Transcript Discovery
Plugin

USER MANUAL

User manual for

Transcript Discovery 25.0

Windows, macOS and Linux

November 21, 2024

This software is for research purposes only.

QIAGEN Aarhus AS
Kalkværksvej 5, 11.
DK - 8000 Aarhus C
Denmark

Contents

Chapter 1

Introduction

The Transcript Discovery plugin is designed to discover transcripts by mapping RNA-Seq sequencing reads to a genomic reference, allowing large gaps (for introns), followed by a transcript discovery process where transcripts are inferred from the read mappings. Note that the Transcript Discovery tool has been tested to work well with other alignment tools including STAR, TopHat2, GSNAP and HISAT2.

The detection of novel transcripts from short-read sequencing data is only possible with low precision and sensitivity. Therefore these tools are focused on improving existing annotations for non-model eukaryotic species, updating an annotation based on RNA-Seq data and/or generating transcript and gene tracks to serve as a common reference for differential expression analysis using the RNA-Seq Analysis tool.

Best practices The proposed workflow for using the Transcript Discovery plugin in combination with the existing RNA-Seq tool in CLC Genomics Workbench is:

1. Run the Large Gap Read Mapping tool using all your RNA-Seq reads and a genomic reference sequence.
2. Run the Transcript Discovery tool on the resulting read mapping to predict transcripts and genes.
3. Inspect the results and if necessary re-run the transcript discovery to refine the settings to produce the desired result.
4. Use the Predicted gene and Predicted Transcript tracks in the existing RNA-Seq tool in the Workbench.

To run an experiment with multiple replicates and tissues, it is possible to supply several Large Gap Read Mappings at once to the tool. These are then processed as one data set. However, you should note that:

- Supplying multiple samples increases coverage, which typically leads to the detection of more low-expression genes.

- Supplying multiple samples where the transcriptome differs markedly between samples may lead to a loss of precision. For example, if Large Gap Read Mappings Track A supports transcript A, and Large Gap Read Mappings Track B supports transcript B, the algorithm may instead call transcript C - which might be a hybrid of A and B - because it does not understand that the reads it sees come from two samples.
- Running two samples sequentially in the order "Sample A" and "Sample B" will give a different set of transcripts than the one obtained when running them in the order "Sample B" and "Sample A".
- Running two samples sequentially in the order "Sample A" and "Sample B" will give a different set of transcripts than the one obtained when supplying "Sample A" and "Sample B" together.

For these reasons, we recommend to run all replicates of the same condition together, and to run different conditions sequentially.

For example, if you had 4 "leaf" samples and 4 "root" samples from a plant, then you should run the tool on all 4 "leaf" samples and provide the output transcript track as input for the next invocation of the tool with the 4 "root" samples. You should later remap the samples separately using RNA-Seq Analysis, and prune away any annotations that have little or no expression in all the conditions. Note that this pruning of annotations can be necessary if, for example, the "leaf" data does not support a long transcript so a short one is predicted. However the long transcript is unambiguously present in the "root" data. Revisiting the leaf data after the long transcript is known might show that the long transcript is a good fit here too. The original short transcript might then be pruned away.

Known limitations The Transcript Discovery has the following known limitations:

- The Large Gap Read Mapping tool can only align reads when at least 10% of the read maps without splicing. This requirement means that reads spanning more than 10 exons are less likely to be mapped.
- Alternative transcript isoforms that are a strict subset of existing transcripts (i.e. they differ only by having TSS and TES at different positions/exons but share all intervening exons), cannot be distinguished. Only the longest transcript will be reported in these cases.
- Transcripts spanning the origin of circular chromosomes will be reported as two disconnected transcripts: one at the start of the chromosome and one at the end.
- If the predictions generated by the Transcript Discovery tool are supplied as annotations, and a new round of prediction is performed on the same input read mapping, then a small number of novel transcripts and genes will still be identified. This is because the set of known annotations can affect which events are filtered, and lead to small changes in the predicted genes and transcripts.
- When used with short read data, all tools that attempt to recover full length transcripts are likely to produce many false positives - typically at least 50% for human RNA-Seq data [Hayer et al., 2015].

Chapter 2

Large Gap Read Mapping

The **Large Gap Read Mapping** tool maps reads to a reference, while allowing for large gaps in the mapping. It is developed to support transcript discovery using RNA-Seq data, since it is able to map RNA-Seq reads that span introns without requiring prior transcript annotations.

The **Large Gap Read Mapping** tool works by iteratively applying the standard read mapper of CLC Genomics Workbench to each read as follows:

1. Find the best match for the read.
2. If the match is good enough (according to the settings, see below), the read is mapped to this position.
3. If there is an unaligned end which is long enough for the mapper to handle (15 bp for standard mapping), this part of the read is used as input to step 1.
4. This continues until no reads have unaligned ends that are longer than 15 bp. The number of rounds required scales with the length of the reads. For short 100 bp reads the maximum will usually be maximum three rounds of mapping (corresponding to spanning two introns). For full-length transcripts more than ten rounds may be required.

The matched region of the read identified in the first round of the mapping is called the seed segment (or just 'seed'). Matched regions found in later rounds are called non-seed segments.

To run the **Large Gap Read Mapping** tool, go to:

Tools | **RNA-Seq Analysis (**  **)** | **Transcript Discovery (**  **)** | **Large Gap Read Mapping**

First specify the RNA-Seq reads that should be analyzed (figure 2.1):

In the next dialog, specify the reference (a sequence track or a sequences list) to which the reads should be mapped (figure 2.2):

In the Mapping options dialog, specify the following parameters (figure 2.3):

- **Maximum number of hits for a segment** is the maximum number of hits that a segment is allowed to have in order for the read to be mapped. If, for a non-seed segment, this number is exceeded, the read is classified as unmapped. If it is not exceeded, all the

