NAME

minimap2 - mapping and alignment between collections of DNA sequences

SYNOPSIS

* Indexing the target sequences (optional):

minimap2 [**-x** *preset*] **-d** *target.mmi target.fa*
minimap2 [**-H**] [**-k** *kmer*] [**-w** *miniWinSize*] [**-I** *batchSize*] **-d** *target.mmi target.fa*

* Long-read alignment with CIGAR:

minimap2 **-a** [**-x** *preset*] *target.mmi query.fa* > *output.sam*
minimap2 **-c** [**-H**] [**-k** *kmer*] [**-w** *miniWinSize*] [**...**] *target.fa query.fa* > *output.paf*

* Long-read overlap without CIGAR:

minimap2 **-x** ava-ont [**-t** *nThreads*] *target.fa query.fa* > *output.paf*

DESCRIPTION

Minimap2 is a fast sequence mapping and alignment program that can find overlaps between long noisy reads, or map long reads or their assemblies to a reference genome optionally with detailed alignment (i.e. CIGAR). At present, it works efficiently with query sequences from a few kilobases to ~100 megabases in length at a error rate ~15%. Minimap2 outputs in the PAF or the SAM format.

OPTIONS

Indexing options

-k *INT* Minimizer k-mer length [15]

-w *INT* Minimizer window size [2/3 of k-mer length]. A minimizer is the smallest k-mer in a window of w consecutive k-mers.

-H Use homopolymer-compressed (HPC) minimizers. An HPC sequence is constructed by contracting homopolymer runs to a single base. An HPC minimizer is a minimizer on the HPC sequence.

-I *NUM* Load at most *NUM* target bases into RAM for indexing [4G]. If there are more than *NUM* bases in *target.fa*, minimap2 needs to read *query.fa* multiple times to map it against each batch of target sequences. *NUM* may be ending with k/K/m/M/g/G. NB: mapping quality is incorrect given a multi-part index.

--idx-no-seq

Don't store target sequences in the index. It saves disk space and memory but the index generated with this option will not work with **-a** or **-c**. When base-level alignment is not requested, this option is automatically applied.

-d *FILE* Save the minimizer index of *target.fa* to *FILE* [no dump]. Minimap2 indexing is fast. It can index the human genome in a couple of minutes. If even shorter startup time is desired, use this option to save the index. Indexing options are fixed in the index file. When an index file is provided as the target sequences, options **-H**, **-k**, **-w**, **-I** will be effectively overridden by the options stored in the index file.

Mapping options

-f *FLOAT*|*INT1*[,*INT2*]

If fraction, ignore top *FLOAT* fraction of most frequent minimizers [0.0002]. If integer, ignore minimizers occuring more than *INT1* times. *INT2* is only effective in the **--sr** or **-xsr** mode, which sets the threshold for a second round of seeding.

--min-occ-floor *INT*

Force minimap2 to always use k-mers occurring *INT* times or less [0]. In effect, the max occurrence threshold is set to the max{*INT*, **-f**}.

-g *INT* Stop chain enlongation if there are no minimizers within *INT*-bp [10000].

-r *INT* Bandwidth used in chaining and DP-based alignment [500]. This option approximately controls the maximum gap size.

-n *INT* Discard chains consisting of <*INT* number of minimizers [3]

-m *INT* Discard chains with chaining score <*INT* [40]. Chaining score equals the approximate number of matching bases minus a concave gap penalty. It is computed with dynamic programming.

-D If query sequence name/length are identical to the target name/length, ignore diagonal anchors. This option also reduces DP-based extension along the diagonal.

-P Retain all chains and don’t attempt to set primary chains. Options **-p** and **-N** have no effect when this option is in use.

--dual=yes|no
If **no**, skip query-target pairs wherein the query name is lexicographically greater than the target name [yes]

-X Equivalent to ’**-DP --dual=no --no-long-join**’. Primarily used for all-vs-all read overlapping.

-p *FLOAT* Minimal secondary-to-primary score ratio to output secondary mappings [0.8]. Between two chains overlaping over half of the shorter chain (controlled by **--mask-level**), the chain with a lower score is secondary to the chain with a higher score. If the ratio of the scores is below *FLOAT*, the secondary chain will not be outputted or extended with DP alignment later. This option has no effect when **-X** is applied.

-N *INT* Output at most *INT* secondary alignments [5]. This option has no effect when **-X** is applied.

-G *NUM* Maximum gap on the reference (effective with **-xsplice**/**--splice**). This option also changes the chaining and alignment band width to *NUM*. Increasing this option slows down spliced alignment. [200k]

-F *NUM* Maximum fragment length (aka insert size; effective with **-xsr**/**--frag=yes**) [800]

-M *FLOAT* Mark as secondary a chain that overlaps with a better chain by *FLOAT* or more of the shorter chain [0.5]

--max-chain-skip *INT*
A heuristics that stops chaining early [50]. Minimap2 uses dynamic programming for chaining. The time complexity is quadratic in the number of seeds. This option makes minimap2 exits the inner loop if it repeatedly sees seeds already on chains. Set *INT* to a large number to switch off this heurstics.

