Bioinformatics, 37(23), 2021, 4572–4574
https://doi.org/10.1093/bioinformatics/btab705
Advance Access Publication Date: 8 October 2021
Applications Note

Sequence analysis

New strategies to improve minimap2 alignment accuracy

Heng Li [1,2,*]

[1]Department of Data Sciences, Dana-Farber Cancer Institute, Boston, MA 02215, USA and [2]Department of Biomedical Informatics, Harvard Medical School, Boston, MA 02215, USA

*To whom correspondence should be addressed.

Associate Editor: Can Alkan

Received on August 7, 2021; revised on October 4, 2021; editorial decision on October 5, 2021; accepted on October 6, 2021

Abstract

Summary: We present several recent improvements to minimap2, a versatile pairwise aligner for nucleotide sequences. Now minimap2 v2.22 can more accurately map long reads to highly repetitive regions and align through insertions or deletions up to 100 kb by default, addressing major weakness in minimap2 v2.18 or earlier.

Availability and implementation: https://github.com/lh3/minimap2.

Contact: hli@ds.dfci.harvard.edu

1 Introduction

Minimap2 (Li, 2018) is widely used for mapping long sequence reads and assembly contigs. Jain *et al.* (2020b) found minimap2 v2.18 or earlier occasionally misaligned reads from highly repetitive regions as minimap2 ignored seeds of high occurrence. They also noticed minimap2 may misplace reads with structural variations (SVs) in such regions (Jain *et al.*, 2020a). These misalignments have become a pressing issue in the advent of telomere-to-telomere human assembly (Miga *et al.*, 2020). Meanwhile, old minimap2 was unable to efficiently align long insertions/deletions (INDELs) and often breaks an alignment around variable-number tandem repeats (VNTRs). This has inspired new chaining algorithms (Li *et al.*, 2020; Ren and Chaisson, 2021) which are not integrated into minimap2. Here, we will describe recent efforts implemented in v2.19 through v2.22 to improve mapping results.

2 Materials and methods

2.1 Rescuing high-occurrence *k*-mers

Minimap2 keeps all k-mer minimizers (Roberts *et al.*, 2004) during indexing. Its original implementation only selected low-occurrence minimizers during mapping. The cutoff is a few hundred for mapping long reads against a human genome. If a read harbors only a few or even no low-occurrence minimizers, it will fail chaining due to insufficient anchors.

To resolve this issue, we implemented a new heuristic to add additional minimizers. Suppose, we are looking at two adjacent low-occurrence k-mers located at position x_1 and x_2, respectively. If $|x_1 - x_2| \geq L$, minimap2 v2.22 additionally selects $\lfloor |x_1 - x_2|/L \rfloor$ minimizers of the lowest occurrence among minimizers between x_1 and x_2. Here, parameter L controls the frequency of sampling. It defaults to 500. This strategy adds necessary anchors at the cost of increasing total alignment time by a few percent on real data.

2.2 Aligning through longer INDELs

The original minimap2 may fail to align long INDELs due to its chaining heuristics. Briefly, minimap2 applies dynamic programming (DP) to chain minimizer anchors. This is a quadratic algorithm, slow for chaining contigs. For acceptable performance, the original minimap2 uses a 500 bp band by default, which means a gap longer than 500 bp will stop chaining. To align through longer gaps, older minimap2 implemented a long-join heuristic as follows. If there is an INDEL longer than 500 bp and the two chains around the INDEL have no overlaps on either the query or the reference sequence, minimap2 may join the two short chains later. This heuristic may fail around VNTRs because short chains often have overlaps in VNTRs. More subtly, minimap2 may escape the inner DP loop early, again for performance, if the chaining result is not improved for 50 iterations. When there is a copy number change in a long segmental duplication, the early escape may break around the event even if users specify a large band.

In minigraph (Li *et al.*, 2020), we developed a new chaining algorithm that finds up to 1 kb INDELs with DP-based chaining and goes through longer INDELs with a subquadratic algorithm (Abouelhoda and Ohlebusch, 2003). We ported the same algorithm to minimap2 for contig mapping. For long-read mapping, the minigraph algorithm is slower. Minimap2 v2.22 still uses the DP-based algorithm to find short chains and then invokes the minigraph algorithm to rechain anchors in these short chains. The rechaining step achieves the same goal as long-join but is more reliable because it can resolve overlaps between short chains. The old long-join heuristic has since been removed.

2.3 Properly mapping long reads with SVs

The original minimap2 ranks an alignment by its Smith-Waterman score and outputs the best scoring alignment. However, when there are SVs on the read, the best scoring alignment is sometimes not the correct alignment. Jain *et al.* (2020a) resolved this dilemma by altering the mapping algorithm.

