9 December 2003

QRNA-2.0.2c documentation

Elena Rivas and Sean R. Eddy

{elena,eddy}@genetics.wustl.edu.

Howard Hughes Medical Institute
Department of Genetics
Washington University, St. Louis, MO 63110, USA

Abstract

Basic how–to guide to install and run QRNA-2.0.2c.

QRNA-2.0.2c is a *very early* prototype of a new QRNA generation that uses probabilistic evolutionary models to be able to tune all the parameters (transition and emission probabilities) of the three QRNA models to any possible degree of sequence divergence.

Install QRNA

```
Source distribution (qrna-2.0.2c.tar.Z):

    setenv QRNADB  (HOME)/qrna-2.0.2c/lib  assign a location for the libraries.

    tar zxf qrna-2.0.2c.tar.Z       Unpacks the archive. (makes a new directory, qrna-2.0.2c).
    cd qrna-2.0.2c                  Moves into the distribution toplevel directory.
    cd squid                        Moves into the directory squid.
    make                            Builds the binaries for the squid library.
    cd ..                           Goes back to toplevel directory.
    cd src                          Moves into the source directory.
    make                            Builds the binaries.
```

It should build cleanly on just about any UNIX machine.

The directory **qrna-2.0.2c** includes the following subdirectories,

Demos A collection of files to demonstrate how to use QRNA.

documentation Contains a userguide manual.

lib Contains all the additional files used as input by QRNA. This directory is accessed by setting the environment variable QRNADB.

Licenses The license for this software.

scripts The wrap around Perl scripts used with QRNA.

squid An old version of Sean Eddy's squid library for sequence handling.

src The source files, and headers. It also includes a Makefile. After building QRNA the executable is located in this directory.

Create a QRNA input file from a blast output

- Start with a WUBLASTN output file **foo.blast**.
  Example, the file in the 'qrna-2.0.2c/Demos/" directory **HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast**, which is the result of blasting a collection of 850 annotated Human RNA genes against the mouse genome.
- Then you need to run the Perl scripts **qrna-2.0.2c/scripts/blastn2qrnadepth.pl**.

```
usage: blastn2qrnadepth.pl [options] <blastfile>
options:
-d <depth>     :  max number of alignment coverage per position  [ default depth = 1       ]
-e <max_eval>  :  maximum evalue   of blast hits allowed         [ default max_eval = 0.01 ]
-g <org_name>  :  name of the blasting organism                  [ default 'org'           ]
-i <min_id>    :  minimum identity of blast hits allowed         [ default min_id = 0      ]
-j <max_id>    :  maximum identity of blast hits allowed         [ default max_id = 100    ]
-l <min_len>   :  minimum length   of blast hits allowed         [ default min_len = 1     ]
-o <outfile>   :  output qfile                                   [ default = 'blastfile.q' ]
-r             :  calculate depth respect to the database instead of the query
-s <shift>     :  position shift when calculating depth          [ default shift = 1       ]
-w <which>     :  criteria to pick alignments                    [ default which = 'SC'    ]
                                  ID -- best % identity
                                  SC -- best score
-x <name>          : ignore given name, use this one for gff outputs
```

- "depth" is a parameter to limit the number of blast alignments that include a given position. If you do not want to constrain the number of blast alignments this way, set "depth" to a very large value. Limiting the depth is important for eukariotic genomes with large amount of repetitive sequences. The default depth = 1 will give the best alignment per position. If you want to allow some redundancy a depth of 5 or 10 is recommended.
- "max_eval" removes blastn alignments with E values above this cutoff.
- "org_name" is the name of the organism as it will appear in the gff file
- "min_id" is the minimum identity of blastn alignments allowed.
- "max_id" is the maximum identity of blastn alignments allowed.
- "min_len" is the minimum length of blastn alignments allowed.
- "shift" is a parameter related to "depth". It indicates how many nucleotides to skip after every check for depth. I recommend not to change it from its default value.
- "which" has to values "SC" or "ID". It determines the criteria to pick the best alignments when applying the depth constraint.

Notice: Even if you do not want to prune **foo.blast**, you have to run this script.

- The result of running
qrna-2.0.2c/scripts/blastn2qrnadepth.pl foo.blast
is the three files:
  - **foo.blast.E<eval>.D<depth>.q** is the input file taken by QRNA.
  It is a collection of sequences in fasta format, where two consecutive sequences are the two component of an alignment with the gaps left in place.
  - **foo.blast.E<eval>.D<depth>.q.rep** is a report of the BLASTN alignment that have been pruned in the process of creating **foo.blast.E<eval>.D<depth>.q** according to the options used in **blastn2qrnadepth.pl**.
  - **foo.blast.E<eval>.D<depth>.q.gff** is the gff format version of the surviving aligned regions of the query genome.

- The result of running
qrna-2.0.2c/scripts/blastn2qrnadepth.pl -g human HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast
is the three files in the directory "qrna-2.0.2c/Demos/":
HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q
HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q.rep
HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q.gff
Looking at the .rep file, we observe that initially the .blast file had $69,553$ alignments. And that after using the default filters, we end up with 867 alignments approximately one alignment per RNA gene which is reasonable since by default we have assumed depth = 1.

