RNAsnoop and SNOOPY tutorial

University of Leipzig

April 2010

Chapter 1

Introduction

RNAsnoop and its helper script SNOOPYare two programs developed to aid the prediction of H/ACA-snoRNA targets. This tutorial and the accompanying example sequences in the examples directory should help the reader to use both RNAsnoopand SNOOPYto achieve target predictions for orphan snoRNAs. In the first section we will present how to install and configure both RNAsnoopand SNOOPY

We will then present how to use RNAsnoopwith single sequences, with alignments and with accessibilities. Finally the use of SNOOPYwith and without accessibilities will be reviewed. Sequence and accessibility are found in the `examples` directory that is found with the manual package. Perlscripts are located in `examples/perlscript`. Files needed by the machine learning approach are located in `examples/svmfiles`:

28S.aln: clustalw formatted alignment of human, mouse and rat 28S.

28S_homo.fa: human 28S sequence

homo_u1_to_30.out: RNAup accessibility profile for human 28S

ACA51_s2.aln: clustalw formatted alignment of the second stem of human, mouse and rat ACA51

ACA51_s2_homo.fa: second stem of human ACA51 snoRNA

perlscript/apply_svm.pl: perlscript using a support-vector machine (SVM) to filter the results from RNAsnoop

perlscript/snoopy.pl: perlscript using the SVM and conservation informatino to filter the results from RNAsnoop

svmfiles/yeast.svm.scale.12.model: SVM model for interaction with accessibility

svmfiles/yeast.svm.scale.5.model: SVM model for interaction without accessibility

svmfiles/yeast.svm.range: scaling factors for the SVM

Chapter 2

RNAsnoop tutorial

2.1 Installing RNAsnoop

Unpack the tar file using `zcat RNAsnoop.tar.gz | tar -xvf -` Change into the `RNAsnoop` directory you just created. Type:

Compile

```
./configure --prefix=dirname
make
make install
```

to build the library and the programs in the `./Progs` directory and and set the toplevel installationo directory to `dirname` . If the default directory are preferred (`/usr/bin/` and `/usr/lib`) do not use the prefix option.

In case you set the `--prefix` option, do not forget your environment variables. . For C-shell the commands would be

Variable setting

```
setenv PATH ${PATH}:/wherever/RNAsnoop/Progs
setenv MANPATH ${MANPATH}:/wherever/RNAsnoop/man
```

You should now be able to execute the program and read the man page. Take a look at the `RNAsnoop` man page by typing, `man RNAsnoop`.

2.1.1 RNAsnoop command-line options

The extensive number of command-line options accepted by `RNAsnoop` allow the user to fine-tune the snoRNA target search. These options are represented in figure **??**.

RNAsnoop options

```
-a : Maximal upper stem asymmetry
-l : Minimal allowed stem loop energy
-h : Minimal stem loop length
-i : Maximal stem loop length
-o : Minimal 5'duplex energy
-p : Minimal 3'duplex energy
-q : Minimal total duplex energies
-L : Maximal interaction length
-b : Minimal lower stem energy
-x : Minimal duplex and upper energy
-y : Minimal total energy
-j : Minimal distance between U and the snoRNA
     3' end
-k : Maximal distance between U and the snoRNA
     3' end
-v : Minimal distance between snoRNA 5'end and
     3'end of the 3'duplex
-w : Minimal distance between snoRNA 3'end and
     5'end of the 5'duplex
-m : Minimal stem start position
-n : Minimal distance between stem 3'end and
     snoRNA 3'end
-e : Energy threshold of suboptimals
```

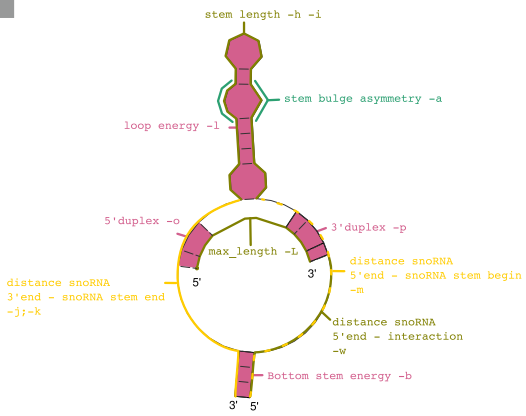

Figure 2.1: **l.h.s** Description of RNAsnoop's command line options. **r.h.s** Graphical representation of RNAsnoop's command line options

Apart from the constraint options, RNAsnoop offers the possibility to search targets in alignments (option -A) or to use the target site accessibility (options -U, -P). Finally with the option -I, RNAsnoop can produce postscript files for interactions returned by a precedent run of RNAsnoop

2.2 Single sequence prediction

The most basic usage of RNAsnoop is to give him two fasta formated files, one containing one or more snoRNA stems and the other one containing target sequences. We should note that RNAsnoop assumes that the sequences in the snoRNA are not the full snoRNA sequences but the stem sequences without the boxes.

`RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa` returns the duplex for which the sum interaction energy and stem energy is the highest. By default, RNAsnoopuses predefined constraints that on the datasets presented in [**?**] gave good results. The corresponding results looks like:

single sequence

```
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa
>ACA51
>homo
<<.<<<<<|.<<<.<<<&.>>>.>>>.((.((((...(((..(((((.....)))))..)))...)))))).))>>>>>.>>.... 806,822;814:  3,63
      (-34.50 = -13.30 + -11.80 + -13.50 +  0.00 + 4.1 )
GUCCUCCCUCGGGAGGG&ACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
```

The second and third line represents the name of the snoRNA and target sequence, respectively. In the third line, interaction structure, location and energy are listed. The region left and right of the $\&$-sign corresponds to the interaction structure for the target and snoRNA sequence, respectively. We have

- $<$ and $>$ representing intermolecular base pairs
- $|$ representing the pseudouridylated uridine
- $.$ representing unpaired nucleotides.
- $($ and $)$ representing intramolecular base pairs

The interaction position on the target and on the snoRNA are separated by $:$. The triplet left of $:$ corresponds to the start, end and pseudouridyle position, respectively. The total interaction energy left of $=$ is the sum of

- two duplex-regions: -13.30 and -11.80 kcal/mol
- upper-stem energy: -13.50 kcal/mol
- low-stem energy: 0.00 kcal/mol
- Duplex penalty: 4.10 kcal/mol

A postscript file of the interaction is automatically generated in the directory where RNAsnoop was run. Its name is made of

- **sno_0_u_**
- position of Ψ: **814**
- name of the target: **homo**
- name of the snoRNA: **ACA51**

that is in our case **sno_0_u_814_homo_ACA51.ps**. This postscript (see **??**)can then be edited by using either the perlscripts available in the ViennaRNA package or with your favourite vector graphics editor (inkscape, adobe illustrator, etc...).

Because of the way how `RNAsnoop`computes the lower stem energy it is wise to look at suboptimals interactions in order to find putative snoRNA-target interactions. This is achieved by using the option $-eE$ where E is an energy range above the energy of the interaction with the lowest sum of stem and duplex energies (see also figure **??**:

single sequence suboptimals

```
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa -e 1
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa -e 1 -N
>ACA51
>homo
<<<<<.<<<<.|.<<.<<<&...(((>>>.>>.((((...(((..(((((.....)))))..)))...)))))..>>>>.>>>>>))) 4468,4486;4479 :
       8,65  (-37.60 = -16.20 + -8.20 + -14.10 + -3.20 + 4.1 )
UGUUCACCCACUAAUAGGG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
<<<<<|.<<<<<<.<<&(..>>.>>>>>>((.(((..(((..(((((.....)))))..)))....))).)).>>>>>...).. 1042,1057;1047 :
     5,62  (-34.40 = -9.80 + -15.10 + -12.80 + -0.80 + 4.1 )
UCUCCUCGUGGGGGGG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
<<<<<|.<<<<<<&...(((>>>>>>((.(((..(((..(((((.....)))))..)))....))).)).>>>>>...))) 1042,1054;1047 :   8,62
      (-37.19 = -9.80 + -15.20 + -12.80 + -3.49 + 4.1 )
UCUCCUCGUGGGG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
<<.<<<<<|.<<<.<<<&(.>>>.>>>.((.((((...(((..(((((.....)))))..)))...))))).))>>>>>.>>.).. 806,822;814 :   4,64
       (-35.30 = -13.30 + -11.80 + -13.50 + -0.80 + 4.1 )
GUCCUCCCUCGGGAGGG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
```

