Novocraft Short Read Alignment Suite

Release 2.07.00 5th August 2010

Introduction

A series of programs designed for accurate and high speed alignment of short reads to reference genomes. Novel features include the use of base qualities in the reads and ambiguous nucleotide codes in the reference sequences for alignment.

Key features:

1. Mapping with base quality values
2. Alignment quality scores
3. Paired end alignment
4. Mismatches and gaps of up to 15bp in single ended reads, longer in paired end reads.
5. Use of ambiguous codes in reference sequences can be used to reduce allelic bias
6. Bisulphite alignment mode for analysis of methylation status.
7. Automatic base quality calibration

The Novo programs use an index of the target or reference sequence and then align reads against the target genome using an iterative algorithm.

Program	Description.
novoindex	A utility to construct an index for the reference sequences. Typically creates a k-mer index that can be loaded into shared memory for access by multiple search processes. The index includes a 4bit per bp compressed copy of the reference sequences.
novoalign	An alignment tool for aligning short sequences against an indexed set of reference sequences. Typically used for aligning Illumina single end and paired end reads. Uses base qualities and affine gap penalties to find the most probable alignment location of the read.
novoalignMPI	A multi-server version of Novoalign that uses MPICH2 messaging passing library to allow multiple servers to cooperate in the alignment of a single file of reads. Command line options are the same as for Novoalign.
novoalignCS	An aligner for reads from the ABI Solid sequencer.
novoalignCSMPI	An MPI version of NovoalignCS

Novoalign is available in two versions, a free version for use in non-profit organisations for internal use and a licensed version that enables some additional features.
Features available in the licensed version are:

1. Multi-threading. Improves performance by using multiple CPUs and improved memory sharing between threads vs processes. When enabled it will create a thread per CPU core on the server unless you use the -c option to reduce the number of threads.
2. Sequence read files in Gzip format can be processed allowing savings in file space. Output files can be compressed by piping into Gzip.

3. A 5' PCR adapter stripping function that is useful with some protocols such as Nimblegen Sequence Capture Arrays where a PCR adapter may have been left on the fragments.
4. BS-Seq, mode for alignment of reads from bisulphite treated DNA.
5. A base quality calibration function that calibrates base qualities based on mismatch rates from actual alignments. This improves sensitivity and specificity and is also useful for recovering alignments from poor quality runs of the Illumina Genome Analyser.
6. Handling of paired end reads where the fragment length is shorter than the read length and the reads have extended into adapter sequence. This function identifies short fragments with adapter by in-silico prepending adapter to each read of a pair and then aligning the two reads to identify short or overlapping fragments. If overlapping reads and adapter are identified the adapter is trimmed from the reads.

NovoalignCS is only available with a license.

Note. A valid license file must be installed adjacent to the executables in order to enable multi-threading and other commercial options. Trial licenses can be obtained from sales@novocraft.com. Please state, organization name and address with requests for trial licenses.

novoindex

First step first. Novoalign requires the target reference sequences to be indexed prior to alignment. The index is saved to a file and can be reused and shared between multiple copies of the aligners. Index construction time is quite fast, a few seconds for a worm to several minutes for human genome so the index can be discarded and rebuilt as required.

Usage:

novoindex *options indexfile sequencefiles....*

Option	Description
-k *99*	is the k-mer length to be used for the index. Typically 14. The indexer will select appropriate values if either of these is not specified.
-s *9*	is the step size for the index. Typical values are from 1 to 3.
-m	lower case masking option. If included then lower case sequence is not indexed.
-b	[1]Creates an index based on insilco bisulphite treatment of the reference sequence. A double index based on C->T and G->A conversion is created. Alignments using an index created with -b option will be done in bisulphite mode.
-c	Creates an index in ABI Solid Colour Space for use with NovoalignCS. -b & -c options are mutually exclusive.
-n *name*	Sets the an internal name for the reference sequence index. This is used in report headers and as the AS: field in SAM SQ record. Defaults to the *indexfile* name.
indexfile	is the filename for the indexed reference sequence generated by novoindex.
sequencefiles	a list of sequence files in fasta format to be included in the index.

Example, to generate an index file named 'celegans' for the sequence file “elegans.dna.fa”

```
novoindex celegans elegans.dna.fa
```

The index includes a copy of the reference sequence compressed to 4-bits per base. The compressed format retains ambiguous nucleotide codes which will be scored appropriately by the alignment process. This feature is especially important for use with genomes that have high numbers of scattered ambiguous codes such as Maize.

When indexing k-mers with ambiguous nucleotide codes, index entries are created for all possible combinations of non-ambiguous codes. For instance if a k-mer contains an N, then 4 index entries will be stored with ACG&T replacing the N. To control possible explosion of index entries this process is limited to two ambiguous codes per k-mer. Any k-mer with more than two ambiguous codes is not indexed.

Index Size Calculations

A normal index comprises three main tables:

1. A k-mer hash table of size 4^{k+1} bytes
2. A sequence offset table of size 4N/s bytes where N is the length of the sequences being index and s is the step size.

[1] Only available in licensed versions.

3. A compressed sequence file of size N/2 bytes.

A bisulphite mode index comprises five tables, the first two being doubled up for the CT and GA indexes.:

1. Two k-mer hash tables of size $4*3^k$ bytes
2. Two sequence offset tables of size 4N/s bytes where N is the length of the sequences being index and s is the step size.
3. A compressed sequence file of size N/2 bytes.

If lower case masking is specified any k-mer composed entirely of lower case codes will not be indexed. The lower case NA codes are still retained in the 4-bit/bp compressed sequence file.

Examples

C Elegans Genome

Genome size is 100Mbp, then using k=13, s=1 the index size is

= 250Mb+ 400Mb + 50Mb
= 700Mbytes

With k =13 and s=3 the size would be
=250Mb + 133Mb + 50Mb
=433Mbyte.

Homo Sapiens Genome

For searching the full human genome on an 8Gbyte RAM server the recommended settings are k=14, s =3. This gives a theoretical index size of:
= 1Gb+ 4Gb + 1.5Gb
= 6.5Gbytes
In practice the size is 6.0Gbytes due to N regions which are not indexed.

For searching the full human genome on an 16Gbyte RAM server the recommended settings are k=15 s =2 or k=14, s=1. The theoretical index size for k=15, s=2 is:
= 4Gb+ 8Gb + 1.5Gb
= 13.5Gbytes

Novoindex is multi-threaded and will use all available CPUs. Typical index build time for Human Genome index (k=14, s=3) on a dual core AMD Athlon CPU is approximately 3 minutes.

Novoalign & NovoalignCS

Aligns sequencing reads against an indexed set of reference sequences. Novoalign uses an iterative search algorithm to find the best alignment and any other alignments with similar score.
Some heuristics are used in calculation of alignment quality scores.

Usage:
novoalign options

Option	Description
-d *dbname*	Full pathname of indexed reference sequence from novoindex
-f *seqfile1* [*seqfile2*]	Files containing the read sequences to be aligned. File formats allowed include Solexa PRB, Sanger FASTQ, FASTA, Solexa FASTQ, Illumina FASTQ, and Illumina qseq_txt. If two files are specified then they are treated as paired end reads.
	NovoalignCS accepts ABI Solid *.csfasta files with _QV.qual quality files or .csfastq files.
-F *format*	Specifies the *format* of the read file. Normally Novoalign can detect the format of read files and this option is not required. However starting with Illumina pipeline version 1.3 the scale for quality values has been changed. If you are using the new format Illumina *_sequence.txt files you need to add the option '-F ILMFQ' to ensure correct interpretation of quality values. Other values for the -F option are:
	FA — Fasta format read files with no qualities.
	SLXFQ — Fastq format with Solexa style quality values. $10\log_{10}(P/(1-P))$ + '@'
	STDFQ — Fastq format with Sanger coding of quality values. $-10\log_{10}(Perr)$ + '!'
	ILMFQ — Fastq with Illumina coding of quality values. $-10\log_{10}(Perr)$ + '@'
	PRB — Illumina _prb.txt format.
	PRBnSEQ — Illumina _prb.txt with _seq.txt files.
	QSEQ — Illumina *_qseq.txt format files from Bustard.
	Note. For various fastq format files, even if the -F option is used Novoalign will still check the actual quality values and verify they are consistent with the -F setting.
	NovoalignCS should can detect file formats however you can still specify the format using the -F option.
	CSFASTA — ABI Solid colour space fasta format with optional _QV.qual file.
	CSFASTQ — Colour space FASTQ format as used in BFAST.
-# 99[K|M]	Sets a limit on the number of reads or pairs to process from the input files. e.g. - 10K will only align the first 10,000 reads.