In our view, this problem is rooted in inappropriate scoring: affine-gap penalty over-penalizes a long INDEL that was often evolutionarily created in one event. We should not penalize an SV by a function linear in the SV length. Minimap2 v2.22 instead rescores an alignment with the following scoring function. Suppose an alignment consists of M matching bases, N substitutions and G gap opens, we empirically score the alignment with

$$S = M - \frac{N+G}{2d} - \sum_{i=1}^{G} \log_2(1+g_i)$$

where $g_i \geq 1$ is the length of the ith gap and

$$d = \max\left\{\frac{N+G}{M+N+G}, 0.02\right\}$$

It approximates per-base sequence divergence except with the smallest value set to 2%. As an analogy to affine-gap scoring, the matching score in our scheme is 1, the mismatch and gap open penalties are both $1/2d$ and the gap extension penalty is a logarithm function of the gap length (Gu and Li, 1995). Our scoring gives a long SV a much milder penalty. In terms of time complexity, scoring an alignment is linear in the length of the alignment. The time spent on rescoring is negligible in practice.

3 Results

We evaluated minimap2 v2.22 along with v2.18, Winnowmap2 v2.03 and lra v1.3.2 (Table 1), using the default setting of each mapper according to the input data types. Both versions of minimap2 achieved high mapping accuracy on simulated Nanopore reads (sim-map). Winnowmap2 aligned more reads at mapping quality 10 or higher (mapQ10). However, it may occasionally assign a high mapping quality to a read with multiple identical best alignments. This reduced its mapping accuracy.

In lack of ground truth for real data, we took Winnowmap2 mapping as ground truth to evaluate other mappers (winno-cmp in Table 1). Out of 1 378 092 reads with mapQ10 alignments by Winnowmap2, minimap2 v2.22 could map all of them. Eight hundred and eighteen reads, less than 0.01% of all reads, were mapped differently by v2.22. 51 of them have multiple identical best alignments. We believe these are more likely to be Winnowmap2 errors. Most of the remaining 67 (=118−51) reads have multiple highly similar but not identical alignments. Minimap2 v2.18 is less consistent with 275 differences including 30 unmapped reads mappable by both Winnowmap2 and v2.22.

For the minimizer rescuing parameter L in Section 2.1, we set its default to 500 such that v2.22 has comparable performance to v2.18 given simulated PacBio and Nanopore human reads. To see the effect of this parameter on real data, we tried several different L values. v2.22 gave 99 mapping differences at $L=200$, 118 at $L=500$ (default), 167 at $L=750$ and 224 differences at $L=1000$ in comparison to Winnowmap2. $L=200$ is 28% slower than the default while $L=1000$ is 9% faster. Changing the default minimizer window size (option '-w') and the initial minimizer occurrence cutoff (option '-f') also affects performance and accuracy to a similar magnitude.

The two benchmarks above only evaluate read mappings when there are no variations between the reads and the reference. To measure the mapping accuracy in the presence of SVs (sim-sv), we reproduced the results by (Jain *et al.*, 2020a). Minimap2 v2.22 is as good as Winnowmap2 now. Note that we were setting the Sniffles mapping quality threshold to 10 in consistent with the benchmarks above. If we used the default threshold 20, v2.22 would miss additional five SVs (accounting for 0.5% of simulated SVs). For four out of these five missing SVs, minimap2 v2.22 mapped more variant reads than Winnowmap2. Sniffles did not call these SVs because minimap2 tended to give them conservative mapping quality. It is worth noting that the simulation here only considers a simple scenario in evolution. Non-allelic gene conversions, which happen often in segmental duplications (Harpak *et al.*, 2017), would obscure the optimal mapping strategies. How much such simple SV simulation informs real-world SV calling remains a question.

To see if minimap2 v2.22 could improve long INDEL alignment, we ran dipcall on contig-to-reference alignments and focused on INDELs longer than 1 kb (real-sv-1k). v2.22 is more sensitive at comparable specificity, confirming its advantage in more contiguous alignment. We could not get dipcall to work well with lra, so did not report the numbers.

Minimap2 spends most computing time on base alignment. As recent improvements in v2.22 incur little additional computing and do not change the base alignment algorithm, the new version

Table 1. Evaluation of minimap2 v2.22. Numbers in the bold fontface indicates the best performing tools

[Benchmark] Metric	v2.22	v2.18	Winno	lra
[sim-map] % mapped reads at Q10	97.9	97.6	**99.0**	97.3
[sim-map] err. rate at Q10 (phredQ)	**52**	**52**	38	24
[winno-cmp] rate of diff. (phredQ)	**41**	37	truth	18
[winno-cmp] CPU time (hour)	**5.0**	5.3	71.8	13.1
[winno-cmp] peak RAM (Gb)	17.1	14.4	**9.6**	12.4
[sim-sv] % false negative rate	**0.5**	2.0	**0.5**	1.4
[sim-sv] % false discovery rate	**0.0**	0.1	**0.0**	0.1
[real-sv-1k] % false negative rate	**7.3**	20.0	13.0	N/A
[real-sv-1k] % false discovery rate	2.7	**2.4**	2.7	N/A