Run QRNA

- QRNA is a collection of programs, written in C.
Source code is in "qrna-2.0.2c/src/". The executable is called "qrna".
- The "makefile" at "qrna-2.0.2c/src/" can be modified to change the C compiler or to change compilation flags.
- To run qrna :
qrna-2.0.2c/src/qrna [options] foo.blast.q
- Options are:

```
qrna -- scores an alignment with the 3 models
          2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
sage: qrna [-options] <input_file.q>
where options are:
   -A              : do an all-to-all comparison between the two input files
   -a              : print alignment
   -B              : sre_shuffle the alignment keeping the gap structure of the window
   -b              : shuffle the alignment
   -c <cfgfile>    : <cfgfile> to use to train the rna model (default = tRNA+rRNA)
   -D <codfile>    : include a file of coding-coding frequencies for the coding model
   -d              : log2 form (default = log2-odds space )
   -e <num>        : number of sequen skipped in second file (for multiple comparisons). default 0.
   -F              : change the overall base composition of the 3 models, based on nts frequencies in the
   -f              : use full dp for the probabilistic models (do not conserve the aligment-default is di
   -G              : change the overall base composition of the 3 models, based on nts frequencies for ea
   -g              : do forward (default is viterbi)
   -H <Hexfile>    : include a file of Hexamer frequencies for the coding model
   -h              : print short help and usage info
   -i              : evolutionary time factor (default i=1)
   -j              : use semi-full dp for the probabilistic model (use the alignment created by OTH)
   -k              : allow pseudoknots (not implemented)
   -l <minlenhit>  : change the minlenhit parameter (default 0)
   -L <maxlenhit>  : change the maxlenhit parameter (default provided by longuest sequence)
   -m              : do Forward and Viterbi Diagonal dp
   -n              : do Forward and Viterbi Full     dp
   -o <outfile>    : direct structure-annotated sequence to <outfile>
   -p <pamfile>    : <pamfile> to use (default = BLOSUM62)
   -P              : pedantic, check your evolutionary models for inconsistencies
   -q              : do Forward and Viterbi semi-full dp
   -r              : do Nussinov rna model (default is a 3-state model)
   -R <ribofile>   : <ribofile> to use to train the rna model (default = RIBOPROB85-60)
   -s              : do global (not dp)
   -S              : sweep a collection of motifs(seqfile1) across another bunch of sequences(seqfile2)
   -t              : print traceback
   -v              : verbose debugging output
   -w <num>        : scanning window (default is full length)
   -x <num>        : slide positions (default is 50)
   -y <num>        : grab n sequences at random from the second data file to compare to each one of the f

   --cyk     : use CYK algorithm to calculate RNA score (default is Inside).
   --latte   : I just called starbucks with your order...
   --ones    : score with the three models only the given strand.
   --parse   : input is a selex file. Por a given ss, it calculates the probablity of either the best or
   --rnass   : print the alignment with the predicted RNA secondary structure.
   --noends  : do not evaluate ends. Default now: calculate the actual boundaries of the model calls
   --scan    : scanning version. no windows. faster.
   --shtoo   : qrna the alignment and also give one shuffled score.
   --twindow : select the divergence time based on the %id of the window. Default is by %id of alignment

 Debugging, experimentation:
   --regress <f> : save regression test information to file <f>
```

Some useful flags:

- For alignments that are too long, you can score them in chunks using -w ⟨windowsize⟩. The option -x ⟨slide⟩ to decide how many nucleotides to move before you score another chunk of the given alignment. Every window

analyzed starts with "length alignment:".

- Option -L determines the maximum length allowed for an alignment to be scored. The default is 1,000. I use this variable to control the memory usage of the program. If you are using the "window" version, memory is determined by the window so you can set -L as large as you want. If not using a window, I would not use a max length larger than $1,500$.

- There are three different scoring algorithms:

  global (-s)

  local Viterbi (default)

  local forward (-g).

  You can report any of them or any desired combination of them. I would recommend to use the default.

- -w <win> -x <slide>
  Scores an alignments in windows of length <win>, moving nucleotides <slide> each time.

- –scan -w <win> -x <slide>
  Same results as before but using a smarter algorithm that saves time. This version is more memory consuming, so do not use it for very large window sizes. This option is incompatible with "–rnass" and shuffling the alignment by windows.

- –rnass
  Calculates a secondary structure for the aligned sequences using the posterior probabilities of each of the possible pairs of positions being basepaired. Implements the Outside algorithm for the RNA grammar. It then uses an unambiguous grammar by Robin Dowell (unpublished) to use those posterior probabilities and generate the best structure.

Looking at a QRNA output:

For file "qrna-2.0.2c/Demos/5s_rRNA.q" which contains an alignment of two 5S rRNAs we have,

qrna-2.0.2c/src/qrna -a qrna-2.0.2c/Demos/5s_rRNA.q > qrna-2.0.2c/Demos/5s_rRNA.q.qrna2

The output looks like:

```
#-------------------------------------------------------------------------------
#      qrna 2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
#-------------------------------------------------------------------------------
#      PAM model =  BLOSUM62
#-------------------------------------------------------------------------------
#      RNA model       =  /mix_tied_linux.cfg
#      RIBOPROB matrix =  /RIBOPROB85-60.mat
#-------------------------------------------------------------------------------
#      seq file  =  ../Demos/5s_rRNA.q
#                   #seqs: 2 (max_len = 143)
#-------------------------------------------------------------------------------
#      full length version:  -- length range = [0,1000]
#-------------------------------------------------------------------------------
# 1  [both strands]
>RF00001.5S_rRNA.U26684/296-411 (143)
>RF00001.5S_rRNA.M35563/5-120 (143)

Divergence time (variable): 0.108746
[alignment ID = 82.05 MUT = 16.24 GAP = 1.71]

length alignment: 117 (id=82.05) (mut=16.24) (gap=1.71)
posX: 0-116 [0-115](116) -- (0.27 0.24 0.28 0.22)
posY: 0-116 [0-115](116) -- (0.23 0.29 0.29 0.18)
```

```
        RF00001.5S_rRNA CCTGATACCCATAGAGCTGTGGTACCACCTGAATCCATGCCGAACTCAGA
        RF00001.5S_rRNA CCTGGTGTCCATAGAGCACTGGAACCACCTGATCCCATCCCGAACTCAGA

        RF00001.5S_rRNA AGTGAAACGCAGCATCGCCGATGGTAGTGTGAG.GTCTCCTCATGTGAGA
        RF00001.5S_rRNA AGTGAAACGGTGCATCGCCGATGGTAGTGTG.GGGCCTCCCCATGTGAGA

        RF00001.5S_rRNA GTAGGACAGTATCAGGT
        RF00001.5S_rRNA GTAGGTCAACGCCAGGC

LOCAL_DIAG_VITERBI -- [Inside SCFG]
OTH ends  (+) =  (0..[117]..116)
OTH ends *(-) =  (0..[117]..116)
COD ends  (+) =  (80..[4]..83)
COD ends *(-) =  (48..[40]..87)
RNA ends  (+) =  (0..[117]..116)
RNA ends *(-) =  (0..[117]..116)
winner = RNA
               OTH =      245.320             COD =      246.502             RNA =      252.033
   logoddspostOTH =        0.000  logoddspostCOD =        1.182  logoddspostRNA =        6.713
     sigmoidalOTH =       -6.744    sigmoidalCOD =       -5.545    sigmoidalRNA =        5.004
```

- Every new blast alignment starts with two lines:
  ">Query_name"
  ">Subject_name"

- "Divergence time" indicates the particular time parameterization of QRNA used. By default QRNA decides on the divergence time (in this case $t = 0.109$) given the percentage identity of the alignment (82.05%) [Divergence time (variable)]. You can set a particular divergence time using the option "-i" [Divergence time (fixed)].

- For each model, and for each strand, you are given the actual local regions (they could be more than one per model and strand) that score according to the model. The notation is (from..[length]..to). Coordinates for *both* strands are given relative to the positive strand. The "*" indicates the strand with the strongest signal for a given model.