Option	Description
Alignment Scoring Options:	
-t 99	Sets the threshold or highest alignment score acceptable for the best alignment. A default threshold is calculated from read length and genome size such that an alignment to a non-repeat should have a quality higher than 30.
-g 99	Sets the gap opening penalty. Default 40
-x 99	Sets the gap extend penalty. Default 15
Bisulphite Alignment Options:	
-u 99	Sets a penalty for unconverted cytosines at CHG and CHH positions as these are less likely to be methylated than CGH sites, thus biasing alignment in favour of methylated CG sites. Default is no penalty. Suggested values are 12 for vertebrate and 8 for plants. Using this option can reduce runti
-b mode	Sets Bisulphite alignment mode. Values for mode are: 4 - Aligns in 4 possible combinations of direction (forward & reverse complement) and index (CA & GT). (Default) 2 - Aligns reads in forward direction using CT index and in reverse complement using the GA index. This option is appropriate if using standard Illumina Bi-seq protocol as it preserves strand of the fragments. Bisulphite mode is not available in NovoalignCS
Quality Control and Read Filtering Options:	
-l 99	Sets the minimum number of good quality bases for a read. Default is set to $\log_4(Ng) + 5$ where Ng is the length of the indexed reference sequences. This test is based on information content of the read using Shannons Entropy.
-h 99 [99]	Sets a threshold for the homopolymer and optionally the dinucleotide repeat filters. All reads are checked to see if they are homopolymers (or dinucleotide repeats) and if so they are not aligned. Base qualities are used in calculating a homopolymer score. If the score is less than the threshold then the read is deemed to be a homopolymer. Default value is 20. Setting a negative value disables homopolymer filtering. The second threshold is used for filtering dinucleotide repeats. This can useful for improving performance when aligning against genomes with high dinucleotide repeats. For paired end reads both reads would have to be homopolymers or dinucleotide repeats for alignment of the reads to be skipped. Reads that are over threshold are reported with a status of 'QC' NovoalignCS only accepts a single threshold which applies to homopolymers in colour space.

Option	Description
-H	Hard clips trailing bases with quality <=2 from reads before aligning them. Hard clipping is applied before the polyclonal filter so that if after trimming theread is high quality it may pass the polyclonal filter.
-p 99,99 [0.9,99]	Sets thresholds for polyclonal filter. This filter is designed to remove reads that may come from polyclonal clusters or beads. Please refer to paper: ***Filtering error from SOLiD Output, Ariella Sasson and Todd P. Michael.*** The first pair of values (n,t) sets the number of bases and threshold for the first 20 base pairs of each read. If there are n or more bases with phred quality below t then the read is flagged as polyclonal and will not be aligned. The alignment status is 'QC'. The second pair applies to the entire read rather than just the first 20bp and is specified as fraction of bases in the read below the given quality. Setting **-p -1** disables the filter. Default for Novoalign is **off**. Default for NovoalignCS is -p 7,10 0.3,10. i.e 10 of first 20bp below Q10 or 30% of all bases below Q10 will be flagged as a low quality read. Low quality reads may still be used in paired end mode if the mate is not low quality.
Read Preprocessing Options:	
-a [*adapter1*] [*adapter2*] [2]	Strips a 3' adapter sequence from read prior to alignment. Default adapter sequence is 'Gex Adapter 2' , "TCGTATGCCGTCTTCTGCTTG". e.g. novoalign -a TCGTATGCCGTCTTCTGCTTG This is usually used when sequencing small RNA. With paired end reads it can be used to strip adapter off fragments that are shorter than the read length. In this case you can specify two adapter sequences, the first for read 1 of each pair and the second for read 2. Default adapter sequences for paired end reads are: Read1: AGATCGGAAGAGCGGTTCAGCAGGAATGCCGAG Read2: AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTA For Illumina mate pair reads, when both short and long fragment lengths have been entered with the -i option, the two reads from a short fragment will be trimmed to remove the adapter and the overlap. This allows proper identification of reads that overlap the circularisation junction. NovoalignCS does not have adapter stripping functions.

[2] Adapter stripping of paired end reads is only available in Licensed versions of Novoalign. Unlicensed versions can strip adapter from single end reads for miRNA projects.

Option	Description
-n 99	Truncates reads to the specified length before alignment. Useful for truncating reads when 3' quality is really bad..
-s [9]	Turns on read trimming for single end reads only. Reads that fail to align will be progressively shortened by specified amount (defaults to 2) until they either align or length reduces to less that the length set by the -l option, in which case the shortened read fails quality control checks. This option only applies to single end reads. Use at your own discretion. e.g. To trim reads in steps of 2 bases... novoalign -s To trim reads in steps of 5 bases... novoalign -s5
-5 sequence [3]	Strips 5' primer sequences from reads before aligning. Default is not to strip 5' sequences. This option is useful where sample preparation protocol involved an additional PCR step with non-Solexa primers that may still be present on the 5' ends of reads. This option is similar to the -a except that it acts on the 5' end of reads. It will strip partial primer sequences. NovoalignCS does not support this function.
Reporting Options:	
-o [*format* | *option*]	Specifies output report format and options.

[3] 5' PCR primer stripping is only available in licensed copies of Novoalign

Option	Description
-o *format* [*readgroup*]	Specifies the report format. **Native, Pairwise, SAM,** or **Extended.** Default is Native. eg. novoalign -o Pairwise or , novoalign -o SAM When SAM format is specified a readgroup record (@RG) can follow the -o SAM option. Note that the @RG record should be tab delimited and in bash shell you can do this using $'...\t...' syntax. e.g. novoalign -oSAM $'@RG\tID:*readgroup*\tPU:*platform-unit*\tLB:*library*' -d ... Novoalign will also convert any '\t' in the option to tabs so you can also use :- novoalign -oSAM “@RG\tID:*readgroup”* -d … The ID, PU & LB values, if present, will be used as tags on the alignment records as per SAM specifications
-o Sync	In multi-threaded mode ensures that the output report is synchronous with the read file. This may increase memory usage.
-o SoftClip	With this option alignments in SAM format will be soft clipped back to the best local alignment. On by default from V2.06.10. This option helps reduce SNP and micro indel noise from the ends of alignments. The option can also be used with Native format to limit SNP calls to those within the best local alignment.
-o FullNW	Turns off softclipping so all bases in the read are (other than adpater trimming) are reported as matches or indels. This may report inserts at the ends of reads that align across the ends of referece sequences or across structural breaks in the genome. NovoalignCS does not suport this option.
-Q 99	Specifies a lower limit on alignment quality for reporting. Default is 0.
-R 99	Specifies a score difference between first two alignments for reporting repeats. If the difference is less than this then the read is treated as aligning to a repeat and '-r method' applies. Default is 5.

<table>
<tr><th>Option</th><th colspan="2">Description</th></tr>
<tr><td>-r method [limit]</td><td colspan="2">Sets the rules for handling of reads with multiple alignment locations. Values are:-</td></tr>
<tr><td></td><td>None</td><td>No alignments will be reported. The read will be reported as a type R with no alignment locations.</td></tr>
<tr><td></td><td>Random</td><td>A single alignment location is randomly chosen from amongst all the alignment results.</td></tr>
<tr><td></td><td>All</td><td>All alignment locations are reported. The 'All' method can optionally specify a limit for the number of lines reported. e.g. '-r A 10' will report at most 10 randomly selected alignments.</td></tr>
<tr><td></td><td>Exhaustive</td><td>Reports all alignments with a P(R|Ai) score less than or equal to the threshold. The 'Exhaustive' method requires that a limit for the number of lines reported. e.g. '-r E 10' will report at most 10 randomly selected alignments per read. This is to avoid situations where high copy number repeats result in reporting millions of alignments for a read.</td></tr>
<tr><td></td><td>0.99</td><td>Posterior probability threshold. All alignments with posterior probability greater than the set value will be reported.</td></tr>
<tr><td>-e 999</td><td colspan="2">Sets a limit on number of alignments recorded for a read during the iterative search process. The limit applies to the number of alignments with score equal to the best alignment. When limit is reached no further alignments are recorded and the search for this read is stopped. Default is 1000 in default report mode, in other report modes the default is no limit.
This limit is designed to reduce CPU utilisation for reads that align to high copy number repeats and that would be reported with an 'R' status.</td></tr>
<tr><td>-q 9</td><td colspan="2">Sets number of decimal places for quality score. Default zero.
Example: -q2 will print quality scores with 2 decimal places.</td></tr>
<tr><td><b><i>Paired End Options:</i></b></td><td colspan="2"></td></tr>
<tr><td>-i [MP|PE|++|+-|-+] 99[|-|,]99
-i MP 99[-|,]99[-|,] 99,99</td><td colspan="2">Sets fragment orientation and approximate fragment length for proper pairs.</td></tr>
<tr><td></td><td>MP</td><td>Sets for Illumina or ABI mate pair orientation</td></tr>
<tr><td></td><td>PE</td><td>Sets paired end orientation, +-.</td></tr>
<tr><td></td><td>+-</td><td>Sets orientation where two reads of a pair are on opposite strands and facing each other. Equivalent to setting PE.</td></tr>
<tr><td></td><td>-+</td><td>Sets orientation where two reads of a pair are on</td></tr>
</table>

Option	Description

	opposite strands and facing away from each other. This is normal mode for Illumina mate pairs.
++	Sets orientation where two reads of a pair are on same strand in ABI SOLiD format. <----F3---- <---R3---- or ----R3----> ----F3---> This mode can also be used for 454 mate pair reads.