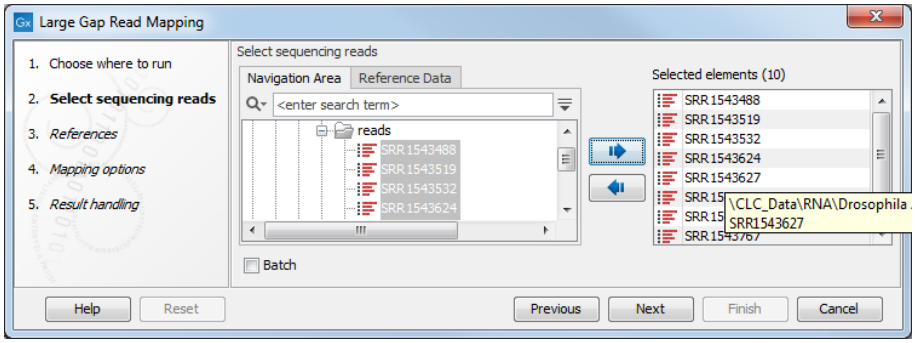


Figure 2.1: *Selecting input reads for the Large Gap Read Mapping tool.*

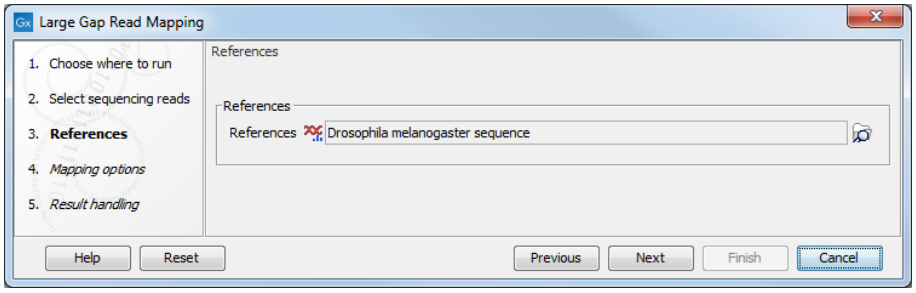


Figure 2.2: *Selecting references for the Large Gap Read Mapping tool.*

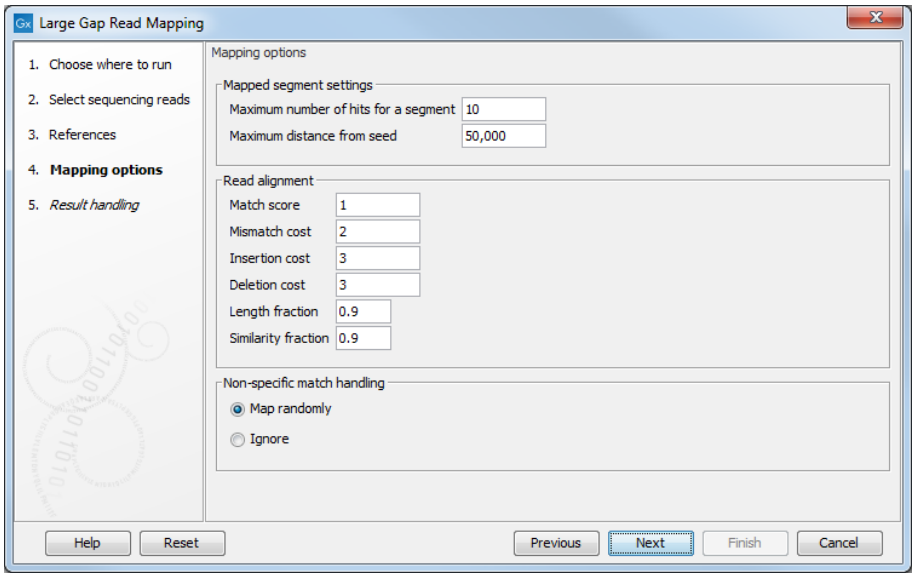


Figure 2.3: *Specify the parameters for the Large Gap Read Mapping tool.*

multiple hit positions will be considered. If the seed makes up the full read it may map in up to 'Maximum number of hits' positions.

- **Maximum distance from seed** is the maximum distance allowed between the first match part (the seed) and any subsequent match part (non-seed segments). Matches that are found further away from the seed than this value are discarded. Note that in many open source tools users are required to specify the maximum intron size. But here, setting the maximum distance between seed and non seed segments means that this value should be

the maximum number of reference bases that a read can span. For long reads spanning many introns, this will be on the order of the length of a typical long gene, and *not* the length of a typical intron.

- **Read alignment**:
  - Match score: The positive score for a match between the read and the reference sequence. It is set by default to 1 but can be adjusted up to 10.
  - Mismatch cost: The cost of a mismatch between the read and the reference sequence. Ambiguous nucleotides such as "N", "R" or "Y" in read or reference sequences are treated as mismatches and any column with one of these symbols will therefore be penalized with the mismatch cost.
  - Insertion cost: The cost of an insertion in the read (a gap in the reference sequence). cost of an insertion of length l will be l*insertion cost.
  - Deletion score: The cost of a deletion in the read (gap in the read sequence). The cost of a deletion of length l will be l*deletion score.
  - Length fraction: The minimum percentage of the total alignment length that must match the reference sequence at the selected similarity fraction. A fraction of 0.5 means that at least half of the alignment must match the reference sequence before the full read is included in the mapping (if the similarity fraction is set to 1).
  - Similarity fraction: The minimum percentage identity between the aligned region of the read (segment) and the reference sequence. This means that all segments must fulfill this requirement. Since segments can be as short as 15 bp, this threshold should not be set too strictly: for example, setting the threshold at 0.9 means that two errors for a segment of 15 bp would discard the match.
- **Non-specific match handling** decides whether **non-specific matches** should be distributed randomly or ignored

Note that in addition to these mapping settings, the Large Gap Read Mapping tool requires that each mapped segment must be of minimum length 15 bp, and that at each mapping step, the mapped segment must comprise at least 10% of the read being mapped. This will initially be the full read length, but in later rounds it will be the length of the remaining unaligned part of the read.

In addition to a reads track (figure 2.4), the tool can generate the following items:

- a report on the mapping, containing various statistics on the mapping, such as the distribution of number of segments per read matching the reference (match parts), the distribution of gaps between the match parts and paired read mapping statistics. In the mapping report, "Unaligned internal gaps" are (small) unmapped parts of the read between mapped segments, whereas "gap between match parts" is the distance between the mapped read segments on the reference. For cDNA reads mapped to genomic sequences, these distances correspond to intron size.
- a list of invalid mapped reads, listing reads for which the Large Gap Read Mapping tool was able to find a mapping, but for which the mappings of the segments were incompatible, i.e., if their positions are not consecutive along the reference, or if they do not have the same direction.

- a list of unmapped reads, for reads that the Large Gap Read Mapping tool was not able to map.

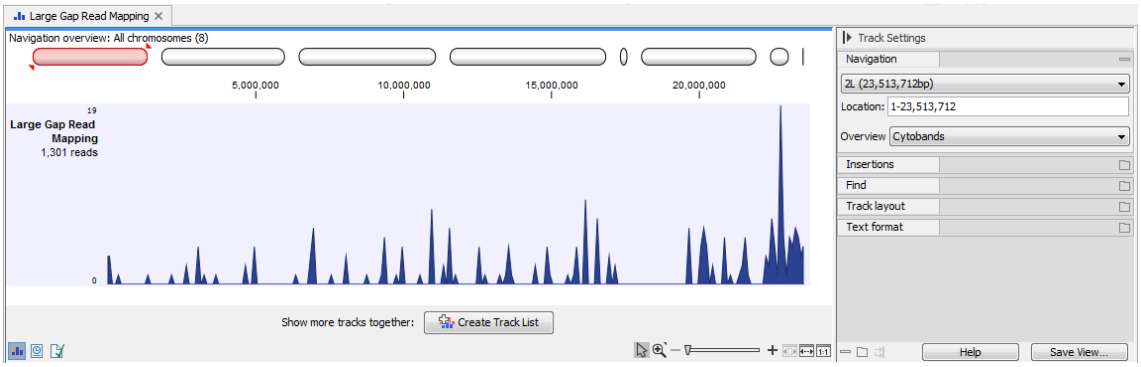


Figure 2.4: *The Large Gap Read Mapping track.*

Chapter 3

Transcript Discovery

3.1 Running the Transcript Discovery tool

To run the **Transcript Discovery** tool, go to:

Tools | **RNA-Seq Analysis (**) | **Transcript Discovery (**) | **Transcript Discovery**

Select the read mapping produced by the **Large Gap Read Mapping** tool (figure 3.1).

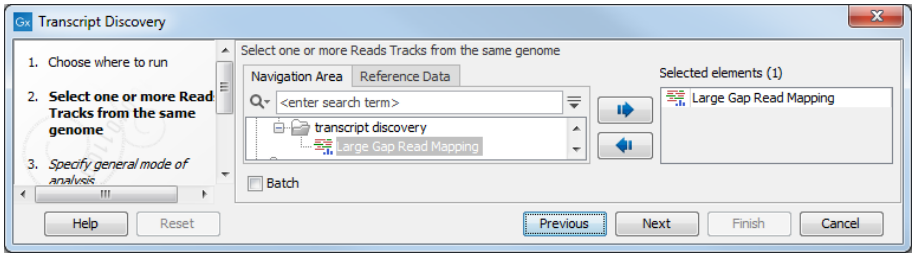


Figure 3.1: *Specify a Large Gap Read Mapping track.*

You are now presented with choices regarding the overall mode of analysis as shown in figure 3.2.

