sRNAmapper

version 1.0.5

- Documentation -

1.1 Scope

sRNAmapper is specifically designed to map small RNA sequences to genomes. To this end it uses a specialized mapping algorithm that requires a perfect 5' seed match (default: 18 nt) and optionally allows non-template 3' end as well as internal mismatches in the part of the sequence that follow the seed region. Allowing non-template 3' ends will ensure the mapping of 3' modified (adenylated/uridylated) small RNAs while allowing internal mismatches can enhance sensitivity considering degressive read quality towards 3' ends. The latest sRNAmapper version can be found at http://www.smallrnagroup.uni-mainz.de/software.html.

1.2 Changes to previous versions

1.0 → 1.0.2 04. July 2016: Resolved problems when using soft-masked genomes due to case sensitivity of the string mapping algorithm.

1.0.2 → 1.0.3 21. November 2016: Bit FLAG in SAM output was not correctly set to 16 when the sequence mapped to the opposite strand.

1.0.3 → 1.0.4 22. March 2017: Bit FLAG in SAM output was not correctly set to 16 when the sequence mapped to the opposite strand. Previous version did not fix this bug.

1.0.4 → 1.0.5 OK, finally we think we have fixed all problems with SAM output format.

2. Getting started

Running sRNAmapper on your local machine requires the installation of a Perl interpreter. Perl is pre-installed on common Linux and Mac systems. For Windows you can download and install either StrawberryPerl (www.strawberryperl.com) or ActivePerl (www.activestate.com/activeperl/downloads). For the mapping process you need a reference (genome file) and a file that contains your small RNA sequence reads in FASTA or FASTQ format. Start sRNAmapper from the command line or terminal with the following command:

```
perl sRNAmapper.pl -input piRNAs.fasta -genome genome.fasta -format sam -alignments best
```

With the above command sRNAmapper will create a map file in SAM format that is named piRNAs.fasta.map (you can give it a different name using the option -output othername.map):

```
@HD     VN:1.5  SO:coordinate
@SQ     SN:Chr1 LN:167399201
seq1    0   Chr1    1     255   23=          7    *   0   23   CAGGTGGATCATGAGGTCAGGAG          *
seq2    0   Chr1    25    255   25=1X1=      1    *   0   27   TCAAGACCAGCCTGGCCAACATGGTAA      *
seq3    0   Chr1    506   255   25=          2    *   0   25   GCCTGGGCAACAGCATGAGACTTGG        *
seq4    0   Chr1    557   255   22=1X5=1X    3    *   0   29   AAAAAAAAAAATTCAGCTAGCATTCTTGT    *
seq5    0   Chr1    927   255   18=          10   *   0   18   GAGTTTTACCATCTTGGC               *
```

A detailed description of the SAM format can be found here. Alternative formats are ELAND (use this as input for proTRAC; Columns refer to chromosome, starting coordinate of the mapped sequence, locus sequence, sequence name, sequence, mismatch, strand.):

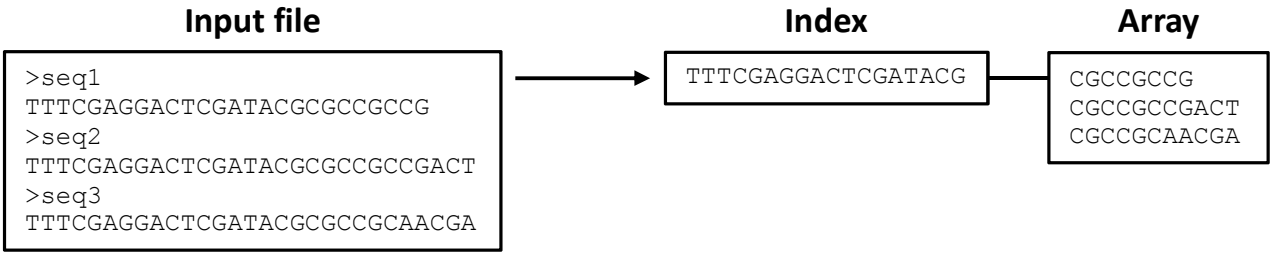
```
Chr1    1    CAGGTGGATCATGAGGTCAGGAG          seq1    CAGGTGGATCATGAGGTCAGGAG          0    +
Chr1    25   TCAAGACCAGCCTGGCCAACATGGTGA      seq2    TCAAGACCAGCCTGGCCAACATGGTAA      1    +
Chr1    506  GCCTGGGCAACAGCATGAGACTTGG        seq3    GCCTGGGCAACAGCATGAGACTTGG        0    +
Chr1    557  AAAAAAAAAAATTCAGCTAGCAATCTTGA    seq4    AAAAAAAAAAATTCAGCTAGCATTCTTGT    2    +
Chr1    927  GAGTTTTACCATCTTGGC               seq5    GAGTTTTACCATCTTGGC               0    +
```

and COMPACT (Columns refer to chromosome, starting coordinate of the mapped sequence, strand, sequence length, string that specifies mismatch to genome [position in sequence, nucleotide in sequence. Will be - if there is no mismatch], sequence name.):

```
chr1    1       +    23  -       seq1
chr1    25      +    27  26A     seq2
chr1    506     +    25  -       seq3
chr1    557     +    29  23T29T  seq4
chr1    927     +    18  -       seq5
```