Expected fragment lengths sizes can be set as a mean and standard deviation or as a range of lengths using '-' as delimiter.
Examples:

-i 250 50	Defaults to paired end Illumina or Mate Pair ABI with 250bp insert and 50bp standard deviation
-i PE 250,50	Uses paired end orientation with 250bp insert and 50bp standard deviation
-i MP 2000,200	Uses mate pair orientation with 2000bp insert and 200bp standard deviation
-i +- 50-300	Sets +- (paired end) orientation with proper pair fragments ranging in length from 50 to 300bp.

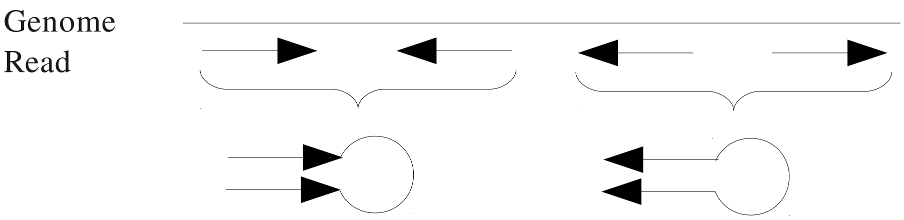
When a range of fragment lengths is specified Novoalign will not apply fragment length penalties and this may impact ability to resolve alignments near tandem and other local repeats.

The second form allows both a long insert length and a short insert length to be set for Illumina mate-pair reads. If a short insert length is specified then Novoalign will map proper fragments of either type. It will also handle the case where the circularisation junction is within one of the reads, reporting the alignment to the longer portion of the read.
Example:

-i MP 2500,600 250,50	Specifies mixed mate pair and paired end reads.

Proper setting of orientation is important. If in doubt about mean fragment length and standard deviation err on the high side.
Default for Novoalign is paired end reads with mean length of 250bp and standard deviation of 30bp.
Novoalign tracks the actual length of fragments with high quality alignments to both reads of the pair and dynamically adjusts fragment length penalties to suit the actual fragments.
For NovoalifgnCS can set paired end mode using -i PE 200,50

Option	Description
	Changing between Paired end and Mate pair mode changes the expected orientation of the alignments in a proper pair. For paired end reads the alignents are on opposite strands and face each other --------> <------- For Illumina mate pairs the alignments face outwards <-------- ------->LINEBREAK NovoalignCS default is mate pairs with mean length of 2500bp and standard deviation of 500bp. For ABI Colour space mate pairs the alignments are on same strand <----F3---- <---R3---- or ----R3----> ----F3--->
-v 99	Sets the structural variation penalty for chimeric fragments. This form uses a single penalty for all pairs that do not fit the fragment length distribution. Default penalty is 70. If Psv is the probability of a structural variation (that might result in a chimeric fragment) in the genome being sequenced vs the reference genome then the SV penalty is $-10\log_{10}(Psv)$. Individual alignments will be reported if their combined score less the structural variation penalty is better than the best pair.
-v 99 99	Sets the structural variation penalties for chimeric fragments. In this form the first penalty applies to chimera where the alignments for the two reads of a pair lie in the same reference sequence. The second penalty applies to chimera that cross reference sequences.
-v 99 99 99 regex	Sets the structural variation penalties for chimeric fragments. The three penalties are for: 1. Penalty for SVs within a group of sequences as defined by the regular expression. 2. Penalty for SVs within a single sequence 3. Penalty for SVs across different sequences and groups. regex defines a regular expression applied to headers of indexed sequences. The regular expression should define one field that selects a group name field from the sequence header.

Option	Description
miRNA mode:	**Novoalign Only**
-m [*99*]	Sets miRNA mode. In this mode each read is given an additional score based on the Needleman-Wunsch alignment of the read to the opposite strand. Precursor miRNA which form hairpin structures should get a better score for the adjacent opposing strand alignment.  *Precursor miRNA forms a hairpin structure which means that there should be adjacent forward and reverse complement alignments to the miRNA. Novoalign reports an additional score for the best nearby alignment on the opposite strand to the primary alignment.* The optional parameter [*99*] controls the length of the sequence region scanned for the reverse complement alignment and is the maximum distance (gap) between the two alignments of the hairpin structure. Default is 100bp. (in earlier versions of Novoalign this was fixed at 50bp) In miRNA mode the repeat reporting is defaulted to 'All'. The miRNA mode does not turn on adapter filtering. This allows use with reads that have already had the adapter stripped from them. Not currently available in NovoalignCS
Multithreading[4]:	
-c 99	Sets the number of threads to be used. On licensed versions it defaults to the number of CPU cores. On free version the option is disabled.
Quality Calibration[5]:	
-k [infile]	Enables quality calibration. The quality calibration data (mismatch counts) are either read from the named file or accumulated from actual alignments. Default is no calibration. Note. Quality calibration does not work with reads in prb format.
-K [file]	Accumulates mismatch counts for quality calibration by position in the read and called base quality. Mismatch counts are written to the named file after all reads are processed. When used with -k option the mismatch counts include any read from the input quality calibration file.

[4] Licensed versions only.

[5] Requires a license

Examples

novoalign -f s_1_sequence.txt -dcelegans	Aligns the reads in file s_1_sequence.txt against the indexed genom of C.Elegans.
novoalign -Q90 -f s_1_sequence.txt -d celegans	As above but limits results to alignmnets with a quality of 90 or better.
novoalign -a -m -f s_1_0001_prb.txt -d hg36	Aligns a set of miRNA reads against the human genome. Adapter sequences are stripped from the reads and an additional miRNA hairpin score is given for each alignment. Reports multiple alignments per read if they exist.
novoalign -R 30 -rAll -f s_1_sequence.txt -d hg36	Aligns a set of reads against indexed human genome, reporting multiple alignments per read. Any read with a score within 30 points of the best alignment will be reported.
novoalign -Q90 -i 170 30 -f read1/s_1_sequence.txt read2/s_1_sequence.txt -d hg36	As above but applies a quality limit of 90.
novoalign -f sim_l.fastq sim_r.fastq -dchrX	Aligns the paired files 'sim_l.fastq' and 'sim_r.fastq' against an index chrX.

Description

Base Qualities and Alignment Scores

Novoalign aligns reads against a reference genome using qualities and ambiguous nucleotide codes. The initial alignment process finds alignment locations in the indexed sequence that are possible sources of the read sequence. The alignment locations are scored using the Needleman-Wunsch algorithm with affine gap penalties and with position specific scoring derived from the read base qualities and any ambiguous codes in the reference sequence. User defined affine gap penalties are used for scoring insert/deletes.

Novoalign uses Needleman-Wunsch alignments with affine gap penalties, the gap opening penalty should be set to -10log$_{10}$(Pgap) – G$_{extend}$ where Pgap is the probability of an insertion deletion mutation vs the reference genome and G$_{extend}$ is the gap extension penalty. Likewise the gap extend penalty can be set to -10log10(Pgap2/Pgap1) where Pgap1 is the probability of a single base indel and Pgap2 is the probability of a 2 base insert/delete mutation. The default gap penalties were derived from the frequency of short insert/deletes in human genome resequencing projects.

Base quality values are used to calculate base penalties for the Needleman-Wunsch algorithm. The base qualities are converted to base probabilities and then to score penalties.

Colour space alignments are implemented using variation of dynamic programming from “Rumble SM, Lacroute P, Dalca AV, Fiume M, Sidow A, Brudno M: **SHRiMP: Accurate Mapping of Short Color-space Reads.** *PLoS Comput Biol* 2009, **5:**e1000386. PubMed Abstract | Publisher Full Text | PubMed Central Full Text “

PRB Quality to Score Conversion

The prb file has quality score Q(b,i) for each base, b, at each position, i, in the read. The quality value is converted to a probability, Pr(b,i) and then to a penalty P(b, i).

$$Pr(b,i) = \frac{10^{\frac{Q(b,i)}{10}}}{(1+10^{\frac{Q(b,i)}{10}})}$$

$$P(b,i) = -10 \times \log_{10}(Pr(b,i))$$

Single Base Quality to Score Conversion

Sanger FASTQ, Solexa FASTQ, Colour Space readsand other read formats such as Phred have a called base S(i) or colour and single quality score Q(i) at each position, i, in the read. The quality value is converted to a probability, Pr(i) and then to a penalty P(S(i), i).

Solexa

$$Pr(i) = \frac{10^{\frac{Q(i)}{10}}}{(1+10^{\frac{Q(i)}{10}})}$$

Fastq or Phred

$$Pr(i) = 1 - 10^{-\frac{Q(i)}{10}}$$

Alignment Penalty

$$P(S(i),i) = -10 \times \log_{10}(Pr(i))$$

$$P(b \in (\{A,C,G,T\} \setminus S(i)),i) = -10 \times \log_{10}((1-Pr(i)) \div 3)$$

Base Penalty Limit

For nucelotide alignments the penalties calculated above are further limited to a maximum of 30 at any base position. For colour space alignments no limit is applied to the penalty for a colour error and a default penalty of 30 is used for SNPs.