- For a given scoring algorithm you get three rows of numbers:

  row 1 The scores of the alignment under each of the three models. The "null" model is a forth model which assumes that the two sequences in the alignment are independent from each other.

$$\log \frac{P(data|OTH)}{P(data|null)} \qquad \log \frac{P(data|COD)}{P(data|null)} \qquad \log \frac{P(data|RNA)}{P(data|null)}$$

  row 2 The two (COD and RNA) log–odds posterior probabilities respect to the OTH model.

$$\log \frac{P(OTH|data)}{P(OTH|data)} \qquad \log \frac{P(COD|data)}{P(OTH|data)} \qquad \log \frac{P(RNA|data)}{P(OTH|data)}$$

  row 3 The row you should be paying the most attention to. The three sigmoidal scores calculated using the other two models as null models. The model with the highest sigmoidal score is the winner.

$$\begin{aligned}
\sigma_{\text{OTH}} &= \log \frac{P(data|OTH)P(OTH)}{P(data|COD)P(COD) + P(data|RNA)P(RNA)} \\
\sigma_{\text{COD}} &= \log \frac{P(data|COD)P(COD)}{P(data|OTH)P(OTH) + P(data|RNA)P(RNA)} \\
\sigma_{\text{RNA}} &= \log \frac{P(data|RNA)P(RNA)}{P(data|OTH)P(OTH) + P(data|COD)P(COD)}
\end{aligned}$$

These scores are called sigmoidal because for each model M_i

$$P(M_i|data) = \frac{e^{\sigma_{M_i}}}{1 + e^{\sigma_{M_i}}},$$

which is a sigmoidal function. This function has the nice property that

$$P(M_i|data) > 1/2 \Longleftrightarrow \sigma_{M_i} > 0.$$

We assume a flat prior for the three models.

- Option -a prints the scored alignment.
- Option -B scrambles the columns of the pairwise alignment. The result is an alignment with the same percentage identity, but without any column correlations. Compare the results of using -B for the alignment of two 5S rRNAs given before:

qrna-2.0.2c/src/qrna -a -B qrna-2.0.2c/Demos/5s_rRNA.q > qrna2-.0.0/Demos/5s_rRNA.q.shuffle.qrna2

```
#-------------------------------------------------------------------------------
#      qrna 2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
#-------------------------------------------------------------------------------
#      PAM model =  BLOSUM62
#-------------------------------------------------------------------------------
#      RNA model       =  /mix_tied_linux.cfg
#      RIBOPROB matrix =  /RIBOPROB85-60.mat
#-------------------------------------------------------------------------------
#      seq file  =  ../Demos/5s_rRNA.q
#                   #seqs: 2 (max_len = 143)
#-------------------------------------------------------------------------------
#      full length version:  -- length range = [0,1000]
#-------------------------------------------------------------------------------
# 1  [both strands] (sre_shuffled)
>RF00001.5S_rRNA.U26684/296-411 (143)
>RF00001.5S_rRNA.M35563/5-120 (143)

Divergence time (variable): 0.108746
[alignment ID = 82.05 MUT = 16.24 GAP = 1.71]

length alignment: 117 (id=82.05) (mut=16.24) (gap=1.71)(sre_shuffled)
posX: 0-116 [0-115](116) -- (0.27 0.24 0.28 0.22)
posY: 0-116 [0-115](116) -- (0.23 0.29 0.29 0.18)


        RF00001.5S_rRNA ATTATGCGTTAGGTAGACTGCGAGAAGCAGCCCAGTCTCTTCGTGTGCGG
        RF00001.5S_rRNA ACTATGCGTAAGGCAGACTGCGTGGAGGACCCCAGTCTCTTCGCGAGCGG

        RF00001.5S_rRNA CGACTCCGTCTCGCAGACGGGCGGTAACCACAA.TGACCCTAGAAATTAA
        RF00001.5S_rRNA CGACTCCGCCTTGCGGACGGGCGGCTACCAC.AGTGACCCTAGATATCAA

        RF00001.5S_rRNA GAACAAAGGTGTCGTTA
        RF00001.5S_rRNA CAACGAAGGTGTCATTA

LOCAL_DIAG_VITERBI -- [Inside SCFG]
OTH ends  (+) =  (0..[117]..116)
OTH ends *(-) =  (0..[117]..116)
COD ends *(+) =  (56..[28]..83)
COD ends  (-) =  (75..[25]..99)
```

```
RNA ends  (+) =  (0..[85]..84)
RNA ends *(-) =  (101..[16]..116)
winner = OTH
              OTH =      245.320             COD =      242.228             RNA =      236.968
   logoddspostOTH =        0.000  logoddspostCOD =       -3.093  logoddspostRNA =       -8.352
     sigmoidalOTH =        3.055    sigmoidalCOD =       -3.097    sigmoidalRNA =       -8.512
```

The alignment still has 82.05% identity as the original one, but now it gets classified as "other" since the RNA structure has been destroyed. We use the -B option to estimate false positives.

Example using the scanning version with a window:

Consider file "qrna-2.0.2c/Demos/Scerevisiae_orf_v_other_yeasts.q" which contains an alignment of a *S. cerevisiae* ORF The alignment has 514 nucleotides, and we would like to score it with QRNA using a window of 150 nucleotides, and moving the window 50 nucleotides each time.

qrna-2.0.2c/src/qrna -w 150 -x 50 qrna-2.0.2c/Demos/Scerevisiae_orf_v_other_yeasts.q >
Scerevisiae_orf_v_other_yeasts.q.W150.X50.qrna2