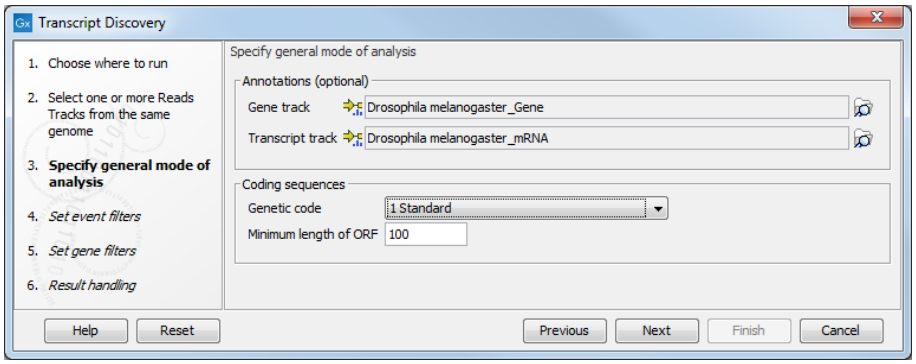


Figure 3.2: *Specifying the overall mode of analysis.*

You can specify a gene and a transcript track to add annotations to your analysis, but this is optional. You can also choose the Genetic code from a drop down menu, and the Minimum length of ORF desired.

In the Set event filters dialog figure 3.3, the options are as follow:

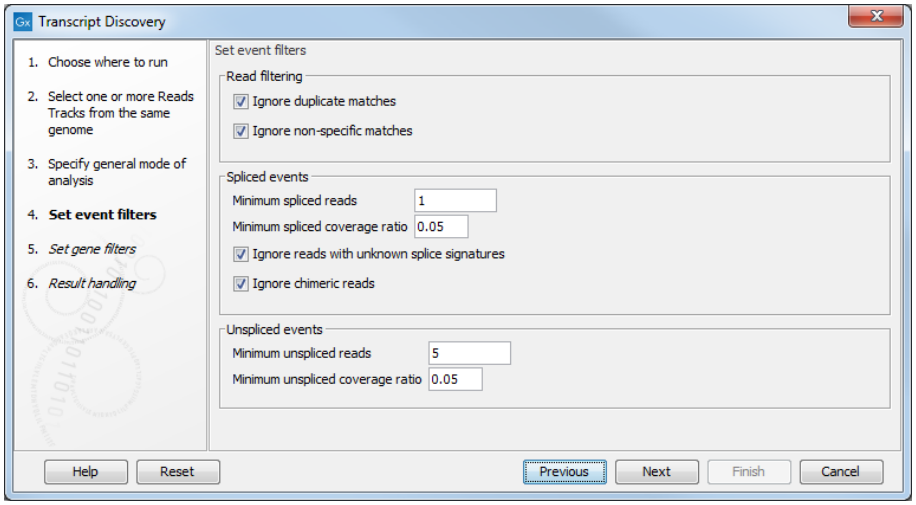


Figure 3.3: *Specifying the event filters.*

- **Read filtering**
  - Ignore match duplicates. For reads that are 100% identical, only one copy is used to define events. This is relevant for the 'supporting read counts' that are used when filtering events. When ticked, identical reads will only be counted as '1' in the read counts.
  - Ignore non-specific matches. Reads that have an equally good match elsewhere on the reference genome (these reads are colored yellow in the mapping view) can be ignored in the analysis. Whether you include these reads or not will be a tradeoff between sensitivity and specificity. Including them may lead to the prediction of transcripts that are not correct, whereas excluding them may mean that you will lose some true transcripts.
- **Spliced events**
  - Minimum spliced reads. This filter removes spliced events with weak evidence by defining the minimum number of *unique* spliced reads that must support a spliced event. Events that do not meet this requirement are ignored. A read is unique if it 'counts' as specified by the 'Read filtering' options: if the 'Ignore duplicate reads' option is checked, identical spliced reads are counted as 1, and if the 'Ignore non-specific matches' is checked, non-specific matches are not counted. Minimum spliced reads is set by default to 1, meaning that by default no filtering is happening.
  - Minimum spliced coverage ratio. This filter removes spliced events with weak relative evidence. The *spliced coverage of a region* is calculated as the number of spliced reads in the gene region, divided by the total length of the region consisting of the union of the exons in the events in the region. Similarly, the spliced coverage of an event is calculated as the number of spliced reads supporting the event, divided by the length of the exon regions of the event. If the spliced coverage of the event divided by the spliced coverage of the region is smaller than this value, the event is ignored. Compared to the filter on absolute read count above, the coverage ratio filter allows filtering of events with weak evidence in regions of high coverage.

- Ignore reads with unknown splice signatures. As canonical splice signatures are overwhelmingly found (GT - AG, GC - AG and AT - AC), this filter typically removes badly mapping reads that do not fit theses signatures and that are likely assembly errors. Consequently, this filter removes events containing uncertain positions (see figure 3.4 for an example of how such an event is represented in the Workbench).

Figure 3.4: *Uncertain position as depicted in a track view.*

- Ignore chimeric reads. This filter removes spliced events that span two or more genes. Keeping these events will typically cause us to merge the genes in the output. The filter works by assigning spliced events to possible "chains" of exons. Events are chained if they are within 1000bp of each other. This value is hard-coded, and is chosen as a good upper-bound on exon length. Note that this length is unrelated to the "Gene merging distance" parameter, because that parameter looks for gaps in coverage between any events, whereas this length only cares about distances between spliced events. If a spliced read passes over a chain end and then a chain start, in that order, and on the same strand as itself, then it is inferred to have spanned two genes.

- **Unspliced events**
  - Minimum unspliced reads. This filter removes events with weak evidence by defining the minimum number of unique unspliced reads that must support an unspliced event. Events that do not meet this requirement are ignored. It is set by default to 5.
  - Minimum unspliced coverage ratio. This filter removes events with weak relative evidence. The *un-spliced coverage of a region* is calculated as the number of un-spliced reads in the transcript event region, divided by the total length of the region consisting of the union of the exons in the events in the region. Similarly, the un-spliced coverage of an event is calculated as the number of un-spliced reads supporting the event, divided by the length of the exon regions of the event. If the un-spliced coverage of the event divided by the un-spliced coverage of the region is smaller than this value, the event is ignored.