Note: In [sim-map], 152 713 reads were simulated from the CHM13 telomere-to-telomere assembly v1.1 (AC: GCA_009914755.3) with pbsim2 (Ono *et al.*, 2021): 'pbsim2 –hmm_model R94.model –length-min 5000 –length-mean 20000 –accuracy-mean 0.95'. Alignments of mapping quality 10 or higher were evaluated by 'paftools.js mapeval'. The mapping error rate is measured in the phred scale: if the error rate is e, $-10\log_{10}e$ is reported in the table. In [winno-cmp], 1.39 million CHM13 HiFi reads from SRR11292121 were mapped against the same CHM13 assembly. 99.3% of them were mapped by Winnowmap2 at mapping quality 10 or higher and were taken as ground truth to evaluate minimap2 and lra with 'paftools.js pafcmp'. [sim-sv] simulated 1000 50 bp to 1000 bp INDELs from chr8 in CHM13 using SURVIVOR (Jeffares *et al.*, 2017) and simulated Nanopore reads at 30-fold coverage with the same pbsim2 command line. SVs were called with 'sniffles -q 10' (Sedlazeck *et al.*, 2018) and compared with the simulated truth with 'SURVIVOR eval call.vcf truth.bed 50'. In [real-sv-1k], small and long variants were called by dipcall-0.3 (Li *et al.*, 2018) for HG002 assemblies (AC: GCA_018852605.1 and GCA_018852615.1) and compared to the GIAB truth (Zook *et al.*, 2020) using 'truvari -r 2000 -s 1000 -S 400 –multimatch –passonly' which sets the minimum INDEL size to 1 kb in evaluation.

has similar performance to older versions. It is consistently faster than Winnowmap2 by several times. Sometimes simple heuristics can be as effective as more sophisticated yet slower solutions.

Acknowledgements

The authors thank Arang Rhie and Chirag Jain for providing motivating examples for which older minimap2 underperforms.

Funding

This work was funded by National Human Genome Research Institute grant [R01HG010040].

Conflict of Interest: none declared.

References

Abouelhoda,M.I. and Ohlebusch,E. (2003) A local chaining algorithm and its applications in comparative genomics. In: *Algorithms in Bioinformatics, Third International Workshop, WABI 2003*, *Budapest*, *Hungary, September 15–20*, *2003*, *Proceedings*, pp. 1–16.

Gu,X. and Li,W.H. (1995) The size distribution of insertions and deletions in human and rodent pseudogenes suggests the logarithmic gap penalty for sequence alignment. *J. Mol. Evol.*, **40**, 464–473.

Harpak,A. *et al.* (2017) Frequent nonallelic gene conversion on the human lineage and its effect on the divergence of gene duplicates. *Proc. Natl. Acad. Sci. USA*, **114**, 12779–12784.

Jain,C. *et al.* (2020a) A long read mapping method for highly repetitive reference sequences. *bioRxiv*, 10.1101/2020.11.01.363887.

Jain,C. *et al.* (2020b) Weighted minimizer sampling improves long read mapping. *Bioinformatics*, **36**, i111–i118.

Jeffares,D.C. *et al.* (2017) Transient structural variations have strong effects on quantitative traits and reproductive isolation in fission yeast. *Nat. Commun.*, **8**, 14061.

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

Li,H. *et al.* (2018) A synthetic-diploid benchmark for accurate variant-calling evaluation. *Nat. Methods*, **15**, 595–597.

Li,H. *et al.* (2020) The design and construction of reference pangenome graphs with minigraph. *Genome Biol.*, **21**, 265.

Miga,K.H. *et al.* (2020) Telomere-to-telomere assembly of a complete human X chromosome. *Nature*, **585**, 79–84.

Ono,Y. *et al.* (2021) PBSIM2: a simulator for long-read sequencers with a novel generative model of quality scores. *Bioinformatics*, **37**, 589–595.

Ren,J. and Chaisson,M.J.P. (2021) lra: a long read aligner for sequences and contigs. *PLoS Comput. Biol.*, **17**, e1009078.

Roberts,M. *et al.* (2004) Reducing storage requirements for biological sequence comparison. *Bioinformatics*, **20**, 3363–3369.

Sedlazeck,F.J. *et al.* (2018) Accurate detection of complex structural variations using single-molecule sequencing. *Nat. Methods*, **15**, 461–468.

Zook,J.M. *et al.* (2020) A robust benchmark for detection of germline large deletions and insertions. *Nat. Biotechnol.*, **38**, 1347–1355.