The output for the first two windows looks like:

```
#-------------------------------------------------------------------------------
#      qrna 2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
#-------------------------------------------------------------------------------
#      PAM model =  BLOSUM62
#-------------------------------------------------------------------------------
#      RNA model       =  /mix_tied_linux.cfg
#      RIBOPROB matrix =  /RIBOPROB85-60.mat
#-------------------------------------------------------------------------------
#      seq file  =  ../Demos/Scerevisiae_orf_v_other_yeasts.q
#                   #seqs: 2 (max_len = 545)
#-------------------------------------------------------------------------------
#      window version: window = 150   slide = 50 -- length range = [0,9999999]
#-------------------------------------------------------------------------------
# 1  [both strands]
>EFB1_I-142172-143158-80<621- (545)
>Contig1207-5-454>998- (545)

length of whole alignment after removing common gaps: 545
Divergence time (variable): 0.023271
[alignment ID = 92.66 MUT = 6.79 GAP = 0.55]

length alignment: 150 (id=96.67) (mut=3.33) (gap=0.00)
posX: 0-149 [0-149](150) -- (0.29 0.20 0.18 0.33)
posY: 0-149 [0-149](150) -- (0.27 0.21 0.20 0.32)
LOCAL_DIAG_VITERBI -- [Inside SCFG]
OTH ends *(+) =  (0..[150]..149)
OTH ends  (-) =  (0..[150]..149)
COD ends  (+) =  (78..[72]..149)
COD ends *(-) =  (6..[144]..149)
RNA ends *(+) =  (0..[28]..27)
RNA ends  (-) =  (0..[150]..149)
winner = COD
              OTH =      437.696             COD =      447.326             RNA =      430.253
   logoddspostOTH =        0.000  logoddspostCOD =        9.629  logoddspostRNA =       -7.444
```

```
     sigmoidalOTH =       -9.629    sigmoidalCOD =        9.621    sigmoidalRNA =      -17.075

length alignment: 150 (id=94.67) (mut=5.33) (gap=0.00)
posX: 50-199 [50-199](150) -- (0.27 0.25 0.18 0.30)
posY: 50-199 [50-199](150) -- (0.27 0.24 0.21 0.28)
LOCAL_DIAG_VITERBI -- [Inside SCFG]
OTH ends *(+) =  (50..[150]..199)
OTH ends  (-) =  (50..[150]..199)
COD ends  (+) =  (78..[72]..149)
COD ends *(-) =  (54..[144]..197)
RNA ends *(+) =  (50..[35]..84)
RNA ends  (-) =  (161..[39]..199)
winner = COD
              OTH =      416.760             COD =      425.202             RNA =      408.614
   logoddspostOTH =        0.000  logoddspostCOD =        8.442  logoddspostRNA =       -8.147
     sigmoidalOTH =       -8.442    sigmoidalCOD =        8.437    sigmoidalRNA =      -16.593
```

How to read the information for each analyzed window:

- Each new analyzed window starts with the line:

  ```
  length alignment:
  ```

- For each window and for each sequence in the alignment we have a line of the form:

  ```
  posX: 50-199 [50-199](150) -- (0.27 0.25 0.18 0.30)
  ```

  The first pair of number represent the first and last coordinates of the window respect to the beginning of the alignment. The pair of numbers in brackets represent the mapping of the window into the coordinate system of sequence X (after removing gaps). The adjacent number in parenthesis is the length of that segment in sequence X. Finally the four decimal numbers in parenthesis are the fraction of A, C, G, and T's in the segment of sequence X involved in that particular window.

Example starting with a blastn output:

File **"qrna-2.0.2c/Demos/HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast"** is a typical WUBLASTN output file. This file is the result of blasting a collection of 850 annotated Human ncRNAs to the mouse genome.

Typing:
qrna-2.0.2c/scripts/blastn2qrnadepth.pl -g human
qrna-2.0.2c/Demos/HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast
creates the file
"qrna-2.0.2c/Demos/HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q".
This file has selected those WUBLASTN alignment of any length, and any % identity, and E-value ≤ 0.01, and with depth = 1.
The two aligned sequence are now ready to be sent to QRNA.

Typing:
qrna-2.0.2c/src/qrna -w 150 -x 50
/qrna-2.0.2c/Demo/HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q >
qrna-2.0.2c/Demo/HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q.W150.X50.qrna2

produces an output that starts with:

```
#-------------------------------------------------------------------------------
#      qrna 2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
```

```
#-------------------------------------------------------------------------------
#      PAM model =  BLOSUM62
#-------------------------------------------------------------------------------
#      RNA model       =  /mix_tied_linux.cfg
#      RIBOPROB matrix =  /RIBOPROB85-60.mat
#-------------------------------------------------------------------------------
#      seq file  =  ../Demos/HG_13_RNAs_gene.fa.MGSCv3.fragchrom.blast.E0.01.D1.q
#                   #seqs: 1734 (max_len = 358)
#-------------------------------------------------------------------------------
#      window version: window = 150   slide = 50 -- length range = [0,9999999]
#-------------------------------------------------------------------------------
# 1  [both strands]
>NT_034563\-2555387\-2555837-71>406- (344)
>chr3\-frag968\-96700001\-96801000-15935>16265- (344)

length of whole alignment after removing common gaps: 344
Divergence time (variable): 0.271641
[alignment ID = 69.48 MUT = 24.42 GAP = 6.10]

length alignment: 150 (id=72.00) (mut=20.00) (gap=8.00)
posX: 0-149 [0-146](147) -- (0.13 0.39 0.34 0.14)
posY: 0-149 [0-140](141) -- (0.11 0.40 0.35 0.13)
LOCAL_DIAG_VITERBI -- [Inside SCFG]
OTH ends *(+) =  (0..[150]..149)
OTH ends  (-) =  (0..[150]..149)
COD ends *(+) =  (87..[12]..98)
COD ends  (-) =  (76..[9]..84)
RNA ends *(+) =  (120..[30]..149)
RNA ends  (-) =  (48..[102]..149)
winner = OTH
              OTH =      224.450             COD =      208.783             RNA =      219.732
   logoddspostOTH =        0.000  logoddspostCOD =      -15.667  logoddspostRNA =       -4.718
     sigmoidalOTH =        4.717    sigmoidalCOD =      -15.721    sigmoidalRNA =       -4.718
```

When alignments have been created using BLASTN, the name of the sequences involved in the alignment contains information about the alignment. That information is extracted from the BLASTN file, and processed by blastn2qrnadepth.pl in the following format:

```
>name-from[><]to-more_info (length_alignment)
```

where > (<) indicates the alignment involves the positive (negative) strand.
For instance in the previous example,

```
>NT_034563-2555387-2555837-71>406-Telomerase_RNA (344)
```

name = NT_034563-2555387-2555837
alignment involves positions 71 to 406 of that clone fragment, in the positive strand.
more_info = telomerase_RNA

```
>chr3-frag968-96700001-96801000-15935>16265- (344)
```

name = chr3-frag968-96700001-96801000
alignment involves positions 15935 to 16265 of the chromosome fragment, in the positive strand.
more_info = none

Analyze the results

There are several Perl scripts to parse through a QRNA output file, and obtain different types of information.