In the Set gene filters dialog figure 3.5, the options are as follow:

- **Gene filtering**
  - Gene merging distance. If two reads map closer than this value and on the same strand, then they are considered to be part of the same gene. This closeness criterion is re-evaluated several times during the process. For example, if an event is filtered away, it may create a gap greater than this size, leading to a new gene region. Setting this value too high may cause genes to be merged together. Setting the value too low may cause genes to be split in two or more pieces in low coverage regions. Genes supplied as known annotations are not merged.
  - Minimum reads in gene. This filter will remove predicted genes that have fewer supporting reads than this number. Note that genes supplied as known annotations

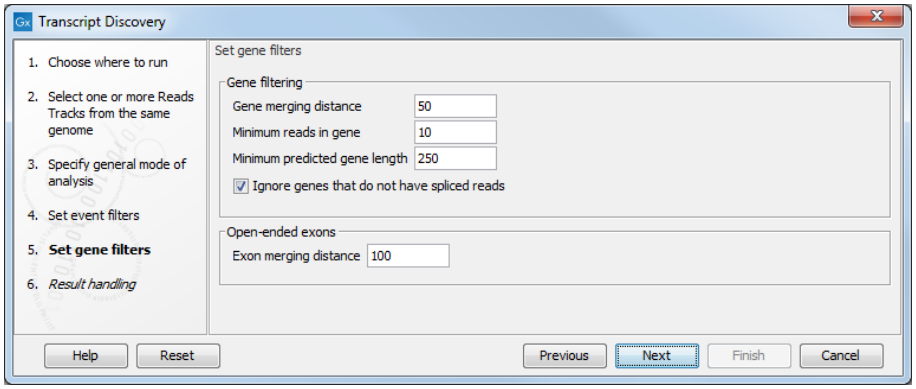


Figure 3.5: *Specifying the gene filters.*

are not ignored. This parameter is set at 10, which is suitable for Illumina reads, but can be lowered to 2 when working with very long reads such as PacBio data.

- Minimum predicted gene length. This filter will remove genes predicted to be shorter than this value. Note that genes supplied as known annotations are not ignored.
- Ignore genes that do not have spliced reads. This filter will remove genes that do not have spliced reads. Note that genes supplied as known annotations are not ignored. This option, on by default, can be turned off when working with PacBio long reads.

- **Open-ended exons**
  - Exon merging distance. Open ended exons that are within this distance of each other, and which do not have a splice junction between them will be joined together into one exon. This helps reconstruct full-length exons from regions with low coverage. Exons supplied as known annotations are not merged.

The Transcript Discovery tool can generate the following outputs (see 3.6).

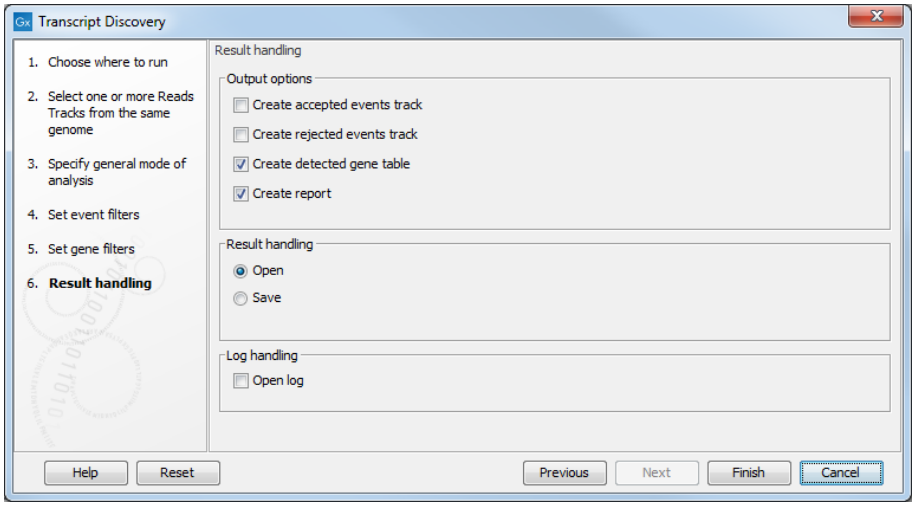


Figure 3.6: *Output options.*

3.2 Transcript Discovery output

The tool outputs are:

- A Predicted gene track
- A Predicted transcript track
- A Predicted CDS track
- An Accepted event track, including all the Events that made up predicted transcripts
- A Rejected event track, including all events and regions that were filtered away during execution of the algorithm. This track can be used to fine-tune settings.
- A Gene table, containing details of detected genes only. These details include the number of unknown and known transcripts per gene.
- A report

3.2.1 Tracks

The primary outputs of the Transcript Discovery tool are a gene and a transcript tracks containing all "known" genes/transcripts (if these were supplied as input), as well as some "unknown" genes/transcripts (i.e., those that we have predicted, unless none were predicted). Note that by definition, every unknown gene/transcript present in the track is detected, while "known" genes/transcripts may or may not be detected.

Predicted gene track A Predicted gene track viewed a table is shown in figure 3.7.

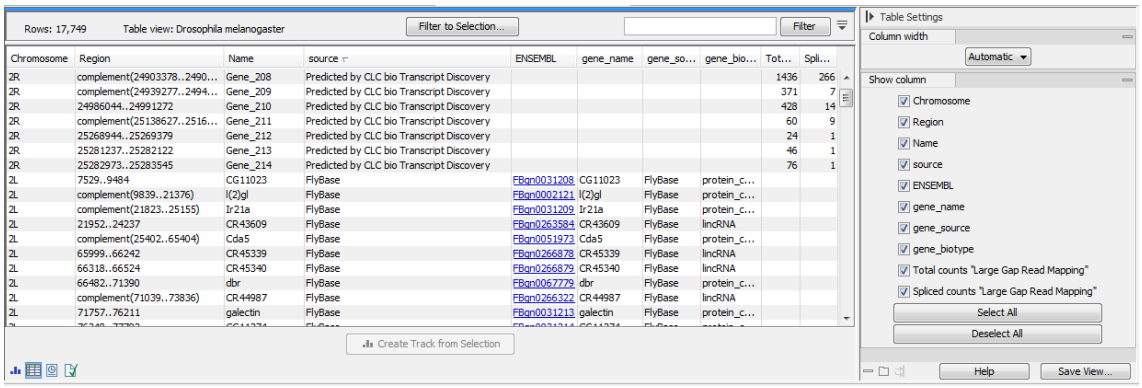


Figure 3.7: *A Predicted gene track seen as a table, and showing "known" and "unknown" genes.*

The column headers of the Predicted gene table are:

- Chromosome
- Region
- Name

- source. Every unknown gene or transcript will be described as "Predicted by CLC bio Transcript Discovery".
- ENSEMBL. Links to the ENSEMBL database for "known" genes.
- gene_name
- gene_source
- gene_biotype
- Total counts "sample name"
- Spliced counts "sample name"

Unknown genes are given names of the form Gene_1, Gene_2 etc. In order to avoid name clashes with previous predictions, the algorithm checks the previously annotated transcripts for genes with names of this form. New predictions will then be output with the next available index. For example, if a "Gene_11" is already present in the previously annotated transcripts, then the first new gene will be Gene_12.

These annotations sometimes include the sample name, such that if the tool is run on "Sample A", then the output transcript track will have table columns of the form "Coverage "Sample A"". This new track can then be supplied as input when the tool is run on "Sample B". The output transcript track will then have table columns "Coverage "Sample A"" and "Coverage "Sample B"".

Predicted transcript track Similarly a transcript track contains all the "known" input transcripts plus "unknown" transcripts if any were detected (figure 3.8).