Alignment Score and Threshold

The alignment score is $-10\log_{10}(P(R|Ai))$ where P(R | Ai) is the probability of the read sequence given the alignment location i.

A threshold of 75 would allow for alignment of reads with two mismatches at high quality base positions plus one or two mismatches at low quality positions or to ambiguous characters in the reference sequence.

If a threshold is not specified then Novoalign will calculate a threshold for each read such that an alignment to a non-repetitive sequence will have an alignment quality of at least 20. I.e. The iterative process of finding an alignment will terminate before finding a low quality chance alignment. Alignments to repetitive sequences may still have qualities less than 20.

Posterior Alignment Probabilities and Quality Scores

The posterior alignment probability calculation includes all the alignments found; the probability that the read came from a repeat masked region or from any regions coded in the reference genome as N's; and an allowance for a chance hit above the threshold based on the mutual information content of the read and the genome.
A posterior alignment probability, P(Ai| R, G) is calculated as:

$$P(A_i|R,G) = \frac{P(R|A_i,G)}{P(R|N,G) + \sum_i P(R|A_i,G)}$$

where P(R|N,G) is the probability of finding the read by chance in any masked reference sequence or any region of the reference sequence coded as N's, and where Σ_i is the sum over all the alignments found plus a factor for chance alignments calculated using the usable read and genome lengths.
The P(R|N,G) term allows for the fact that a fragment could have been sourced from portions of the genome that are not represented in the reference sequence. For instance in Human genome build 36 there is approximately 7% of sequence represented by large blocks of N's.

A quality score is calculated as $-10\log_{10}(1 - P(Ai| R, G))$, where P(Ai|R, G) is the probability of the alignment given the read and the genome.

Adapter Stripping

Single End Reads - miRNA

Adapter stripping does an ungapped global alignment of the adapter against the read and then trims the read from the start of the optimum alignment.

A few details:

1. The read and base qualities are first converted to a weight matrix where each base will score max(30, -10log(P)) where P is probability of the base. This results in a match scoring 0 and a

mismatch at high quality base position scoring 30

2. During adapter stripping we subtract 7 from the weights so at a high quality base position a match scores -7 and a mismatch 23.
3. If the optimum alignment scores <= -7 it is stripped.
4. There are no penalties for unmatched letters at the beginning of the read or at the end of the adapter.

Paired End Reads – Short Fragments

If a DNA fragments is shorter than the read length then both reads of the pair will have extended into adapter or primer sequence and unless stripped off will be used in alignment.

If there are only a few bases of adapter the read may still align but with some mismatches or indels in the adapter portion of the alignment. This contributes to SNP noise and reduced consensus quality.

When there is more than a few bases of adapter the read is unlikely to align which isn't a problem except that there has been an attempt to align it that will have tried to align with possible 8 mismatches and up to 7 indels. This attempt to align the read with so many mismatches can consume considerable CPU time so it's desirable to identify these reads before aligning them.

Novoalign identifies short fragments by aligning the two reads of a pair against each other to detect overlap and adapter sequence. If overlap is detected then any adapter is trimmed from the two reads.

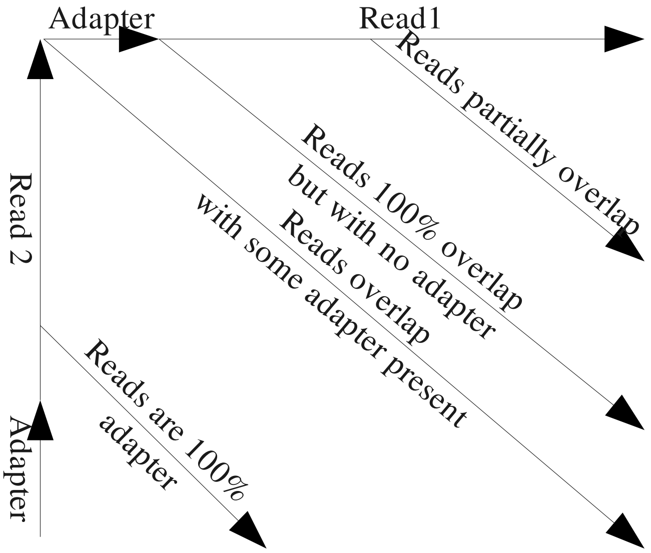

Adapter
Read1
Read 2
Adapter
Reads partially overlap
Reads 100% overlap but with no adapter
Reads overlap with some adapter present
Reads are 100% adapter

Illustration 1: Dynamic Programming alignment of two paired-end reads with insilco pre-pended the first 12bp of the adapter sequence. High scoring diagonal identifies the amount of overlap and adapter sequence present in the read. False positive rate is low as reads must be complementary and align to the adapter to get a good score

Read Quality

Reads with too many low quality base positions will not be aligned. This is controlled by the -l options and effectively sets the minimum length, or minimum number of high quality base positions in order for an alignment to be attempted. The read length calculation uses base qualities to calculate the information content of the read.

Homopolymer reads are also deemed low quality and not aligned. These are fairly frequent in real data and are possibly the result of dust on slides.

Reads with Multiple Alignments

There are times that reads will align to multiple locations with very similar alignment scores. Situations where this might occur are reads originating from repeats and the alignment of very short reads such as small RNA.

Depending on the users project and objectives reads and alignments may be or not be of interest.

Every read will have multiple alignment locations however the alignment score could be very different, so for detection of repeats novo programs use the difference in score between the best alignment and the rest of the alignments. This score difference is set by the '-R99' option and defaults to 5 which corresponds to the best alignment being approximately 3 times more probable than the next best alignment. For example, two alignments with probabilities 0.7 (score 1) and 0.3 (score = 5) would be considered as multiple alignments to the read. Two alignments with probabilities 0.8 (Score 0) and 0.2 (score 7) would be treated as a unique alignment to the location with the higher probability.

Having identified a read as having multiple alignment locations we then have several options for reporting.

Option	Description
None	No alignments will be reported. The read will be reported as a status R with a count of the number of alignments. No alignment locations will be reported.
Random	A single alignment location is randomly chosen from amongst the alignment results. The choice is made using posterior alignment probabilities.
All	All alignment locations are reported. Note, that this is all alignments with a score within 5 points of the best alignment unless you use the -R99 option to extend the range.
Exhaustive	This option bypasses the iterative alignment process and the normal repeat alignment detection. It finds all alignments with a score no worse than the threshold (-t 99 option) and reports all the locations.
0.99	Sets a posterior probability threshold. Any alignment with a posterior probability, P(Ail R, G) greater than this value will be reported. Eaxmple: -r 0.01 will report all alignments with a probability greater then 0.01.

Sequence file formats

Read files are introduced using the -f options. Novoalign examines the file name and the first few lines of each file to determine the file format.
Licensed versions of Novoalign will also process read files compressed with gzip.

Format	File Names	Description and detection method
FASTA	*.fa *.fna *.fasta	Standard FASTA format input file can be used. This file type is recognised by the name matching *.fa, *.fna , or *.fasta or by the first line starting with a '>' character. e.g. >sequence_0 GATGTCACTCAGTATGAGAAAGAGGCAGGTTCTGGG >sequence_1 ACACGCAGCGCCGCGCATGCTTGCGCCGCCACTCCA >sequence_2 ACCTGCGCTCTGCCCTGAAACCACTGTTGGCTTGAG Example: novoalign -f reads.fa -d celegans
.FASTA & Quality	as above with *.qual	Fasta file are detected and then the folder is checked for a quality file. If novoalign detects a fasta format read file it looks for a matching *.qual file in the same folder. If found then it will be used for base qualities. >sequence_0 40 40 40 40 40 40 40 40 40 40 40 40 14 40 40 40 40 40 40 40 40 40 25 40 40 40 40 40 40 5 40 8 9 21 40 4 >sequence_1 40 19 7 22 4 40 8 40 40 40 9 40 28 40 40 40 17 31 11 40 32 24 4 9 14 10 36 16 40 9 2 8 6 16 3 3
Sanger FASTQ	*.fastq	Sanger format FASTQ files are recognised by the file name matching *.fastq. Quality scores are from ASCII code – 33. For non-standard filenames this format is detected by an '@' character starting the first line and by a test on the quality codes of the first read. Sanger fastq files are automatically detected as the ASCII coded qualities are lower than for a Solexa format FASTQ file. Example: novoalign -f reads.fastq -d celegans

