Because of the changes introduced in HairSplitter, the limitations underlined in the conclusion are not relevant anymore, and the part about directions for future work was rewritten: 'HairSplitter encounters a major limitation when strains have many homozygous regions. In these regions, it is not possible to assign reads to specific haplotype groups, making it necessary to duplicate the homozygous regions to their correct multiplicity in order to fully recover the strains. This study has demonstrated that this is a challenging problem that current assemblers have not been able to successfully address in the HiFi dataset. Further investigation is needed to solve this issue. A possible way would be to use astutely the topology of the assembly graph.'