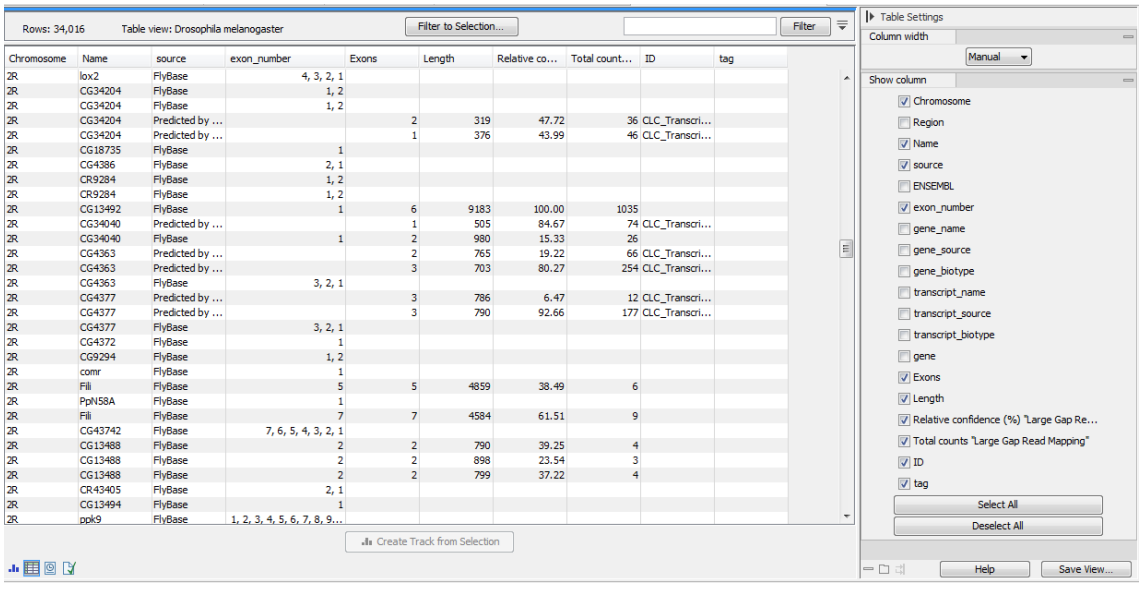


Figure 3.8: *A Predicted transcript track seen as a table, and showing "known" and "unknown" transcripts.*

For detected transcripts, the following annotations are available:

- Chromosome
- Region
- Name. Unknown transcripts wear the name of their gene, which guarantees that they can be linked with the gene in the RNA-Seq Analysis tool. The gene name for a particular transcript is extracted from the GFF3 parent/child relation if available (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=GFF3_format.html to learn more about GFF3 format).
- source. Every unknown gene or transcript will be described as "Predicted by CLC bio Transcript Discovery".
- gene
- Exons
- Length
- Relative confidence (%) "sample name". The abundance of a transcript: % of transcripts for a given gene that are expected to come from that transcript.

  Note that observed total relative confidence for all reported transcripts for a given gene can add up to less than 100%, as transcripts below 5

  The relative confidence does not use the % of reads, as a long transcript will produce more reads than a short transcript.
- Total counts "sample name"
- ID

Predicted CDS track Predicted CDS tables have, on top of some of the column headers described above, the following ones:

- codon_start
- Parent. This is the ID of the corresponding predicted transcript.

Note that uncertainty in whether an ORF might start before the first observed start codon can be seen by the use of an "open" position, for example <130 for an ORF that is annotated at position 130, but may actually begin earlier in the reference figure 3.9

Accepted events track Accepted events tables have, on top of some of the column headers described above, the following ones:

- Evidence for transcripts
- Boundaries moved based on nearby events
- Spliced evidence
- Coverage "sample name"

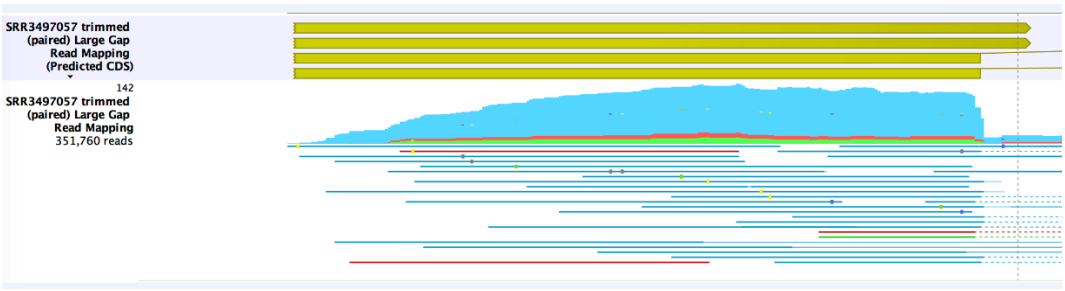


Figure 3.9: *A jagged line at the beginning of the annotation shows that the ORF may start before the first observed start codon.*

Rejected events track Rejected events tables have, on top of some of the column headers described above, the following ones:

- Name, which describes which filter caused an event to be excluded from the final set of predicted transcripts (see figure 3.10)
- Unspliced counts "sample name"
- Max intron length

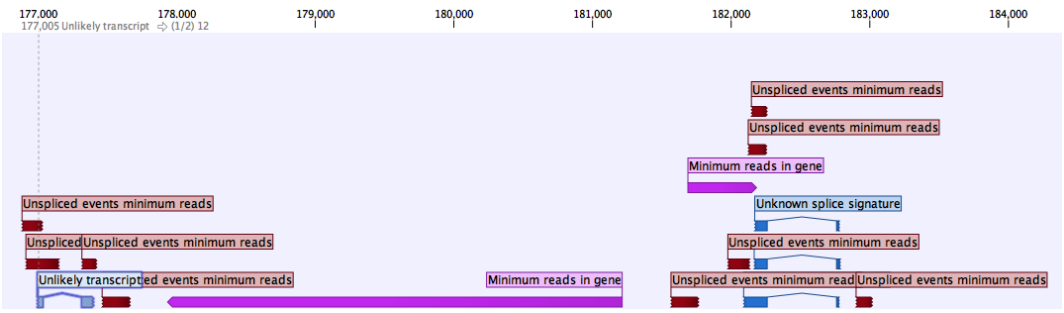


Figure 3.10: *Rejected events' Names as seen in the track view of the Rejected events track.*

The Name describing the filter causing an event to be rejected also refers to which filter can be adjusted in a second iteration of the tool to report the event in the Accepted events track:

- Event filters can be configured differently to accept the following rejected events:
  - **Minimum spliced reads**
  - **Minimum spliced coverage ratio**
  - **Unknown splice signature**
  - **Chimeric reads**
  - **Unspliced events minimum reads**
  - **Unspliced events minimum coverage ratio**
- Gene filters can be configured differently to accept the following rejected events:
  - **Gene without spliced reads**

- **Minimum reads in gene**
- **Minimum predicted gene length**

However, there are no configurable filters to change the rejection of the following events:

- **Unlikely transcript**. The event is only present in transcripts that are estimated to make up <5
- **Unspliced event inside**. intron The event is located entirely within introns of spliced events, and does not overlap with other unspliced events. It may be transcriptional noise.
- Low coverage outside spliced region The event is at the boundaries of a gene and has <25

Track list It can be very handy to see annotations reflecting the various steps in the transcript discovery in a Track list. An example is shown in figure 3.11