Solexa FASTQ and Illumina FASTQ	*_sequence.txt	Files produced by Illumina pipeline with Solexa variant of the FASTQ format. Solexa quality scores are ASCII letter code – 64; See Gerald documentation for a full description. These files are named like s_*lane*_sequence.txt and recognised by matching the file name against s_*_sequence.txt. For non-standard filenames this format is detected by an '@' character starting the first line and by a test on the quality codes of the first read. Solexa fastq files are automatically detected as the ASCII coded qualities are higher than for a Sanger format FASTQ file. Starting from Version 1.3 of the Illumina Casava Pipeline the coding of quality values was changed to the Phred scale. If you are using Pipeline 1.3 you may need to add the option -F ILMFQ. This option will treat quality codes as being coded as $-10\log_{10}(Perr)$ + '@'. The old Solexa format is the default for _sequence.txt files and interprets quality values according to formula $-10\log_{10}(P/(1-P))$ + '@'
Solexa PRB	*_prb.txt	Illumina/Solexa prb file from the base calling program Bustard. This file has quality values (probabilities) for each of the 4 bases at each position in the read. This format is recognised by file name matching s_*_prb.txt. For non-standard filenames a prb format file is identified as having a first line that consists only of digits, minus sign and whitespace.
Solexa PRB & SEQ	as above with *_seq.txt	If a prb file is detected by filename test then we look for the corresponding seq file produced by Bustard base caller. This file contains lane, tile and X,Y coordinates of the read which are then used as the read sequence identifier. It is recognised by file name s_*_seq.txt.
Illumina QSEQ	*_qseq.txt	Illumina qseq file format. e.g. SOLEXA 90403 4 1 23 1566 0 1 ACCGCTCTCGTGCTCGTCGCTGCGTTGAGGCTTGCG `aaaaa```aZa^`]a``a``a]a^`a\Y^`^^]V` 1 The 7 fields before the read sequence are converted to a header by prefixing with a '>' and substituting tabs with underscores '_'. The last field which is Illumina quality flag is currently ignored.

ABI Solid CSFASTA	*.csfasta *_QV.qual	*.csfasta >2_14_26_F3 T0112131222002211230321112210212101313322222101 >2_14_192_F3 T1100212211003100301200220322221113210221122223 *_QV.qual >2_14_26_F3 24 24 22 27 23 10 13 13 20 19 19 18 24 20 22 12 14 5 20 17 14 20 18 17 19 11 21 19 13 13 12 25 9 19 19 6 5 12 20 13 11 8 12 7 14 >2_14_192_F3 14 19 21 13 24 17 18 18 25 21 8 12 21 8 7 11 14 7 19 23 11 24 7 11 29 12 28 17 7 19 7 11 5 11 5 14 13 9 24 8 7 20 0 8 9
CSFASTQ	*.csfastq	Colour Space FASTQ with primer base quality @SRR015241.1 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_34_F3 length=50 T32322133300002330031001022230020232002203222030231 +SRR015241.1 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_34_F3 length=50 !21(()+%'+%40*.%%**)&%&*&%%%&%%%%%%%%%%%%%%(+%%%%' @SRR015241.2 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_269_F3 length=50 T01212120333223322020022322232232232222022232033230 +SRR015241.2 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_269_F3 length=50 !,*+*()+*(%'+)%%%&%+&%%'%%%%%%%%%%%%%%%%%%%%'+%%%%%
BFASTQ	*.csfastq	Colour Space FASTQ without primer base quality @SRR015241.1 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_34_F3 length=50 T32322133300002330031001022230020232002203222030231 +SRR015241.1 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_34_F3 length=50 21(()+%'+%40*.%%**)&%&*&%%%&%%%%%%%%%%%%%%(+%%%%' @SRR015241.2 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_269_F3 length=50 T01212120333223322020022322232232232222022232033230 +SRR015241.2 CLARA_20071207_2_CelmonAmp7797_16bit_26_88_269_F3 length=50 ,*+*()+*(%'+)%%%&%+&%%'%%%%%%%%%%%%%%%%%%%%'+%%%%%

Output Formats

Three output formats are provided.

1. Native
2. Extended Native
3. Pairwise
4. SAM

Native Report Format

The native format is designed to be compact, giving essential information necessary for downstream processing. This is default report format.

```
# novoalign (1.0) - short read aligner with qualities.
# (C) 2008 NovoCraft
# Licensed for evaluation and educational Use Only
# novoalign -d ssuis -f ../../s_8_0100/s_8_0100.fa -q ../../s_8_0100/s_8_0100.qual
# Index Build Version: 1.0
# Hash length: 11
# Step size: 1
# Interpreting input files as FASTA with Phred quality file.
>I8_100_293_551  S   CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCACC IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII NM
>I8_100_880_947  S   TTATTATCTTTATTGACGTACCTCTAGAAGACCCAA IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII;>1 U
    0   150 >Ssuis   420732   R    .    .
>I8_100_975_684  S   AGTAGACACCTGGTGAACGAACCAACTGAGAAACGA IIIIIIIIIIIIIIIIIIIIIIIIII-EII)IIIIG U
```

```
    1   150 >Ssuis  111343  R   .   .
>I8_100_874_727  S   GTGAAAGCCAGCGTCTTTAGGCGCTGGGTGGTGGTG IIIIIIIIIIIIIIIIIIIIIIIIIII%IIIII,59 R
    4
>I8_100_244_639  S   AACATAATTAGACAGAATATAAGATATGACTAATTC IIIIIIIIIIIIIIIIIIIIIIIIIIIIIII9H2)I U
    1   150 >Ssuis  1364843 R   .   .
>I8_100_492_8    S   ANNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN I"""""""""""""""""""""""""""""""""""  QC
>I8_100_515_741  S   GGAAATCACGGAGCAGGAGTTTCGTGAGCTTCGCCG IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII/I U
    49  141 >Ssuis  429042  F   .   .   .   35G>C 36A>G
>I8_100_510_804  S   AACCGACAGTTGCTTCGTCTACAATCACAATACCCG IIIIIIIIIIIIIIIIC9II='II$II&I,&H89+0 U
    54  117 >Ssuis  1499130 R   .   .   .   4C>G 9T>G 15T>G
>I8_100_188_601  S   ACTACGTTCACAGAAAATCTAGCCTTTGTACTAGAC IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII/II% U
    9   150 >Ssuis  145620  R   .   .   .   1T>G
>I8_100_63_601   S   AGCGGCAGGGCTTGTTCCAGCTAAGGCTCCGATTTT IIIIIIIIIIIIII'IIIIIIIIHII%A%,I&IIII U
    114 57  >Ssuis  1997459 R   .   .   .   8T>G 9T>A 11T>C 22T>A 27T>C
>I8_100_331_271  S   GGATTATGTGAAACAACATGCTGATGCACCGCTTAA IIIIIIIIIIIIIIIIIIIIIIIIIIIIIII,II@& U
    18  150 >Ssuis  1883394 R   .   .   .   5T>G
>I8_100_408_934  S   ATGATATTAGGTCCTATCTTACTTTTCTCAACCAAC IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII= U
    0   150 >Ssuis  580585  R   .   .
>I8_100_269_390  S   GTGTTCCCAAACCTGCTGCAGGGATAACGGCTTTTT IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII8 U
    28  150 >Ssuis  1853977 R   .   .   .   1G>A
...
>I8_100_768_102  S   AAACACATGGTGTTATNAAACTCGCGACGTAGTCAT )%)II"&I$,I9I)")"I'*7IGI9"2'IE$$I&%$  NM
>I8_100_582_231  S   TAAGCAAAAAACATAATTCCAGGATATGCAACCAGT '%%"&#&)$%%$$"$$##%'%$$%##"$)%#%$$#"  QC
>I8_100_240_200  S   AATAAAGCCTAAACAATGGACAAACAAACTACACAC :%$%/$#%&$&$$$&$"#"$&%#$%'%%%#$$$$'&  QC
#     Read Sequences:    10959
#            Aligned:     9699
#   Unique Alignment:     9442
#   Gapped Alignment:       92
#     Quality Filter:      273
# Homopolymer Filter:        5
#       Elapsed Time: 19,046s
# Done.
```

Normally a read is printed on one line with a series of tab delimited fields. The fields are :-

Field	Description
Read Header	The fasta or fastq header of the read sequence.
S, L or R	S indicate this is an alignment for single ended read. For paired end reads L indicates the read is from the first file. R indicates the read is from the second file.
Read Sequence	The read sequence.
Base Qualities	Standard (Sanger) Fastq format base qualities, empty for fasta input unless using quality calibration. If quality calibration is used these are calibrated qualities.
Nucleotide Sequence	For NovoalignCS only, this field is the decoded nucleotide sequence.
Aligned Base Qualities	For NovoalignCS only, this field is the base qualities for the decoded nucleotide sequences. This follows the BFAST & MAQ 0.7.1 convention from BFAST Wiki (http://sourceforge.net/apps/mediawiki/bfast/index.php?title=Mapping_Quality). For ABI SOLiD data, base qualities are calculated using the following formula: • If the base is the last decoded base (last base sequenced), then the base quality is equal to the colour quality of the last colour. • Else if the two colours observing the base are not called sequencing errors, then the base quality is the sum of the two colour qualities.

- Else if exactly one out of the two colours observing the base are called sequencing errors, then the base quality is calculated from the difference between the colour penalties of the non-sequencing-error-colour and the sequencing-error-colour.
- Else the base quality is zero.

Note that colour qualities are converted to alignment penalties before alignment and then alignment penalties are converted back to qualities.