--no-long-join
Disable the long gap patching heuristic. When this option is applied, the maximum alignment gap is mostly controlled by **-r**.

--splice Enable the splice alignment mode.

--sr Enable short-read alignment heuristics. In the short-read mode, minimap2 applies a second round of chaining with a higher minimizer occurrence threshold if no good chain is found. In addition, minimap2 attempts to patch gaps between seeds with ungapped alignment.

--frag=no|yes
Whether to enable the fragment mode [no]

--for-only Only map to the forward strand of the reference sequences. For paired-end reads in the forward-reverse orientation, the first read is mapped to forward strand of the reference and the second read to the reverse stand.

--rev-only Only map to the reverse complement strand of the reference sequences.

--heap-sort=no|yes
If yes, sort anchors with heap merge, instead of radix sort. Heap merge is faster for short reads, but slower for long reads. [no]

Alignment options

-A *INT* Matching score [2]

-B *INT* Mismatching penalty [4]

-O *INT1[,INT2]*
Gap open penalty [4,24]. If *INT2* is not specified, it is set to *INT1*.

-E *INT1[,INT2]*
Gap extension penalty [2,1]. A gap of length *k* costs $\min\{O1+k*E1,O2+k*E2\}$. In the splice mode, the second gap penalties are not used.

-C *INT* Cost for a non-canonical GT-AG splicing (effective with **--splice**) [0]

-z *INT1[,INT2]*
Truncate an alignment if the running alignment score drops too quickly along the diagonal of the DP matrix (diagonal X-drop, or Z-drop) [400,200]. If the drop of score is above *INT2*, minimap2 will reverse complement the query in the related region and align again to test small inversions. Minimap2 truncates alignment if there is an inversion or the drop of score is greater than *INT1*. Decrease *INT2* to find small inversions at the cost of performance and false positives. Increase *INT1* to improves the contiguity of alignment at the cost of poor alignment in the middle.

-s *INT* Minimal peak DP alignment score to output [40]. The peak score is computed from the final CIGAR. It is the score of the max scoring segment in the alignment and may be different from the total alignment score.

-u *CHAR* How to find canonical splicing sites GT-AG - **f**: transcript strand; **b**: both strands; **n**: no attempt to match GT-AG [n]

--end-bonus *INT*
Score bonus when alignment extends to the end of the query sequence [0].

--splice-flank=yes|no
Assume the next base to a **GT** donor site tends to be A/G (91% in human and 92% in mouse) and the preceding base to a **AG** acceptor tends to be C/T [no]. This trend is evolutionarily conservative, all the way to S. cerevisiae (PMID:18688272). Specifying this option generally leads to higher junction accuracy by several percents, so it is applied by default with **--splice**. However, the SIRV control does not honor this trend (only ˜60%). This option reduces accuracy. If you are benchmarking minimap2 on SIRV data, please add **--splice-flank=no** to the command line.

--end-seed-pen *INT*
Drop a terminal anchor if $s<\log(g)+INT$, where s is the local alignment score around the anchor and g the length of the terminal gap in the chain. This option is only effective with **--splice**. It helps to avoid tiny terminal exons. [6]

Input/output options

-a Generate CIGAR and output alignments in the SAM format. Minimap2 outputs in PAF by default.

-Q Ignore base quality in the input file.

-L Write CIGAR with >65535 operators at the CG tag. Older tools are unable to convert alignments with >65535 CIGAR ops to BAM. This option makes minimap2 SAM compatible with older tools. Newer tools recognizes this tag and reconstruct the real CIGAR in memory.

-R *STR* SAM read group line in a format like **@RG\tID:foo\tSM:bar** [].

-c Generate CIGAR. In PAF, the CIGAR is written to the ‘cg’ custom tag.

--cs[=*STR***]** Output the **cs** tag. *STR* can be either *short* or *long*. If no *STR* is given, *short* is assumed. [none]

-Y In SAM output, use soft clipping for supplementary alignments.

--seed *INT* Integer seed for randomizing equally best hits. Minimap2 hashes *INT* and read name when choosing between equally best hits. [11]

-t *INT* Number of threads [3]. Minimap2 uses at most three threads when indexing target sequences, and uses up to *INT*+1 threads when mapping (the extra thread is for I/O, which is frequently idle and takes little CPU time).

-2 Use two I/O threads during mapping. By default, minimap2 uses one I/O thread. When I/O is slow (e.g. piping to gzip, or reading from a slow pipe), the I/O thread may become the bottleneck. Apply this option to use one thread for input and another thread for output, at the cost of increased peak RAM.

-K *NUM* Number of bases loaded into memory to process in a mini-batch [500M]. Similar to option **-I**, K/M/G/k/m/g suffix is accepted. A large *NUM* helps load balancing in the multi-threading mode, at the cost of increased memory.