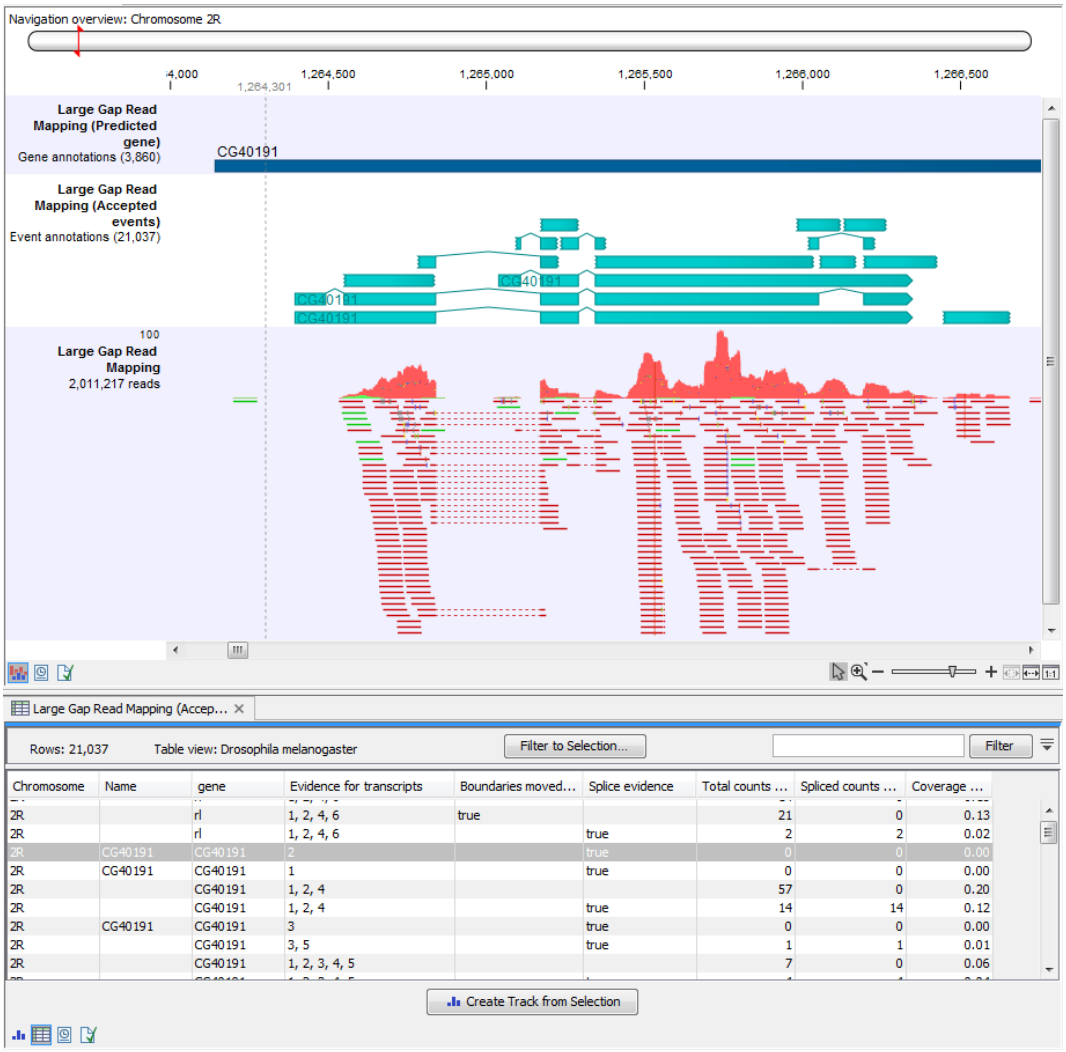


Figure 3.11: *A track list showing the read mapping along with the Accepted events track and the Predicted gene track.*

3.2.2 Gene table

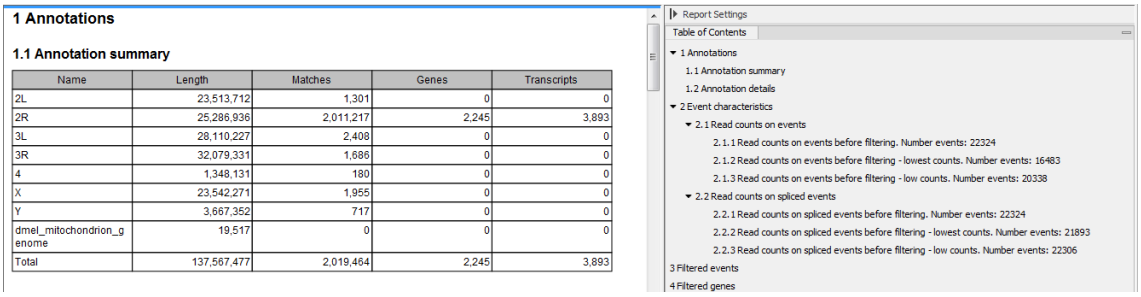
This table contains a row for each predicted gene. If annotation tracks were specified during the tool set up, annotations marked with a * in the following list are included in the table.

- Reference. The name of the chromosome or mapping in which the gene was predicted.
- Gene. The name of the gene if it was annotated prior to the analysis. If it is a new predicted gene the name will be 'Gene' followed by a number (e.g. 'Gene_1').
- Unknown*. No if the gene was annotated prior to the analysis; yes if it is a new predicted gene.
- Length. The length of the gene region.
- Start. The start of the gene region.
- End. The end of the gene region.
- Strand. The strand on which the gene was predicted.
- Transcripts. The number of detected transcripts for the gene (including prior annotated as well as new predicted).
- Known transcripts*. The number of prior annotated transcripts for the gene that were detected as being expressed in the sample.
- Unknown transcripts*. The number of new predicted transcripts for the gene.
- Longest transcripts. The length of the longest transcript for the gene.
- Novel splice junctions*. The number of novel splice junctions.
- Reads. The sum of the read counts of the events from which the transcript annotations were built.
- Spliced reads. The sum of the spliced read counts of the events from which the transcript annotations were built.
- New 5' sequence*. Yes, if the gene region extends 5' of the prior gene annotation if there was one, else no.
- New 3' sequence*. Yes, if the gene region extends 3' of the prior gene annotation if there was one, else no.
- Splicing description*. A summary of the types of new splice sites found for transcripts for the gene ('Alternative acceptor/donor' and/or 'new exon').

Note, that while predicting genes and CDS's, the Transcript Discovery tool will also attempt to identify the strandedness. The strandedness is determined from the canonical splice sites in the spliced reads. However, sometimes that information is not present for some of the predicted genes. This can be because there are no spliced reads or because those that are there do not use any of the canonical splice sites. In these instances, the strand will be indicated with a "?" because it can not be determined.

3.2.3 Summary report

The summary report holds various statistics on the annotations generated in the analysis, such as distributions of the lengths of genes, the numbers of transcripts per gene and the numbers of exons per transcripts. Also, there are statistics on the read counts on events and spliced events, as well as filtering of events and genes. These summaries can be used to get an overview of the overall performance of the generation of annotations, and may give a rough indication of whether the filtering was appropriate for your particular aim (figure 3.12).



1 Annotations

1.1 Annotation summary

Name	Length	Matches	Genes	Transcripts
2L	23,513,712	1,301	0	0
2R	25,286,936	2,011,217	2,245	3,893
3L	28,110,227	2,408	0	0
3R	32,079,331	1,686	0	0
4	1,348,131	180	0	0
X	23,542,271	1,955	0	0
Y	3,667,352	717	0	0
dmel_mitochondrion_genome	19,517	0	0	0
Total	137,567,477	2,019,464	2,245	3,893

Figure 3.12: *An overview of the report.*

3.2.4 Export and compatibility

It is possible to convert the tracks generated by the current version of the plugin to the previous format (annotated sequence list) using the Utility Tools | Tracks | Track Conversion | Convert from Tracks tool. We recommend that this conversion be performed prior to exporting results in Genbank or GTF format. Export to GFF3 format does not require conversion.

3.3 The Transcript Discovery algorithm

The Transcript discovery tool takes Large Gap Read Mappings tracks and optionally gene and transcript annotations as input, and produces gene and transcript track annotations as output. The annotations are generated by examining the read mapping and identifying likely regions of genes, their exons and splice sites, and, for each gene region, a set of transcript annotations that explain the observed exons and splice sites in this region.