Colour Quality	**Colour Error Penalty**
0	0
1	0
2	2
3	5
4	7
5	8
6	10
7	11
8	12
9	13
10	14
11	15
12	16
>=13	Quality + 5

5' trim count Count of bp trimmed from the 5' end of a read. Refer -5 command line option. Only present in Extended Native format

3' trim count Count of bp trimmed from the 3' end of a read. Refer -a & -s command line options. Only present in Extended Native format

Current versions of NovoalignCS do not support read trimming.

Status

Status	**Meaning**
U	A single alignment with this score was found.
R	Multiple alignments with similar score were found.
QC	The read was not aligned as it bases qualities were too low or it was a homopolymer read.
NM	No alignment was found.
QL	An alignment was found but it was below the quality threshold.

Alignment Score This is the Phred format alignment score -10log$_{10}$(P(R|Ai)).
For status of 'R' and when not report alignment locations for repeats, this field becomes the number of alignments to the read.

For paired end the alignment score includes the fragment length penalty.

Alignment Quality
This is the Phred format alignment quality score $-10\log_{10}(1 - P(Ai|R, G))$ using Sanger fastq coding method.

Proper pair flag
A value of 1 indicates that the read pair was aligned as a proper pair. Only present in extended native format.

miRNA score
Alignment score for adjacent opposite strand alignment. Optional, only included in miRNA mode.

Aligned Sequence
The fasta header of the aligned sequence. This is truncated at first space.

Aligned Offset
The 1-based position of the alignment in the sequence.

Strand
F/R Indicator of alignment direction.

Pair Sequence
The fasta header of the sequence the reads pair was aligned to. For single ended reads, or pairs where both ends aligned to the same sequence, this field is set to '.'.
If a paired alignment that fits the fragment length distribution is not found and we are reporting two individual alignments for the pair then the pair alignment location is only reported if both alignments have an alignment quality > 10.

Pair Offset
The 1-based position of the alignment to the pair of this read. For single ended reads this field is a '.'.

Pair Strand
F/R Indicator of alignment direction of the pair of this read. '.' for single ended reads.

Mismatches
A list of base indels, mismatches and bases inserted or deleted. Format is '*offset*''*refbase*'>'*readbase*' where the offset is 1 based position of difference relative to the 'Aligned Offset'.

Note. Offset of mismatches are relative to the alignment location. They are not the location of the mismatches in the read. This distinction is important when the alignment contains indels and/or is soft clipped back to the best local alignment.

Inserts are in format '*offset*'+'*insertedbases*' and deletes in format '*offset*'-'*refbase*'
The mismatch list is space delimited.
A mismatch is only reported if the probability of the base is less than 0.16. For fastq files this corresponds to a Perr ≃ 0.5
When using soft clipping the number of bases soft clipped from the 5' (as aligned) end of the alignment is reported using format 0x'*n*', and for 3' end as '*offset*'x'*n*' where n is the number of bases soft clipped.

Pairwise Report Format

Pairwise format has some similarity to Blast and is designed to be easily read. To use this report format add the option -oPairwise to the command line.

```
Query=@ILunknown_unknown_8_100_35_698
Length=36
ALIGNMENTS
>Streptococcus_suis
Length=2007491

 Score=0, Quality=150
 Strand=Minus/Plus

Query         36 ATTTTATACTCATATTTTTATATTGTCAATCATATA  1
                 ||||||||||||||||||||||||||||||||||||
Sbjct    1945571 ATTTTATACTCATATTTTTATATTGTCAATCATATA  1945606

Query=@ILunknown_unknown_8_100_293_551
Length=36
No significant similarity found.

Query=@ILunknown_unknown_8_100_605_15
Length=36
ALIGNMENTS
>Streptococcus_suis
Length=2007491

 Score=84, Quality=21
 Strand=Plus/Plus

Query          1 TATGNAGNNAANANATTCGATTCNNNTNTNTNTNNN  36
                 |||| ||  || | |||||||||   | | | |
Sbjct      11456 TATGTAGCTAATAAATTCGATTCTAATTTTTATCAA  11491

Query=@ILunknown_unknown_8_100_874_727
Length=36
 REPEAT, 4 ALIGNMENTS
```

SAM Report Format

SAM report format is for use with SAMtools, just add the option -oSAM to the command line.

The report format is documented as part of SAM/BAM specification at http://samtools.sourceforge.net/

Novoalign adds two custom tag fields for the Novoalign status and count of multiple alignments.

Tag	Type	Description
ZS	Z	Novoalign alignment status. Not present for unique alignments. **Status** – **Meaning** R – Multiple alignments with similar score were found. QC – The read was not aligned as it bases qualities were too low or it was a homopolymer read. NM – No alignment was found.

QL	An alignment was found but it was below the quality threshold.

ZN i Number of alignments to read. Only present if there was more than one alignment.

ZO Z Indictaes long or short insert fragment for mate pair alignments when short insert has been enabled.

Value	Meaning
'+-'	Indicates pair was aligned as a short insert fragment.
'-+'	Pair was aligned as a long insert fragment.

This tag is only present for Illumina mate pairs when a short fragment length size has been specified with the -i option and reads are aligned as a proper pair .

Example	Explanation
ZS:Z:R ZN:i:2	Indicates a read aligned to two locations.
ZS:Z:NM	No alignment was found.
ZS:Z:QC	The read failed quality checks and was not aligned.

When using SAM report format the run headers and statistics normally output as part of Native format reports are now written to stderr.

```
# novoalign -oSAM -d ssuis -f ../../s_8_0100/s_8_0100.fa -q ../../s_8_0100/s_8_0100.qual
# Index Build Version: 1.0
# Hash length: 11
# Step size: 1
# Interpreting input files as FASTA with Phred quality file.
#     Read Sequences:    10959
#            Aligned:     9699
#   Unique Alignment:     9442
#   Gapped Alignment:       92
#     Quality Filter:      273
# Homopolymer Filter:        5
#       Elapsed Time: 19,046s
# Done.
```

Paired End Mode

Scoring

Novoalign aligns paired reads against a reference genome using qualities and ambiguous nucleotide codes. The scoring system is based on Phred quality scores and the score for a paired alignment is $-10\log_{10}(P(F \mid Ai))$ where $P(F \mid Ai)$ is the probability that the fragment read by the sequencer originated from the alignment location.

A paired alignment score comprises three parts, Needleman-Wunsch alignment scores for each end of the pair in the form $-10\log_{10}(P(R| Ai))$ and a fragment length penalty in the form $-10\log_{10}(P(l | F))$ calculated from the fragment length distribution, F.

A posterior alignment score or quality is also given and is $-10\log_{10}(1 - P(Ai| Ai, G, F))$ where P(Ai| Ai, G, F) is the probability of the alignment location given the read, R; the genome, G; and the fragment length distribution, F. For paired end reads the quality score is limited to not more than 150.

Setting of gap penalties and threshold is similar to single end novoalign.

Alignment process

With paired end reads Novoalign can have "proper fragments" and pairs that don't fit the fragment model.

The alignment process works as follows:

For Read1 Novoalign uses a seeded alignment process to find alignment locations each with a Read1 alignment score. For each good location found Novoalign does a Needleman-Wunsch alignment of the second read against a region starting from the Read1 alignment and extending 6 standard deviations beyond mean fragment length. The best alignment for Read2 will define the pair score for Read1/Read2. All the alignments are added to a collection for Read1.

This process is repeated using Read2 seeded alignment and then N-W for Read1, creating a collection of Read2/Read1 pairs. There are very likely duplicates amongst the two collections.

Novoalign then decides whether there is a "proper pair" or not. To do this a structural variation penalty is used as follows.

Novoalign has a proper pair if the score of the best pair (Read1/Read2 or Read2/Read1 combined score including fragment length penalty) is less than the structural variation penalty (default 70) plus best single-end Read1 score plus best single-end Read2 score.

If Novoalign has a proper pair, Read1/Read2 & Read2/Read1 lists are combined, removing duplicates and sorting by alignment score. At this point Novoalign has list of one or more proper pair alignments. This list is passed to reporting which can report one or more alignments depending on the options.

If there wasn't a proper pair then Novoalign basically report as single-end alignments to each read and reporting options will decide whether Novoalign report one or more alignments.

The result of the paired search can be two paired alignments where the pairing is more probable than a structural variation, or it can be two individual alignments, one to each end of the pair.

Given the threshold, gap penalties and reads it is quite possible for novoalign to find alignments with gaps in both ends of the reads. There are no design restrictions that prevent this type of result and it depends only on the scoring parameters and threshold.

Output Format

Native, Pairwise and SAM report formats are supported.