To exemplify some of those possibilities in directory "qrna-2.0.2c/Demos/" I have included the following files

- "m52nc.fas-salmonella_typhi.q"
  which is an already QRNA–processed collection of 12,037 WUBLASTN alignments of intergenic *E. coli* to *Salmonella typhi*.
- "m52nc.fas-salmonella_typhi.q.W150.X50.qrna2"
  which is the result of running the command line:
  qrna-2.0.2c/src/qrna -w 150 -x 50 qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q

qrna2gff.pl

This script converts a QRNA output to gff format.
Command line:
qrna-2.0.2c/scripts/qrna2gff.pl [**options**] **foo.qrna2**

```
usage:  qrna2gff.pl [options] file.qrna

options:
-c <case>          :  cases (default is case = 1)
                        possible cases are:
                        0=GLOBAL
                        1=LOCAL_DIAG_VITERBI 2=LOCAL_DIAG_FORWARD
                        3=LOCAL_SEMI_VITERBI 4=LOCAL_SEMI_FORWARD
                        5=LOCAL_FULL_VITERBI 6=LOCAL_FULL_FORWARD
-i <id_max>        :  max identity of alignments analysed (default is id_max = 100)
-g <typetarget>    :  which type of loci you want to analyze (default is all)
                        possible types of loci are:
                        OTH | COD | RNA
-s <type_of_score> : type of score (sigmoidal | simple)       [default = sigmoidal]
-u <cutoff>        : default is cutoff = 0
-w <whichorg>      : default is whichorg = 1  (use 1-for-org1 2-for-org2 12-for-both)
-x <name>          : ignore given name, use this one for gff outputs
```

We can convert to gff format the file "m52nc.fas-salmonella_typhi.q.W150.X50.qrna2" using the following command,
qrna-2.0.2c/scripts/qrna2gff.pl -u 0.0 qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2

As a result "qrna2gff.pl" produces an gff output file: "m52nc.fas-salmonella_typhi.q.W15.X50.qrna2.all.CUTOFF0.0.gff" which contains the coordinates and winning scores of all windows which score above a given cutoff. The beginning of the file looks like this,

```
ecoli.m52       QRNA    OTH     41      188     7.01119847819906    .   .   gene 'ecoli.m52' id '75.33'
ecoli.m52       QRNA    OTH     89      189     3.49107499855769    .   .   gene 'ecoli.m52' id '81.19'
ecoli.m52       QRNA    OTH     256     336     2.63094043431709    .   .   gene 'ecoli.m52' id '93.90'
ecoli.m52       QRNA    OTH     5561    5674    0.493014423761848   .   .   gene 'ecoli.m52' id '75.86'
ecoli.m52       QRNA    OTH     5550    5673    3.49729862650143    .   .   gene 'ecoli.m52' id '70.54'
ecoli.m52       QRNA    OTH     5536    5675    8.50675930661753    .   .   gene 'ecoli.m52' id '65.99'
ecoli.m52       QRNA    OTH     5578    5670    2.45497250350133    .   .   gene 'ecoli.m52' id '72.92'
ecoli.m52       QRNA    OTH     5599    5680    5.14832544377935    .   .   gene 'ecoli.m52' id '72.29'
ecoli.m52       QRNA    OTH     5561    5680    4.96820913680488    .   .   gene 'ecoli.m52' id '65.62'
ecoli.m52       QRNA    OTH     5567    5672    6.03622921371645    .   .   gene 'ecoli.m52' id '69.72'
ecoli.m52       QRNA    OTH     5570    5670    0.731578035173864   .   .   gene 'ecoli.m52' id '66.35'
ecoli.m52       QRNA    OTH     5549    5600    1.49810737033746    .   .   gene 'ecoli.m52' id '83.02'
ecoli.m52       QRNA    RNA     5549    5626    3.04432435938766    .   .   gene 'ecoli.m52' id '72.62'
```

phase_count_fast.pl

After a genome–to–genome QRNA screen, one needs to post process the QRNA output. The first kind of processing one would like to do with a QRNA output is to extract the actual independent genomic regions (loci) that are identified as RNAs or coding by QRNA. A locus can be composed of a single window, a collection of continuous windows from a given alignment, or a collection of windows from different alignments that include that particular region, and are identified with the same function by QRNA. To obtain loci we have the script **phase_count_fast.pl**.

Command line:
qrna-2.0.2c/scripts/phase_count_fast.pl [options] foo.qrna2 query_organism database_organism

---

```
usage:  phase_count_fast.pl [options] file.qrna org1 org2

options:
-c <case>          :  cases (default is case = 1)
                        possible cases are:
                        0=GLOBAL
                        1=LOCAL_DIAG_VITERBI 2=LOCAL_DIAG_FORWARD
                        3=LOCAL_SEMI_VITERBI 4=LOCAL_SEMI_FORWARD
                        5=LOCAL_FULL_VITERBI 6=LOCAL_FULL_FORWARD
-i <id_max>        :  max identity of alignments analysed (default is id_max = 100)
-l <loci_overlap>  :  minimun overlap required to build loci (default is loci_overlap = -1)
-g <typetarget>    :  which type of loci you want to analyze (default is all three)
                        possible types of loci are:
                        OTH | COD | RNA
-o <output>        : output file [default = ]
-q <file.q>        : include qfile to check if all the alignments were analysed
-s <type_of_score> : type of score (sigmoidal | simple)       [default = sigmoidal]
-t                 : towhomness -- obtains corresponding loci in the other organism
-u <cutoff>        : default is cutoff = 5
-w <whichorg>      : default is whichorg = 1  (use 1-for-org1 2-for-org2 12-for-both)
-x <name>          : ignore given name, use this one for gff outputs
```

We can extract the genomic loci (with scores larger than a cutoff set with option -u to 0 bits) from the file "m52nc.fas-salmonella_typhi.q.W150.X50.qrna2" using the following command,
qrna-2.0.2c/scripts/phase_count_fast.pl
-u 0.0 qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2 ecoli.m52 Styphi

There are three output files to this script.

- **qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2.allloci.CUTOFF0.0**
  Lists all loci ("RNA", "COD", and "OTH") with score larger than 0.0 for both organisms.
- **qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2.allloci.CUTOFF0.0.ecoli.m52.gff**
  List in gff format of all loci ("RNA", "COD", and "OTH") with score larger than 0.0 for the query organism, in this case "ecoli.m62".
- **qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2.allloci.CUTOFF0.0.Styphi.gff**
  List in gff format of all loci ("RNA", "COD", and "OTH") with score larger than 0.0 for the database organism, in this case "Styphi".