Events Alignments from a read mapping are by default filtered away if they are multi-matches, or duplicates. The definition of a duplicate match is that the start and end positions of the match and its sequence are identical to a previous match. Each non-filtered alignment in a read mapping is then converted to one or two events.

An event can be defined as information about the location of splice junctions obtained from e.g. a single read, two overlapping reads, or a previously annotated transcripts when annotation tracks are supplied during the tool set up. Events can be described as:

- **Known** when they come from previously annotated transcripts.
- **Spliced** when they contain a splice junction.
- **Unspliced** when they do not contain a splice junction.

For short read sequencing data, events are most commonly unspliced, in which case they will have a single interval region with start and end positions. For these unspliced events, the start and end position are only upper and lower bounds on the location of a splice junction.

In case a spliced event is found, all spliced reads are realigned around the splice site. The realignment realigns 15bp of read before and after the splice junction (2 x 15 = 30bp in total) against a 60bp region of the reference. Note that the algorithm:

- uses affine gap costs on the read (-5 open, 0 extension) and linear gap costs on the reference. This is because it expects a splice junction to lead to a large deletion in the read when compared to the reference.
- is global, not local. This is because the full alignment extends either side of the snippet that is being realigned, and so end gaps should be penalized.
- finds all equally high-scoring alignments, then filters these to keep only those with maximal intron size, and filters again to keep only those with intron size of at least 15bp.

After realignment, there may be 0, 1 or many possible ways of placing the splice junction:

- If 0 places are found, the original alignment is used, and the realignment is discarded.
- If 1 or more places are found, the tool picks the most likely junction from a prioritized list of canonical splice signatures (different places typically have different signatures). Ties are resolved by picking the first equally good junction. If no junction matches a signature, the splice junction still gets called but is given a range of possible positions for the start and end.

The strandedness of the transcript (whether it is forward or reverse on the reference genome) that a given event supports can be determined from the splice signatures of the event. This allows the noise to be filtered by discarding events that support a mixture of plus and minus strands. For example, two splice junctions from a single-end read may disagree on the strand, or the splice junctions of each read in a pair may disagree.

There is special handling for non-overlapping paired end reads. In these cases each read is separately turned into an event as described above and information about the strand is shared between the two events. If one event supports the plus strand, but the other does not allow the strand to be determined, then we infer that the other is also on the plus strand. If the two events support different strands, both are discarded.

Regions The parameter "Gene merging distance" defines regions in the mappings that contain events close enough to belong to the same gene. Regions of coverage are identified by iterating through events while calculating the distance between the current and the next event. If this is less than the "Gene merging distance" value, the current region is extended. If not, a new region is started.

First round of merging events The algorithm merges events that unequivocally support the same splice sites. Two events are strictly overlapping if

1. the events overlap
2. all introns and exons of the events in the overlapping region are the same
3. the non-overlapping parts of the events do not extend across any splice site positions of any other events in the coverage region. This last requirement ensures that we do not merge an event with another event in cases where there are more events supporting different splice sites with which it could be merged.

Two events can only be merged if they are on the same strand, share exactly the same splice boundaries, and disagree with all other events about the location of exactly the same splice boundaries. This also guarantees that it is not possible to merge a spliced event with an unspliced event. The resulting merged event has a region that is the union of the two input events. Events cannot be merged if they overlap different splice boundaries on unknown strand.

Correct end- and splice-points Many events will have end regions that are slightly off, mostly because the first few bases of the intron and the start of the following exon are identical (leading to exonic sequence being accidentally mapped to the intron), or because the events really should have been mapped with a gap but the read was too short on one side of the gap for the mapper to discover this. To correct this we identify start and end open positions whose bounds might be due to over-alignment. For example, if an exon starts at position <30, but a splice site is known at position 35, then the exon start will be corrected to <35. Correction is only applied to open positions within 8bp of a splice junction whose position is known exactly. Additionally, events with uncertain positions (available only if the option "Ignore reads with unknown splice signatures" is checked off) are corrected if their range of uncertainty overlaps a splice junction whose position is known exactly.

Regions on a minus strand are preferentially corrected by known splice sites on that strand. They are never corrected by splice sites on the plus strand. A similar statement holds for the plus strand. Regions whose strand is unknown are corrected twice: once assuming they are on a plus strand and once assuming they are on minus strand. The correction with most evidence is returned on the given strand.

Correction is careful to distinguish between the splice junction boundaries at the start and end of an intron. For example, if an exon starts at position <30, it can be corrected if an intron spans the range (35,50), but not if the intron spans the range (0,35). This is because the first case could arise due to over-alignment, but the second case cannot. Similarly, uncertain positions (only defined by a range) may only be corrected by splice junction boundaries that are at the same side of an intron as they are, or if their range of uncertainty overlaps a splice junction whose position is known exactly.

Merge/Filter cascade The fixing of end and splice points above alters the set of events within a coverage region, and typically causes events that could not be merged previously to now be mergeable. To further reduce the number of events, we carry out a second merging of events that unequivocally support the same splice sites.

Afterwards, a series of hard filters is applied to events. The order of the filters is important, because after one filter has been applied, events may become mergeable, and the merged event may then pass a filter that the original events would fail.

Filters and merges happen in the following order:

1. Ignore chimeric reads
2. Ignore reads with unknown splice signatures
3. Merge
4. Minimum spliced reads
5. Merge
6. Assign to strand: Unspliced events with undefined strand are assigned to the plus or minus strand if events on that strand overlap in their exons, and do not overlap in their introns. If an event can be assigned to both plus and minus, it is kept as undefined.
7. Merge
8. Minimum unspliced reads. Note that since we have just performed a merge, removing these unspliced events does not result in changes that allow new merges to be made according to the merge requirements (3).

Split in genes If gene annotations have been provided, they are used for the first time at this point in the process: Known Events (i.e., events made from previously annotated transcripts) are assigned to regions that correspond to the "known" gene to which they belong. These known genes/regions ultimately predict the genes to which the unknown events belong.

- If there is just one possible gene for an event, it is assigned to that gene.
- An event that cannot be assigned to a single gene is assigned to a list of possible genes to which it might belong. If the event is stranded, this list contains all of the known genes in the region that have the same strand as the event, and whose region intersects with the event. If the event is un-stranded, the list contains all genes whose region intersects with the event.
- If there are more possible events, we prefer a gene that completely overlaps with the event. If there are multiple such genes, we prefer the shortest.
- If there are no possible genes for an event, but it intersects with an event that has been assigned to a gene and that is not on a different strand, it is assigned to the gene of that event.

If, after this process, there are events left that have not been assigned to a gene, then they are assigned to new genes as follows:

1. Assigning events with an undefined strand to a strand:
    - If all the events of known strand within the minimum distance between genes are on either plus or minus, then that strand is used.
    - If both plus or minus strand events lie within the minimum distance, the event is assigned to the strand with the event of most similar coverage to itself.

- if there are no nearby event of known strand, but this event lies between the first and last events on the plus strand, and does not lie between the first and last events on the minus strand, then it is assigned to the plus strand (and vice versa).
- Otherwise the event will be assigned to the strand supported by the most reads, or the minus strand in the event of a tie.

2. Splitting events by strand and coverage: The events are split into two regions, one for each strand. Each of these is again split up if possible.

A second filter cascade A series of hard filters is again applied to events. Three of these remove events based on the average coverage of the regions that contain them.

1. Minimum spliced coverage ratio (default set at 0.05)
2. Merge
3. Minimum unspliced coverage ratio (default set at 0.05)
4. Filtering of unspliced events that are low coverage (defined as <0.25 x maximum coverage of any event in their region) and that lie strictly before/after the first/last spliced event in that region. By strictly before/after, we mean that the event being filtered ends before/starts after the spliced event.
5. Removal of unspliced events that do not overlap with the exons of any spliced event, and lie completely within the intron of one or more spliced events.