Paired End Native Report Format

This example is for native format with good pairs found. The alignment score for one of the reads in the pair will include the fragment length penalty. The quality score is based on the posterior fragment alignment probability.

```
# novoalign (2.0) - short read aligner with qualities.
# (C) 2008 NovoCraft
# Licensed for evaluation and educational Use Only
#  novoalign -d ssuis -f ../../simlft/s_1_sequence.txt
../../simrgt/s_1_sequence.txt
# Index Build Version: 1.0
# Hash length: 11
# Step size: 1
@Ssuis_633667_633825_0/1   L   GCTCAATGACTATCCGCAGATTGAGGGGTTTCTGCT
   IIIIIIIIIIIIIIIII,IIIII(,;!%$3C;I>!! U   51  150 >Ssuis 633790 R   .   633667 F
@Ssuis_633667_633825_0/2   R   GTCTGACTCATGGCTGTGCGAATGGCTTCTTCCCTA IIIIIIIIIIII-
IIIIIIIIIIIIIIIII0%%!!U   16  150 >Ssuis 633667 F   .   633790 R
@Ssuis_1657428_1657600_1/1 L   AGTACGTGTCAATATCGTCCACTCTGCAGGTGGTCC
   IIIIIIII+IIIIIIIIIIIIIIIIIIII+CB-4%7 U   42  150 >Ssuis 1657565   R   .
   1657428   F   2C>G 7A>C
@Ssuis_1657428_1657600_1/2 R   TGTAAATGATGCTGTGAAGACGTACTTCAACATCAT
   IIIIIIIIIIIIIIIIIBIIII6I<)IIIII)I+7( U   3   150 >Ssuis 1657428   F   .
   1657565   R
@Ssuis_973563_973724_f/1   L   TTACCAAGCGTGGTAATCCCTACGCTAGAAAGATTC
   IIIIIIIIICIIIIIIIIIIIIII'%$2III-II-2 R   2
@Ssuis_973563_973724_f/2   R   TGGCACCAATCGTGTGCAGCTTCGTTGAAGTCGTTT III%!!%
+IIIIIIIIIIIIIIIIII+IIIIII,II R   2
...
#       Paired Reads:     2000
#      Pairs Aligned:     2000
#     Read Sequences:     4000
#            Aligned:     4000
#   Unique Alignment:     3940
#   Gapped Alignment:        9
#     Quality Filter:        0
# Homopolymer Filter:        0
#       Elapsed Time: 0,313s
# Done.
```

This example is for native format when a good pair was not found. In this case both alignments were on different chromosomes. The quality values reflect the quality of the individual end alignments.

```
@SLXA-EAS1_34_FC4751_R1_1_1_53_21       L
TTGATGGATCAATTGTAGTTGCCTGCAATAAGAGG     ???????????????????????7??:?????9+2$
U       23      150     >III    7197040 R       >IV
11532213        F       3G>T
@SLXA-EAS1_34_FC4751_R1_1_1_53_21       R       AATTGGAAGAGGACAGAAGAGATGA
=====================&==+       U       1       93      >IV     11532213
```

```
F       >III    7197040 R
@SLXA-EAS1_34_FC4751_R1_1_2_993_712     L
GTGCCTACCATTGTGATTCGACTATATACGCGCTC     ???????8?8?????????5?09?5?7?*(&&7%7,
U       6       150     >IV     5943661 F       >I      4229259 R
@SLXA-EAS1_34_FC4751_R1_1_2_993_712     R       GGGAAAAGGTGCCAAAAAGTATAGA
<<<1<<<<<<<1<-//4-<31<3<-       U       0       94      >I      4229259 R
>IV     5943661 F
```

This example is for native format with multiple alignments to a read and using -r All option.

```
>8_100_1_16  L  TTACCAAGCGTGGTAATCCCTACGCTAGAAAGATTC IIIIIIIIICIIIIIIIIIIIIII'%
$2III-II-2   R  6  3  >Streptococcus_suis 973563 F  .  973689 R
>8_100_1_16  R  TGGCBDCAATCGTGTGCAGCTTCGTTGAAGTCGTTT III%"#%
+IIIIIIIIIIIIIIIIII+IIIIII,II R  41 3  >Streptococcus_suis 973689 R  .  973563
   F
>8_100_1_16  L  TTACCAAGCGTGGTAATCCCTACGCTAGAAAGATTC IIIIIIIIICIIIIIIIIIIIIII'%
$2III-II-2   R  6  3  >Streptococcus_suis 1717310  R  .  1717184  F
>8_100_1_16  R  TGGCBDCAATCGTGTGCAGCTTCGTTGAAGTCGTTT III%"#%
+IIIIIIIIIIIIIIIIII+IIIIII,II R  41 3  >Streptococcus_suis 1717184  F  .
   1717310  R
```

Paired End Pairwise Report Format

The pairwise (Blast like) output format includes a pair header. The details of the pairwise format depend on whether the alignment process found a pair or whether it is reporting individual alignments.

In this example, both paired reads aligned to a fragment that fit the fragment distribution.

```
# novoalign (2.0) - short read aligner with qualities.
# (C) 2008 NovoCraft
# Licensed for evaluation and educational Use Only
#  novoalign -o P -d ssuis -f simlft/s_1_sequence.txt simrgt/s_1_sequence.txt
# Index Build Version: 1.0
# Hash length: 11
# Step size: 1

Pair Query1=@Streptococcus_suis_633667_633825_0/1
Query2=@Streptococcus_suis_633667_633825_0/2

ALIGNED PAIRS:
Pair Alignment(1) >Streptococcus_suis 633667<->633790 Score=-67 Quality= 150
Query=@Streptococcus_suis_633667_633825_0/2
Length=36
>Streptococcus_suis
Length=2007491

 Score=16, Quality=150
 Strand=Plus/Plus

Query          1 GTCTGACTCATGGCTGTGCGAATGGCTTCTTCCCTA  36
                 ||||||||||||||||||||||||||||||||||
Sbjct     633667 GTCTGACTCATGGCTGTGCGAATGGCTTCTTCCCGG  633702

Query=@Streptococcus_suis_633667_633825_0/1
Length=36
```

```
>Streptococcus_suis
Length=2007491

 Score=51, Quality=150
 Strand=Minus/Plus

Query         36 AGCAGAAACCCCTCAATCTGCGGATAGTCATTGAGC  1
                   ||||| | ||||||||||||||||||||||||||
Sbjct     633790 GTCAGAATCACCTCAATCTGCGGATAGTCATTGAGC  633825

...
Pair Query1=@Streptococcus_suis_1362887_1363089_7cf/1
Query2=@Streptococcus_suis_1362887_1363089_7cf/2

ALIGNED PAIRS:
Pair Alignment(1) >Streptococcus_suis 1363054<->1362887 Score=-35 Quality= 150
Query=@Streptococcus_suis_1362887_1363089_7cf/2
Length=36
>Streptococcus_suis
Length=2007491

 Score=3, Quality=150
 Strand=Minus/Plus

Query         36 AAAATCCTCACGAATTTTTCGATTTGGATAATATTT  1
                 ||||||||||||||||||||||||||||||||||||
Sbjct    1363054 AAAATCCTCACGAATTTTTCGATTTGGATAATATTT  1363089

Query=@Streptococcus_suis_1362887_1363089_7cf/1
Length=36
>Streptococcus_suis
Length=2007491

 Score=32, Quality=150
 Strand=Plus/Plus

Query          1 ACGATACCTGTTAAGGCAGTCGGGAATAGAATTTAC  36
                 ||||||||||||||||||||||| |||||||||| |
Sbjct    1362887 ACGATACCTGTTAAGGCAGTCGGTAATAGAATTTTC  1362922
#       Paired Reads:     2000
#      Pairs Aligned:     2000
#     Read Sequences:     4000
#            Aligned:     4000
#   Unique Alignment:     3940
#   Gapped Alignment:        9
#     Quality Filter:        0
# Homopolymer Filter:        0
#       Elapsed Time: 0,318s
# Done.
```

In this example a paired alignment could not be found so alignments to individual reads were reported. The second read of the pair failed to align.

```
Pair Query1=@22_6989814_6989984_6a/1 Query2=@22_6989814_6989984_6a/2
No significant pairs found, reporting individual algnments.

Query=@22_6989814_6989984_6a/1
Length=25
ALIGNMENTS
```

```
>22
Length=10058659

 Score=0, Quality=58
 Strand=Plus/Plus

Query          1 GGGCTCAGCGCTCTTCCTAAGCGGC  25
                 |||||||||||||||||||||||||
Sbjct    6989880 GGGCTCAGCGCTCTTCCTAAGCGGC  6989904

Query=@22_6989814_6989984_6a/2
Length=25
No significant similarity found.
```

Paired End SAM Report Format

SAM report format is for use with SAMtools, just add the option -oSAM to the command line.

The report format is documented as part of SAM/BAM specification at http://samtools.sourceforge.net/

Bisulphite Mode

Bisulphite mode requires building of a double index, the first uses a hash table with all Cs translated to T's and the second a hash table with Gs translated to A's for fragments off the complementary strand.

Memory utilisation for the index may be higher in bisulphite mode than normal mode as we now have two hash tables. Novoindex will choose k &s values that allow the index to fit in RAM if possible. You can reduce memory further by increasing s or decreasing k.