3. How `sRNAmapper` works

During the mapping process `sRNAmapper` initially searches for perfect 5' matches (referred to as seed matches) of a defined length (default: 18 nt). Therefore, `sRNAmapper` generates indexes corresponding to nucleotides 1-`seedmatch` that are linked to an array that contains all the variants of 3' ends for the given index.



After indexing the reads from the input file `sRNAmapper` scans the genome file for perfect seed matches. Once a perfect seed match is found it starts to align the corresponding 3' ends from the array linked to the index. The alignment ends with the annotation of a valid hit or aborts in case of exceeding the maximum number of allowed mismatches (internal mismatches [`-mismatch`], non-template 3' nucleotides [`-nontemplates`]). Below you see an example of a valid alignment using the default parameters (`-seedmatch 18 -mismatch 1 -nontemplates 2`):

```
genome  5'-TGCGTATTAGCTCGCATGACTCGCATAGCTACGTGGTAGC-3'
                ||||||||||||||||||x||||||xx
sRNA          5'-TTAGCTCGCATGACTCGCTTAGCTAAA-3'
                    S E E D
```

Before sRNAmapper creates the final output file the hits will be written to a `RAW` file. This `RAW` file is then processed to the final `MAP` file. There are two reasons for doing this: i) The hits in the final sRNAmapper output files are sorted according to genome coordinates. Initially there might occur deviations from the correct order that result from nearby hits on different strands. ii) When using the option `-alignments best` sRNAmapper will output only the best alignments (in terms of mismatch counts) for one sequence to the final output file. Therefore it writes hits to the `RAW` file only if the alignment is of equal or better quality than previous alignments of the same sequence. It is likely that for some sequences the best alignment is not the first alignment. These bad alignments present in the `RAW` file will be removed in the final `MAP` file.

4. Command line options

-a -alignments	Must be `<all>` or `<best>`. With `<all>` all valid alignments will be written to the final output file. With `<best>` only the best alignments (in terms of mismatch counts) for one sequence will be written to the final output file. Default is `<all>`.
-d -direction	Defines if mapping is performed on the plus strand only (`<0>`) or on both strands (`<1>`). Default is `<1>`.
-f -format	Must be `<sam>`, `<eland>` or `<compact>`. For a description of the different formats see examples above. For

	creating proTRAC input files use `<eland>`. Default is `<eland>`.
-g -genome	Specifies the reference/genome file in FASTA format.
-h -help	Will print some information to `stdout`.
-i -input	Specifies the input file that contains your sequence reads in FASTA or FASTQ format.
-m -mismatch	Defines the maximum number of internal mismatches in the part of the sequence that follows the seed match. Default is `<1>`.
-n -nontemplates	Defines the maximum number of non-template 3' nucleotides. Default is `<2>`.
-o -output	Name of the output file. If no name is specified the output file will have the name `input.fasta.map` (input file: `input.fasta`).
-r -replace	Titles from the reference/genome file can optionally be replaced by a consecutively numbering to reduce output size and save diskspace. Valid values are `<0>` (keep original names) and `<1>` (replace names by numbers). Default is `<0>`.
-s -seedmatch	Defines the number of 5' nucleotides that must produce a perfect hit (seed match). Default is `<18>`.
-v -verbosity	Defines verbosity of the program during the mapping process. Valid values are `<0>` (silent), `<1>` (concise) and `<2>` (verbose). Default is `<2>`.

6. Contact

If you have any questions or comments or found any bugs in the software please do not hesitate to contact:

David Rosenkranz
Institute of Organismic and Molecular Evolution
Anthropology, small RNA group
Johannes Gutenberg University Mainz, Germany
Email: rosenkranz@uni-mainz.de
Web: http://www.smallRNAgroup.uni-mainz.de

7. Citation

If you use the sRNAmapper software for your publication please cite one of the following papers:

- Roovers EF, Rosenkranz D, Mahdipour M, Han CT, He N, Chuva de Sousa Lopes SM, van der Westerlaken LAJ, Zischler H, Butter F, Roelen BAJ and Ketting RF. Piwi proteins and piRNAs in mammalian oocytes and early embryos. *Cell Rep* 2015 10:2069-2082.
- Rosenkranz D, Han CT, Roovers EF, Zischler H, Ketting RF. Piwi proteins and piRNAs in mammalian oocytes and early embryos: From sample to sequence. *Genomics Data* 2015 5:309-313.

The latter one is a detailed Methods paper connected to the original publication.