The first output file **qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2.allloci.CUTOFF0.0** lists all the loci for both organism, starting with the "RNA" loci, and then proceeding with the "COD" loci, and finally the "OTH" loci. The file looks like,

---

```
---------------Some General Statistics-------------------
FILE:                    m52nc.fas-salmonella_typhi.q.W150.X50.qrna2
```

```
method:                 LOCAL_DIAG_VITERBI
Cutoff:                 0.0

max id:                 100

# blastn hits:          12037
# windows:              14466
---------------------------------------------------------

---------------Statistics by Windows---------------------
# windows:              14466

RNA>0:                  5634/14466
RNA>cutoff:             5634/14466

COD>0:                  329/14466
COD>cutoff:             329/14466

 in phases:             14430/14466
        RNA:                    5634/14430
        COD:                    329/14430
        OTH:                    8467/14430

 in transitions:        0/14466
        RNA/COD:                0/0
        RNA/OTH:                0/0
        COD/OTH:                0/0
        RNA/COD/OTH:            0/0
---------------------------------------------------------

---------------Statistics for RNA loci (ecoli.m52):-------------------
# loci: 431
ave_length:      151.84

1-loci ecoli.m52 5549 5626 (78) 6 RNA -10.62 2.63
2-loci ecoli.m52 9192 9305 (114) 1 RNA -11.95 0.70
3-loci ecoli.m52 14080 14167 (88) 1 RNA -12.04 3.56
4-loci ecoli.m52 20509 20593 (85) 1 RNA -10.08 0.75
5-loci ecoli.m52 20732 20813 (82) 1 RNA -10.84 3.69
```

For instance, we see that 431 independent RNA loci have been identified in *E. coli* that score as RNA above 0 bits out of the 14,466 overlapping windows analyzed.
The file then proceeds to list the coordinates of each of those loci. The information given for each locus has the following form

```
num-loci name_seq loc_from loc_to (loc_length) number_wind type_loc COD_sc RNA_sc
```

Therefore,

```
1-loci ecoli.m52 5549 5626 (78) 6 RNA -10.62 2.63
```

means that the first *E. coli* RNA locus corresponds to sequence named "ecoli.m52" (whole genome). The RNA locus has a length of 78 nucleotides and covers the region from nucleotide 5549 to 5626. Three different windows have contributed to this RNA locus, and the average sigmoidal score for the coding model is -10.62 bits, while the average sigmoidal score for the RNA model is 2.63 bits.

The same information is included in gff format separated by organism. For instance for the intergenic ecoli all the loci in gff format can be found in file

qrna-2.0.2c/Demos/m52nc.fas-salmonella_typhi.q.W150.X50.qrna2.allloci.CUTOFF0.0.ecoli.m52.gff. The beginning of the file looks like,

```
ecoli.m52       QRNA_loci       RNA     5549    5626    2.63439499740037    .   .   gene "ecoli.m52"
ecoli.m52       QRNA_loci       RNA     9192    9305    0.697044668748066   .   .   gene "ecoli.m52"
ecoli.m52       QRNA_loci       RNA     14080   14167   3.56460655273722    .   .   gene "ecoli.m52"
ecoli.m52       QRNA_loci       RNA     20509   20593   0.750425446582323   .   .   gene "ecoli.m52"
ecoli.m52       QRNA_loci       RNA     20732   20813   3.69398772210946    .   .   gene "ecoli.m52"
ecoli.m52       QRNA_loci       RNA     25702   25825   8.78943436198997    .   .   gene "ecoli.m52"
```

Here we report the beginning and end coordinates of a given locus respect to the "ecoli.m52" sequence, with the corresponding QRNA winning score, which is an average of the QRNA winning scores of all the windows contributing to define that locus.

qrna2col.pl

This script converts a QRNA output to "col" format.
Command line:
qrna-2.0.2c/scripts/qrna2col.pl [**options**] **foo.qrna2**

```
usage:  qrna2col.pl [options] file.qrna

options:
-c <case>          :  cases (default is case = 1)
                        possible cases are:
                        0=GLOBAL
                        1=LOCAL_DIAG_VITERBI 2=LOCAL_DIAG_FORWARD
                        3=LOCAL_SEMI_VITERBI 4=LOCAL_SEMI_FORWARD
                        5=LOCAL_FULL_VITERBI 6=LOCAL_FULL_FORWARD
-d <qfile_dir>     : location of the .q file. (default is assumes ../qfile)
-f                 :  create qfile-type file with the sequences
-i <id_max>        :  max identity of alignments analysed (default is id_max = 100)
-g <typetarget>    :  which type of loci you want to analyze (default is all)
                        possible types of loci are:
                        OTH | COD | RNA
-q <qfile>         : name of the qfile  (default given by the qrna name)
-s <type_of_score> : type of score (sigmoidal | simple)       [default = sigmoidal]
-u <cutoff>        : default is cutoff = 0
```

Using the file"Rfam.seed.RF00017.SRP.fa.blast.D10.q.rnass.qrna2", which is the result of running the command line:
qrna-2.0.2c/src/qrna –rnass qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q,

the command line:
qrna-2.0.2c/scripts/qrna2col.pl -d qrna-2.0.2c/Demos -g RNA -u 0.0
qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.rnass.qrna2

produces the file
qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.rnass.qrna2.RNA.CUTOFF0.0.col
as output. The first two entries of this file are the first two sequences fragments of the first window that scores as "RNA" with a score higher than 0.0.
The file **qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.rnass.qrna2** was the result of running QRNA with the option "–rnass" which produces a *a posteriory* secondary structure of the region of the window scored as "RNA". As a result the "col"–format file has a sixth column which includes the position a base is basepaired to or a "." is single stranded. If the option "–rnass"is not used, the sixth column is blank.

```
; generated by qrna2col.pl
; ================================================================
; QRNA  qrna 2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
; TYPE  RNA (cutoff=0.0)
; COL1  label
; COL2  residue
; COL3  seqpos
; COL4  alignpos
; COL5  align_bp
; ENTRY RF00017.SRP.AE000759/8335-8411-1>77-
; LEN_A 77
; START 1
; ID    100.00
; SCORE 8.01556851137287
; ----------
N       G       1       1       77
N       C       2       2       76
N       C       3       3       75
N       C       4       4       74
N       T       5       5       73
N       G       6       6       72
N       C       7       7       71
N       G       8       8       70
N       G       9       9       68
N       C       10      10      67
N       G       11      11      66
N       G       12      12      65
N       G       13      13      64
N       A       14      14      63
N       C       15      15      62
N       A       16      16      61
N       G       17      17      60
N       G       18      18      .
N       G       19      19      .
N       T       20      20      .
N       G       21      21      58
N       A       22      22      57
N       A       23      23      56
N       C       24      24      55
N       T       25      25      54
N       C       26      26      53
N       C       27      27      51
N       C       28      28      50
N       C       29      29      49
N       C       30      30      48
N       A       31      31      47
N       G       32      32      46
N       G       33      33      44
N       C       34      34      43
N       C       35      35      42
N       C       36      36      41
N       G       37      37      .
N       A       38      38      .
N       A       39      39      .
N       A       40      40      .
N       G       41      41      36
N       G       42      42      35
```

```
N       G       43      43      34
N       A       44      44      33
N       G       45      45      .
N       C       46      46      32
N       A       47      47      31
N       A       48      48      30
N       G       49      49      29
N       G       50      50      28
N       G       51      51      27
N       T       52      52      .
N       A       53      53      26
N       A       54      54      25
N       G       55      55      24
N       C       56      56      23
N       C       57      57      22
N       C       58      58      21
N       G       59      59      .
N       C       60      60      17
N       C       61      61      16
N       G       62      62      15
N       T       63      63      14
N       C       64      64      13
N       C       65      65      12
N       C       66      66      11
N       G       67      67      10
N       T       68      68      9
N       G       69      69      .
N       C       70      70      8
N       G       71      71      7
N       C       72      72      6
N       A       73      73      5
N       G       74      74      4
N       G       75      75      3
N       G       76      76      2
N       T       77      77      1
; **********
; QRNA  qrna 2.0.2c (Mon Dec  8 16:31:48 CST 2003) using squid 1.5m (Sept 1997)
; TYPE  RNA (cutoff=0.0)
; COL1  label
; COL2  residue
; COL3  seqpos
; COL4  alignpos
; COL5  align_bp
; ENTRY RF00017.SRP.AE000759-8335-8411-1>77-
; LEN_A 77
; START 1
; ID    100.00
; SCORE 8.01556851137287
; ----------
N       G       1       1       77
N       C       2       2       76
N       C       3       3       75
N       C       4       4       74
N       T       5       5       73
N       G       6       6       72
N       C       7       7       71
N       G       8       8       70
```