Alignment is done iteratively gradually increasing error tolerance until a match is found. Each round of iteration will align the read in forward and reverse complement against the CT and the GA index. During CT alignment Cs in the read are translated to Ts for hash lookup, then during alignment, T's in the read can align to a T or a C in the reference sequence with no penalty. The process is then repeated for the GA alignment.

Scoring for alignments is similar to normal alignment scoring with difference that T in the read can align to a C in the reference without any penalty (or A to G for GA index alignments). This means that methylation status does not affect the alignment score.

I addition there is a command line option, -u, to impose a penalty on unconverted cytosines at CHG and CHH positions. If specified each unconverted cytosine in CHG or CHH positions in a read will be penalised thus biasing alignment in favour of methylated CGs.

The low-level of non-CpG methylation in vertebrates and the incomplete bisulphite conversion of unmethylated cytosines should be factored in to selecting this value. As a rough guide, a penalty can be worked out as follows:

Let P_{UC} be the probability an non-methylated cytosine is not converted, P_{CG} the probability that a cytosine at CpG is methylated and P_{CH} be the probability that a cytosine at a CHG or CHH is methylated. Then the probability of reading a cytosine at a CG position is:

$P(C|CG) = P_{CG} + (1 - P_{CG}).P_{UC}$

and the probability of reading a C at a CHN position is:

$P(C|CH) = P_{CH} + (1 - P_{CH}).P_{UC}$

We can then convert to log (phred) scale and calculate a penalty as:

$Penalty = -10\log_{10}(P(C|CH)) + 10\log_{10}(P(C|CG))$

Applying values from Ramsahoye et al. [6] for Drosophila

$P_{CG} = 62\%$, $P_{CH} = 3\%$ (derived)

and

$P_{UC} = 1\%$

$Penalty = -10\log_{10}(.03 + .97 * .01) + 10\log_{10}(.62 + .38 * .01)$

$= -10\log_{10}(.04) + 10\log_{10}(.66))$

[6] Ramsahoye BH, Biniszkiewicz D, Lyko F, Clark V, Bird AP, Jaenisch R. Non-CpG methylation is prevalent in embryonic stem cells and may be mediated by DNA methyltransferase 3a. Proc. Natl Acad. Sci. USA (2000) 97:5237–5242.[Abstract/Free Full Text]

= 14 – 2

= 12

As mentioned above, using a penalty for unconverted cytosines at CHG and CHH positions will slightly bias alignment in favour of methylated CG sites. This will mainly have an effect when there are multiple alignment sites with similar scores.

Novoalign will switch to bisulphite alignment mode whenever a bisulphite index is used.

Bisulphite Report Format

The differences to the output format are:

1) an indication of whether CT or GA index was used for the alignment. This is reported before mismatches and delimited from mismatches by a space.
2) Mismatches caused by unmethylated cytosines are shown with a hash '#' rather than a greater than '>' symbol. e.g. 5C#T to indicate a C in reference aligns to a T in the read and may be an unmethylated cytosine that was converted to uracil by bisulphite treatment. Similarly, 6G#A indicates a Cytosine on the complementary strand was unmethylated and hence appears in the read as an A.

The mismatch list does not show methylated cytosines as they match the reference sequence.

```
@chr2_98467308_98467344_1/1     S       CGGTATTGTAGAATAGTGTATATTAATGAGTTATAA
CBC??-@@BBBBBBB@@@@@BB??;??,6247092.    U       0       15      >chr2
98560453        R       .       .       .       GA 7G#A
@chr2_115989213_115989249_1/1   S       CGGTTTATTTTTTTTGGGGAATAGATTAAGTTTAAT
CCCCC-CCCCCCC==-?775BBB>>BC=B899;;9>    U       0       107     >chr2
116079348       F       .       .       .       CT 10C#T 13C#T
@chr2_48440862_48440898_1/1     S       CGGATATGTTATTTTAGGAGAAAAGAGGAAAAAATT
CCCCC=CCCCCCCCDD?BBC@CCCC@?2::<<022B    U       44      23      >chr2
48578844        R       .       .       .       GA 2T>A 9T>C 15G#A
@
```

Quality Calibration

Quality calibration is the process of reevaluating base qualities using the actual counts of mismatches from alignments. The calibration in Novoalign is base specific which means two things:

1. We keep mismatch counts based on the actual base called so we can detect situations where, say, T is overcalled and likely to be wrong but calls of A, C &G are likely to be correct.
2. Rather than count “mismatches” we maintain counts for each of the bases aligned. This allows us to detect situation where a wrong call of , say, a T is more likely to be an A than a C. We can then calculate mismatch penalties specific to each base at each position in a read.

These counts are used to calculate an actual mismatch probability or penalty as a function of: the position in the read; the “as called” base quality; the base called; and the base aligned. The mismatch probability is then used in Novoalign alignment process in place of the “as called” base quality to set penalties for the alignment dynamic programming.

Categories used for counting mismatches are:

- The read within the pair (0 for first read, 1 for second read)
- The base position in the read, zero based.
- The “as called” quality
- The base or colour called

For each combination, Novoalign maintains the count of the number of alignments to each of the four bases, M_A, M_C, M_G & M_T. Only ungapped alignments with a quality > 60 , or >120 for paired end, are used to count mismatches.

The first step in the process of calculating calibrated qualities for each category involves binning counts across read length and quality values. Binning helps to increase the counts and to smooth fluctuations. Bins are 5 bases long and have variable number of quality values. At low qualities bins take a single quality value, in mid range bins are 3 quality values wide and above a quality of 30 they are 5 wide. There is a bin for each base position and quality values so mismatch counts get added to multiple overlapping bins, this design eliminates edge effect between bins.

The second step involves adding priors to the count of calls and mismatches. Use of a prior helps stabilise calibrated quality values when counts are low. The prior is a minimum value for mismatch count and if the actual mismatch count is below the prior then we add extra mismatches to bring the count up to the prior and then a corresponding number of extra matches based on the “as called” quality. Unaligned reads (status NM) are also added to the priors as examples of correct base and quality calls.

Novoalign then calculates 4 base penalties is $P_I = -10\log_{10}(M_I/N)$ for I in [ACGT] where M_I is the number of times an alignment matched base I and N is the total calls for this bin. The penalties are used in the dynamic programming alignment.

A Phred scaled quality value is also calculated as $P = -10\log_{10}(M/N)$ where M is the total mismatches and N the total calls for the bin. This calibrated quality value is used in the report for the base qualities.

For colour space quality calibration we only track the number of correct calls and colour errors for each category. Calibrated penalties are specific to colour called, position in read and quality called, but not to the substituted colour.

Using Quality Calibration

Quality calibration works for read files in the following formats:

● Solexa & Illumina FASTQ	
● Sanger FASTQ	
● FASTA	Every base is assumed to have a starting quality of 30.
● FASTA with separate quality file	
● CSFASTA	Without a quality file we assume a colour quality of 20.
● CSFASTQ	

Quality calibration does not work with prb files.

The simplest way to use quality calibration is just to add the option **-k** to the Novoalign command line. This turns on calibration with calibration based on actual alignments. The calibration will start off neutral as a result of the priors and gradually, as more alignments are added, the calibration will shift to reflect the actual mismatch counts.

Novoalign also has the ability to save the mismatch count data and then use this as input to the calibration of a following run of Novoalign. Scenarios where this might be used include:

- Using mismatch counts from phiX lane to calibrate another lane
- Running an initial Novoalign at a low threshold to get mismatch statistics for use in a following run, possibly at a higher threshold. This would remove some startup effects from a single pass run.

Operation is controlled by two command line option:

-k [infile]	Enables quality calibration. The quality calibration data (mismatch counts) are either read from the named file or accumulated from actual alignments. Default is no calibration. Note. Quality calibration does not work with reads in prb format.
-K [file]	Accumulates mismatch counts for quality calibration by position in the read and called base quality. Mismatch counts are written to the named file after all reads are processed. When used with -k option the mismatch counts include any counts read from the input quality calibration file.

These two options can be used in several combinations :

-k	Turns on calibration with mismatch counting. Effects of calibration can be seen after a few thousand reads have been aligned. Calibration data is recalculated periodically as more reads are aligned.

-k *infile* Turns on calibration with mismatch counts read from *infile*. Mismatch counts from alignments are not used.

-K *outfile* Turns on mismatch counting without calibration. At the end of the run the mismatch counts are written to the *outfile* ready for use as input in another run.

-k -K *outfile* Turns on calibration with mismatch counting. At the end of the run the mismatch counts are written to the *outfile* ready for use as input in another run.

-k *infile* -K *outfile* Turns on calibration and mismatch counting. Initial mismatch counts are loaded from *infile*, new alignments are added to the counts, and then at the end of the run the mismatch counts are written to the *outfile* ready for use as input in another run. Calibration data is recalculated periodically as more reads are aligned.

Quality Calibration and Novoalign Reports

There is no change to the report format, for Novoalign the quality string displayed is now the calibrated qualities. For NovoalignCS the calibrated qualities are not displayed. They are used internally during alignment as colour error penalties and then used to calculate base qualities.