From events and regions to genes Events are joined in a so-called 'Splice Graph' where each path through the graph combines a series of events into a possible isoform. The algorithm enumerates up to 10 000 possible isoforms from each graph, which usually exhausts all possibilities. The EM algorithm from the RNA-Seq Analysis tool is then used to determine the set of isoforms and their abundances that best explain the observed reads (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=_EM_estimation_algorithm.html).

Now that events have been associated to genes, genes (and their associated transcripts) are filtered according the following configurable criteria:

- Minimum predicted gene length (default 250bp)
- Minimum reads in gene (default 10)
- Ignore genes that do not have spliced reads (on by default)

Transcripts with abundance that is not greater than 0.05 are additionally removed as likely noise. The abundances are not renormalized, to be able to distinguish between genes that have a few transcripts with high abundance, and genes with many transcripts with lower abundance and a lot of other noise transcripts. The abundance is reported as the relative confidence, see section 3.2.1.

The Transcript Discovery tool returns a gene track that contains all "known" genes (i.e. those supplied as input) plus any "unknown" genes (i.e. those that we have predicted). Similarly the transcript track contains all the "known" input transcripts plus any "unknown" transcripts.

Chapter 4

Install and uninstall plugins

Transcript Discovery plugin is installed as a plugin.

4.1 Installation of plugins

Note: In order to install plugins and modules, the *CLC Workbench* must be run in administrator mode. On Windows, you can do this by right-clicking the program shortcut and choosing "Run as Administrator". On Linux and Mac, it means you must launch the program such that it is run by an administrative user.

Plugins and modules are installed and uninstalled using the Workbench Plugin Manager. To open the Plugin Manager, click on the **Plugins ( ) button** in the top Toolbar, or go to the menu option:

Utilities | **Manage Plugins... ( )**

The Plugin Manager has two tabs at the top:

- **Manage Plugins** An overview of your installed plugins and modules is provided under this tab.
- **Download Plugins** Plugins and modules available to download and install are listed in this tab.

To install a plugin, click on the **Download Plugins** tab (figure 4.1). Select a plugin. Information about it will be shown in the right hand panel. Click on the **Download and Install** button to install the plugin.

Accepting the license agreement

The End User License Agreement (EULA) must be read and accepted as part of the installation process. Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

Installing a cpa file

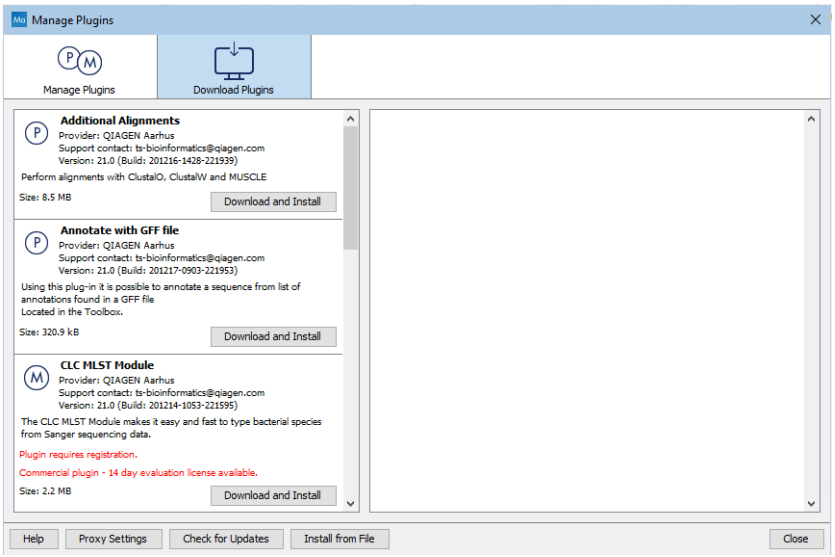


Figure 4.1: *Plugins and modules available for installation are listed in the Plugin Manager under the Download Plugins tab.*

If you have a .cpa installer file for Transcript Discovery, you can install it by clicking on the **Install from File** button at the bottom of the Plugin Manager.

If you are working on a system not connected to the internet, plugin and module .cpa files can be downloaded from `https://digitalinsights.qiagen.com/products-overview/plugins/`using a networked machine, and then transferred to the non-networked machine for installation.

Restart to complete the installation

Newly installed plugins and modules will be available for use after restarting the software. When you close the Plugin Manager, a dialog appears offering the opportunity to restart the *CLC Workbench*.

4.2 Uninstalling plugins

Plugins and modules are uninstalled using the Workbench Plugin Manager.To open the Plugin Manager, click on the **Plugins ( ) button** in the top Toolbar, or go to the menu option:

Utilities | **Manage Plugins... ( )**

This will open the Plugin Manager (figure 4.2). Installed plugins and modules are shown under the Manage Plugins tab of the Plugins Manager.

To uninstall a plugin or module, click on its entry in the list, and click on the **Uninstall** button.

Plugins and modules are not uninstalled until the Workbench is restarted. When you close the Plugin Manager, a dialog appears offering the opportunity to restart the *CLC Workbench*.

Disabling a plugin without uninstalling it

If you do not want a plugin to be loaded the next time you start the Workbench, select it in the

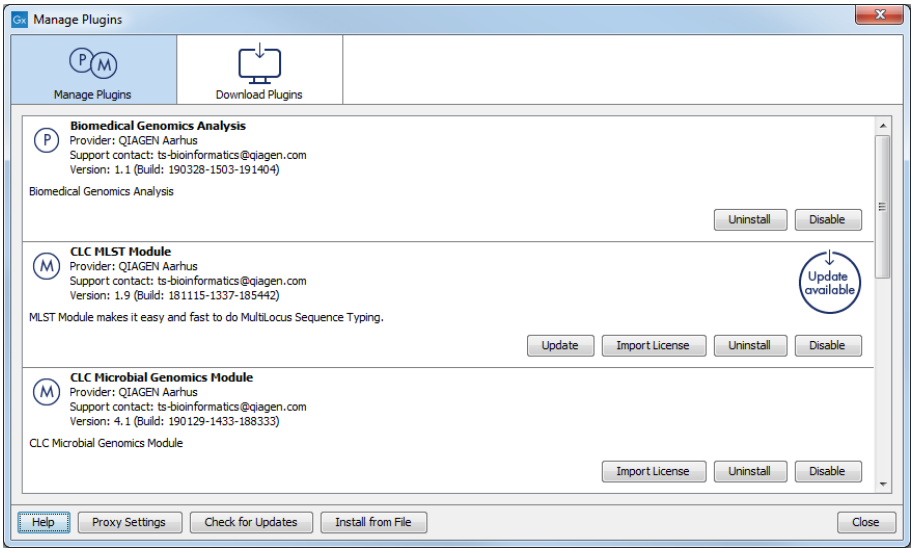


Figure 4.2: *Installed plugins and modules are listed in the Plugins Manager under the Manage Plugins tab.*

list under the Manage Plugins tab and click on the **Disable** button.

Bibliography

[Hayer et al., 2015] Hayer, K. E., Pizarro, A., Lahens, N. F., Hogenesch, J. B., and Grant, G. R. (2015). Benchmark analysis of algorithms for determining and quantifying full-length mrna splice forms from rna-seq data. *Bioinformatics*, 31(24):3938–45. https://academic.oup.com/bioinformatics/article/31/24/3938/197198.