N	G	9	9	68
N	C	10	10	67
N	G	11	11	66
N	G	12	12	65
N	G	13	13	64
N	A	14	14	63
N	C	15	15	62
N	A	16	16	61
N	G	17	17	60
N	G	18	18	.
N	G	19	19	.
N	T	20	20	.
N	G	21	21	58
N	A	22	22	57
N	A	23	23	56
N	C	24	24	55
N	T	25	25	54
N	C	26	26	53
N	C	27	27	51
N	C	28	28	50
N	C	29	29	49
N	C	30	30	48
N	A	31	31	47
N	G	32	32	46
N	G	33	33	44
N	C	34	34	43
N	C	35	35	42
N	C	36	36	41
N	G	37	37	.
N	A	38	38	.
N	A	39	39	.
N	A	40	40	.
N	G	41	41	36
N	G	42	42	35
N	G	43	43	34
N	A	44	44	33
N	G	45	45	.
N	C	46	46	32
N	A	47	47	31
N	A	48	48	30
N	G	49	49	29
N	G	50	50	28
N	G	51	51	27
N	T	52	52	.
N	A	53	53	26
N	A	54	54	25
N	G	55	55	24
N	C	56	56	23
N	C	57	57	22
N	C	58	58	21
N	G	59	59	.
N	C	60	60	17
N	C	61	61	16
N	G	62	62	15
N	T	63	63	14
N	C	64	64	13
N	C	65	65	12

```
N       C       66      66      11
N       G       67      67      10
N       T       68      68      9
N       G       69      69      .
N       C       70      70      8
N       G       71      71      7
N       C       72      72      6
N       A       73      73      5
N       G       74      74      4
N       G       75      75      3
N       G       76      76      2
N       T       77      77      1
; **********
```

---

plot_qrna2.pl

Another kind of information that one might want to extract from a QRNA output is a distribution of scores according to percentage identity of the alignments involved. For this purpose we have the script **plot_qrna2.pl**.

Command line:
qrna-2.0.2c/scripts/plot_qrna2.pl [options] foo.qrna2 (foo.shuffle.qrna2)

```
usage: plot_qrna2.pl [options] qrnafile qrnafilesh
options:
-c <case>            :  cases (default is case = 1)
                           possible cases are:
                           0=GLOBAL
                           1=LOCAL_DIAG_VITERBI 2=LOCAL_DIAG_FORWARD
                           3=LOCAL_SEMI_VITERBI 4=LOCAL_SEMI_FORWARD
                           5=LOCAL_FULL_VITERBI 6=LOCAL_FULL_FORWARD
-a <max_n_ali>       : max number of alignments                [default all]
-e <evalue>          : fit to an e-value
-i <min_id_plot>     : for ID plots                             [default min_id_plot = 50]
-j <max_id_plot>     : for ID plots                             [default max_id_plot = 100]
-I <min_id>          : min ID for analysis                      [default min_id = 0]
-J <max_id>          : max ID for analysis                      [default max_id = 100]
-g <max_gap_plot>    : for GAP plots                            [default max_gap_plot = 50]
-l <max_num_win>     : max number of windows to be considered   [default all]
-m <max_mut_plot>    : for MUT plots                            [default max_mut_plot = 50]
-n <time increments> : for TIME histo                           [default inc_t = 0.001]
-s <type_of_score>   : type of score (sigmoidal | simple)       [default = sigmoidal]
-t <max_time_plot>   : for plots                                [default max_time_plot = 0.8]
-T <max_time>        : max time for analysis                    [default max_time = 0.8]
-S <min_time>        : min time for analysis                    [default min_time = 0]
-u <cutoff>          : plot all logodds, but add stats for those above cutoff [default cutoff = 0]
-w <displaycutoff>   : plot logodds larger than dislplaycutoff               [default display_cutoff = 0]
-y <min_ord>         : min ordinate for plotting scores
-Y <max_ord>         : max ordinate for plotting scores
```

To exemplify some of those possibilities in directory "qrna-2.0.2c/Demos/" I have included the following files

- "Rfam.seed.RF00017.SRP.fa.blast.D10.q"
  which is an already QRNA–processed collection of 813 WUBLASTN alignments between the 95 SRP RNAs in the seed of the database of noncoding RNAs Rfam 3.0.

- "Rfam.seed.RF00017.SRP.fa.blast.D10.q.qrna2"
  which is the result of running the command line:
  qrna-2.0.2c/src/qrna qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q
- "Rfam.seed.RF00017.SRP.fa.blast.D10.q.shuffle.qrna2"
  which is the result of running the command line:
  qrna-2.0.2c/src/qrna -B qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q

The command line
qrna-2.0.2c/scripts/plot_qrna2.pl Rfam.seed.RF00017.SRP.fa.blast.D10.q.qrna2 Rfam.seed.RF00017.SRP.fa.blast.D10.q.shuffle.qrna2

produces four output files in postscript:

- for COD scores
  - qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.qrna2.id.spreadcod_histo_with_sh.ps
  - qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.qrna2.id.spreadcod_scores_with_sh.ps
- for RNA scores
  - qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.qrna2.id.spreadrna_histo_with_sh.ps
  - qrna-2.0.2c/Demos/Rfam.seed.RF00017.SRP.fa.blast.D10.q.qrna2.id.spreadrna_scores_with_sh.ps

For each kind of score (coding or RNA), the two postscript files include the following information,

- The first postscript file is a histogram of the number of coding (RNA) scoring windows above a chosen cutoff (zero bits by default). See Figure 1 for an example.
- The second files produces a distribution of the coding (RNA) scores (average with standard deviation) with the percentage identity of the alignments. See Figure 2 for an example.

