RNArobo 2.1.0 – Quick Start Guide

September 8, 2013

RNArobo is a fast RNA structural motif search tool. RNArobo can search sequence databases in FASTA format for a motif defined by a "descriptor", which can specify primary and secondary structure constraints.

The format of an RNArobo descriptor is an extension of the descriptor format used by RNABob (a search tool by Eddy [1996]), thus RNABob descriptors are compatible with RNArobo. A descriptor consists of three parts:

1. a **motif map** – a list of individual *structural elements* ordered from 5' to 3' end along the sequence
2. a detailed **specification** of each structural element
3. an optional *search order*

Each structural element is either single stranded (denoted by `s`) or helical (denoted by `h` or `r`). Detailed specification of each element consists of (the fields in bold are **mandatory**, while fields in italic are *optional*):

1. number of **mismatches** allowed (in helical elements mismatches are allowed only in the positive strand)

(1b.) number of **mispairs** allowed (for helical elements only)

2. number of *insertions* allowed
3. **primary sequence** constraints: a string composed of IUPAC codes and wild cards "`*`" that allow to match any character or to be skipped; alternatively, an abbreviation for e.g. 10 wild cards can be written as "`[10]`"

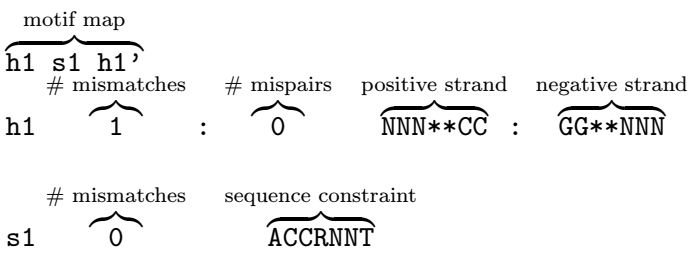
(3b.) primary **sequence constraints** for the **negative strand** of a helical element; in helical elements wild cards can occur only in pairs, i.e. for every wild card there must be a corresponding wild card in the other strand at the exactly opposite position

4. IUPAC code for *allowed insertions*

(5.) a **transformation** string specifying pairings allowed in the *relational* element `r`

Example Descriptors

See the following simple motif composed of two elements – a helix `h1` and a single strand `s1`:



Unlike RNABob, RNArobo enables you to allow nucleotide insertions in a structural element. Syntax of allowing insertions is similar to specification of the maximal number of mismatches (or mispairs). You can specify the maximal number of insertions and what nucleotides are allowed as an insertion. To specify the nucleotide constraints, you can use IUPAC code as for any other primary sequence constraints. Insertions are not allowed at the very beginning and end of the matched regions and in a helix insertions must not be adjacent nor opposite. Usage should be clear from the example descriptor:

```
h1 s1 h1'
h1 0:0:2  NNN**CC:GG**NNN:A
s1 0:1    ACCRNNT:Y
```

In the helix we allow up to 2 insertions of Adenosine, while in the single strand only one insertion of a pyrimidine nucleotide is allowed ('`Y`' stands for Cytosine or Thymine/Uracil).

Note, RNArobo doesn't discriminate Thymine and Uracil, you can use them interchangeably in both the descriptor and searched FASTA sequence.

To specify custom pairing function for a helical element, you can use an *relational* element instead for standard helix, e.g.:

```
r1 s1 r1'
r1 0:0:2  NNN**CC:GG**NNN:A  TGCA
s1 0:1    ACCRNNT:Y
R s1 h1
```

This variation of the previous descriptor allows only canonical base-pairs `A-T` and `C-G` in the relational element `r1`. The individual IUPAC codes in the *transformation* string `TGCA` define nucleotides that can pair with `A`, `C`, `G`, and `T`, respectively in this order. For default helical elements (e.g. `h1`) RNArobo allows also Watson-Crick base pairs, as the default *transformation* string is `TGYR`.

(**Optional**) The last line of the example descriptor above illustrates usage of an optional reorder command which specifies the order in which elements are internally searched by the RNArobo algorithm, similarly to RNAMot [Gautheret et al., 1990]. If this command is absent or does not contain all elements, an automatic data-driven method is used to determine the best possible ordering of all remaining elements. This command has no principal impact on the actual results of the search, but defining a previously trained order can speed up the search by few seconds.

Installation / Usage

To run RNArobo on your system, you need GCC C++ compiler (tested with version 4.4.5) or for 64-bit Linux systems we directly provide an executable binary.

1. **Download** the most recent version of RNArobo at `http://compbio.fmph.uniba.sk/rnarobo`. There you can download the executable binary for 64-bit Linux systems as well as the source code package.

2a. If you are going to use the provided binary, set the "**executable bit**" by command:

```
chmod a+x rnarobo-2.1.0-linux64
```

Now you are ready to run RNArobo. In what follows we refer to the binary as "rnarobo", so please substitute "rnarobo-2.1.0-linux64" for "rnarobo" where applicable.

2b. (Recommended) If you cannot use the provided binary, you have to **compile RNArobo** from the source code on your own. For this step you need to have GCC compiler installed. First unpack the downloaded package, than go to the unpacked directory and execute "make" command.

```
tar -zxf rnarobo-2.1.0.tar.gz
cd rnarobo-2.1.0/
make
```

By now, you should have an executable file called "rnarobo".

(**Optional**) To install rnarobo to be available for every user and from every directory, execute "sudo make install" command (you will need to enter the superuser password) that will copy the binary file to `/usr/local/bin/` .

3. Run RNArobo by the command:

```
./rnarobo [options] <descriptor-file> <sequence-file>
```

where "<descriptor-file>" is the path to the descriptor file and "<sequence-file>" is the path to the sequence database in FASTA format. If rnarobo is properly installed, you can run it from every directory by the same command, but without the "./" prefix.

If you also want to **search in the complementary strands** and **show only non-overlapping matches** of the sequences, run RNArobo with "-c" and "-u" flags, e.g.:

```
./rnarobo  -cu  motif.des  db.fa  > occurrences.txt
```

Output of an RNArobo run is printed on the standard output and consists of a header and of a list of found matches. Matches in the list are in the order as they were found in the database file from its beginning to its end. Every match is compounded of two lines. The first line gives the name and description (if any) of the sequence where this match occurs, the beginning position where the match starts in the sequence and the ending position where the match ends. This line is followed by a line containing the match itself, that is, the substring of the sequence defined by the starting and ending positions. A symbol of pipe "|" delimits individual elements of the match.

Available RNArobo Options:

-c search both strands of the database

-u report only non-overlapping occurrences

-f print output in plain FASTA format

-s print output in FASTA format with element separators

--nratio FLOAT set max allowed ratio of "N"s in reported occurrences to their length; must be within $\langle 0, 1 \rangle$

Advanced Options

To override default search order training parameters (not recommended though):

--k INT set length of tuples used in training

--limit INT set max size of training set (max number of tuples)

--alpha FLOAT set significance level for Welch's t-test, must be: 0.2, 0.1, 0.05, 0.025 or 0.01

--iterative BOOL iteratively train whole the ordering TRUE / FALSE

--tonly perform only the order training itself

The defaults are: --k 3 --limit 50 --alpha 0.01 --iterative TRUE

References

S.R. Eddy. RNABob: a program to search for RNA secondary structure motifs in sequence databases. unpublished, 1996.

D. Gautheret, F. Major, and R. Cedergren. Pattern searching/alignment with RNA primary and secondary structures: an effective descriptor for tRNA. *Comput Appl Biosci*, 6(4):325–31, 1990. ISSN 0266-7061.