--secondary=**yes**|**no**
Whether to output secondary alignments [yes]

--version Print version number to stdout

Preset options

-x *STR* Preset []. This option applies multiple options at the same time. It should be applied before other options because options applied later will overwrite the values set by **-x**. Available *STR* are:

map-pb PacBio/Oxford Nanopore read to reference mapping (**-Hk19**)

map-ont
Slightly more sensitive for Oxford Nanopore to reference mapping (**-k15**). For PacBio reads, HPC minimizers consistently leads to faster performance and more sensitive results in comparison to normal minimizers. For Oxford Nanopore data, normal minimizers are better, though not much. The effectiveness of HPC is determined by the sequencing error mode.

asm5 Long assembly to reference mapping (**-k19 -w19 -A1 -B19 -O39,81 -E3,1 -s200 -z200 --min-occ-floor=100**). Typically, the alignment will not extend to regions with 5% or higher sequence divergence. Only use this preset if the average divergence is far below 5%.

asm10 Long assembly to reference mapping (**-k19 -w19 -A1 -B9 -O16,41 -E2,1 -s200 -z200 --min-occ-floor=100**). Up to 10% sequence divergence.

asm20 Long assembly to reference mapping (**-k19 -w10 -A1 -B6 -O6,26 -E2,1 -s200 -z200 --min-occ-floor=100**). Up to 20% sequence divergence.

ava-pb PacBio all-vs-all overlap mapping (**-Hk19 -Xw5 -m100 -g10000 --max-chain-skip 25**).

ava-ont Oxford Nanopore all-vs-all overlap mapping (**-k15 -Xw5 -m100 -g10000 --max-chain-skip 25**). Similarly, the major difference from **ava-pb** is that this preset is not using HPC minimizers.

splice Long-read spliced alignment (**-k15 -w5 --splice -g2000 -G200k -A1 -B2 -O2,32 -E1,0 -C9 -z200 -ub --splice-flank=yes**). In the splice mode, 1) long deletions are taken as introns and represented as the '**N**' CIGAR operator; 2) long insertions are disabled; 3) deletion and insertion gap costs are different during chaining; 4) the computation of the '**ms**' tag ignores introns to demote hits to pseudogenes.

sr Short single-end reads without splicing (**-k21 -w11 --sr --frag=yes -A2 -B8 -O12,32 -E2,1 -r50 -p.5 -N20 -f1000,5000 -n2 -m20 -s40 -g200 -2K50m --heap-sort=yes --secondary=no**).

Miscellaneous options

--no-kalloc

Use the libc default allocator instead of the kalloc thread-local allocator. This debugging option is mostly used with Valgrind to detect invalid memory accesses. Minimap2 runs slower with this option, especially in the multi-threading mode.

--print-qname

Print query names to stderr, mostly to see which query is crashing minimap2.

--print-seeds

Print seed positions to stderr, for debugging only.

OUTPUT FORMAT

Minimap2 outputs mapping positions in the Pairwise mApping Format (PAF) by default. PAF is a TAB-delimited text format with each line consisting of at least 12 fields as are described in the following table:

Col	Type	Description
1	string	Query sequence name
2	int	Query sequence length
3	int	Query start coordinate (0-based)
4	int	Query end coordinate (0-based)
5	char	'+' if query/target on the same strand; '-' if opposite
6	string	Target sequence name
7	int	Target sequence length
8	int	Target start coordinate on the original strand
9	int	Target end coordinate on the original strand
10	int	Number of matching bases in the mapping
11	int	Number bases, including gaps, in the mapping
12	int	Mapping quality (0-255 with 255 for missing)

When alignment is available, column 11 gives the total number of sequence matches, mismatches and gaps in the alignment; column 10 divided by column 11 gives the BLAST-like alignment identity. When alignment is unavailable, these two columns are approximate. PAF may optionally have additional fields in the SAM-like typed key-value format. Minimap2 may output the following tags:

Tag	Type	Description
tp	A	Type of aln: P/primary, S/secondary and I,i/inversion
cm	i	Number of minimizers on the chain
s1	i	Chaining score
s2	i	Chaining score of the best secondary chain
NM	i	Total number of mismatches and gaps in the alignment
AS	i	DP alignment score
ms	i	DP score of the max scoring segment in the alignment
nn	i	Number of ambiguous bases in the alignment
ts	A	Transcript strand (splice mode only)
cg	Z	CIGAR string (only in PAF)
cs	Z	Difference string

The **cs** tag encodes difference sequences in the short form or the entire query *AND* reference sequences in the long form. It consists of a series of operations:

Op	Regex	Description
=	[ACGTN]+	Identical sequence (long form)
:	[0-9]+	Identical sequence length
*	[acgtn][acgtn]	Substitution: ref to query
+	[acgtn]+	Insertion to the reference
-	[acgtn]+	Deletion from the reference
~	[acgtn]{2}[0-9]+[acgtn]{2}	Intron length and splice signal

LIMITATIONS

* Minimap2 may produce suboptimal alignments through long low-complexity regions where seed positions may be suboptimal. This should not be a big concern because even the optimal alignment may be wrong in such regions.

* Minimap2 requires SSE2 or NEON instructions to compile. It is possible to add non-SSE2/NEON support, but it would make minimap2 slower by several times.

SEE ALSO

miniasm(1), minimap(1), bwa(1).