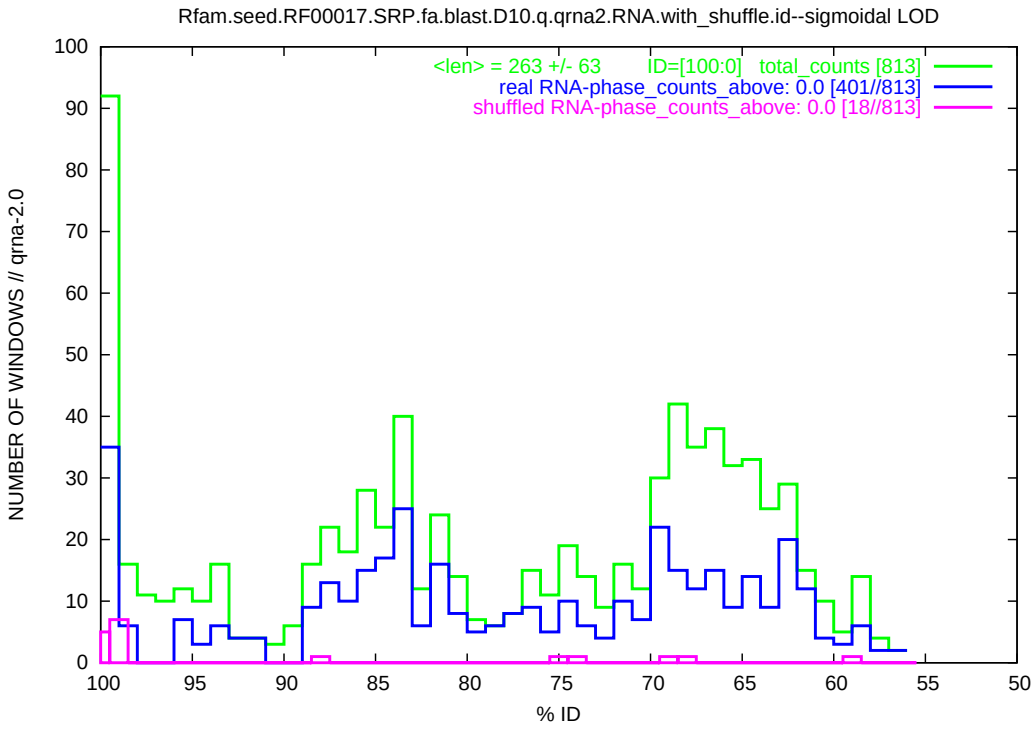

Figure 1: Analysis of the collection of 813 BLASTN alignments between SRP RNAs contained in file "Rfam.seed.RF00017.SRP.fa.blast.D10.q". Alignments have been grouped by percentage identity. This figure represent the histogram of the number of alignments bined in each percentage identity interval. In green we find the histogram for the total number of windows analyzed. In blue we have the histogram for those windows that score as RNA above a cutoff of 0 bits. In red we find the histogram for the windows that score as RNA above a cutoff of 0 bits after shuffling the alignments (false positives).

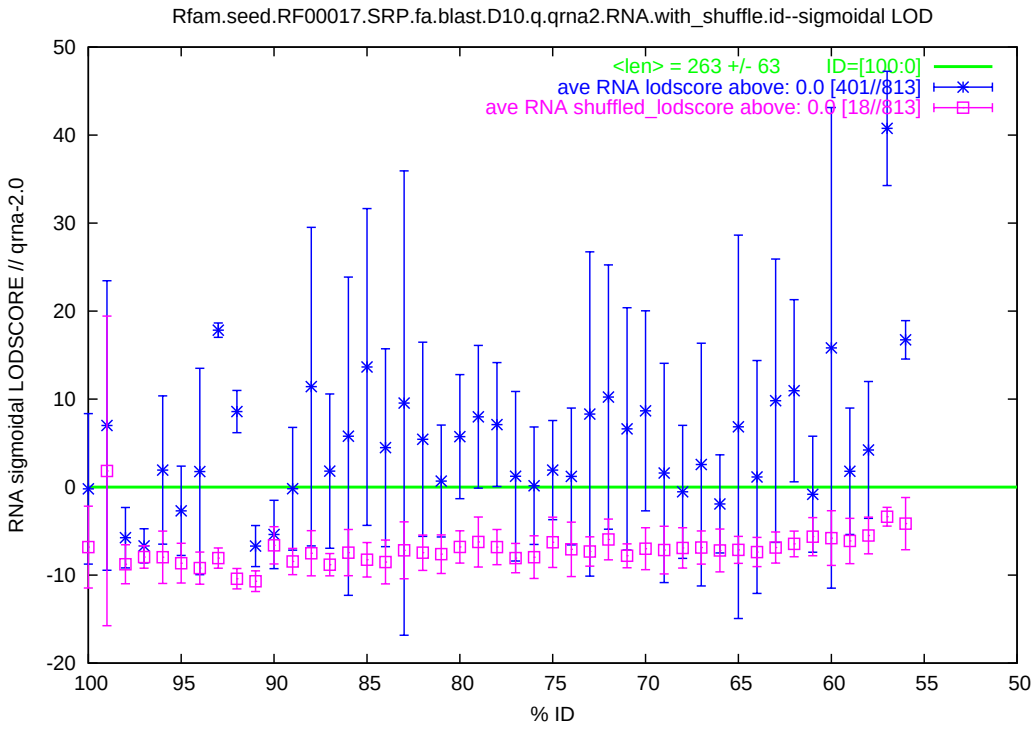

Figure 2: Analysis of the collection of 813 BLASTN alignments between SRP RNAs contained in file "Rfam.seed.RF00017.SRP.fa.blast.D10.q". Alignments have been grouped by percentage identity. This figure represent the RNA scores of all the alignments as a function of the percentage identity of the alignments. "$*$" represents the average of the RNA scores. "□" represents the average of the RNA scores of the corresponding shuffled alignments (false positives). The error bars correspond to one standard deviation.