Here the option -N lets RNAsnoop produce postscript file for each suboptimals found. In general, a range of 10 kcal/mol is enough to find all interesting putative interactions. Here again the name of the output files is standardized:

postscript outputs

```
sno_0_u_4479_homo_ACA51.ps
sno_1_u_1047_homo_ACA51.ps
sno_2_u_1047_homo_ACA51.ps
sno_3_u_814_homo_ACA51.ps
```

2.2.1 Single sequence prediction with accessibility profile

Accessibility profiles in form of RNAup or RNAplfold output files can be used to take the target site accessibility into account. Producing accessibility profiles is easy. Given the target sequence file is named 28S_homo.fa and the sequence header is homo, command-line arguments and the corresponding name of the accessibility profiles look like this

RNAup/RNAplfold

```
RNAplfold -u 30 -O < 28S_homo.fa
RNAplfold -u 40 -O < 28S_homo.fa
RNAup -w -u 1-30   < 28S_homo.fa
RNAup -w -u 1-40   < 28S_homo.fa
```

Accessibility profile

```
homo_openen
homo_openen
homo_u1-to-30.out
homo_u1-to-40.out
```

The accessibility files produced by RNAup have names that depend on the command-line arguments used in conjunction with RNAup. This is not the case for RNAplfold.

In order to use accessibility profiles produced by RNAplfold or RNAup, RNAsnoop has to be called like this:

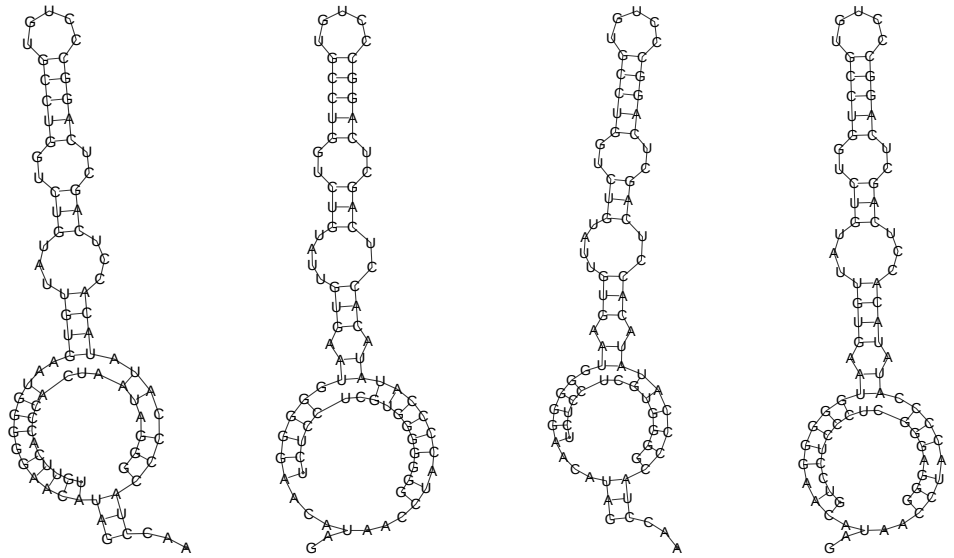

Figure 2.2: Suboptimal produced by RNAsnoop. 28S-4479 stands for the position of the modified uridyle and the name of the target.

RNAsnoop with accessibility

```
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa -e 1 -N -P .                    1
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa -e 1 -P .                       2
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa -e 1 -N -S . -U u1-to-30.out    3
RNAsnoop -t 28S_homo.fa -s ACA51_s2_homo.fa -e 1 -S . -U u1-to-40.out       4
```

The option $-P$ indicates where RNAplfold accessibility files ending with _openen are located. $-U$ indicates where RNAup accessibility files are found and $-S$ tells RNAsnoop how their suffixes look like. Similar to the single sequence RNAsnoop without accessibility, suboptimal ($-e$) and postscript ($-N$) options are available.

Postscript files produced with accessibility definition begin with the string sno_XS_ instead of sno_. They also contains color-coded accessibility information for the target site, where red is a highly accessible nucleotide and green a completely unaccessible nucleotide (see figure **??**).

2.3 Target prediction based on alignments

RNAsnoop can also compute conserved RNA-RNA interactions by taking as input clustalw-formated alignments (-A options). On the exception of the accessibility related options(-U -P -S), all the single sequence command-line options can be used in the alignment variant of RNAsnoop. Before us-

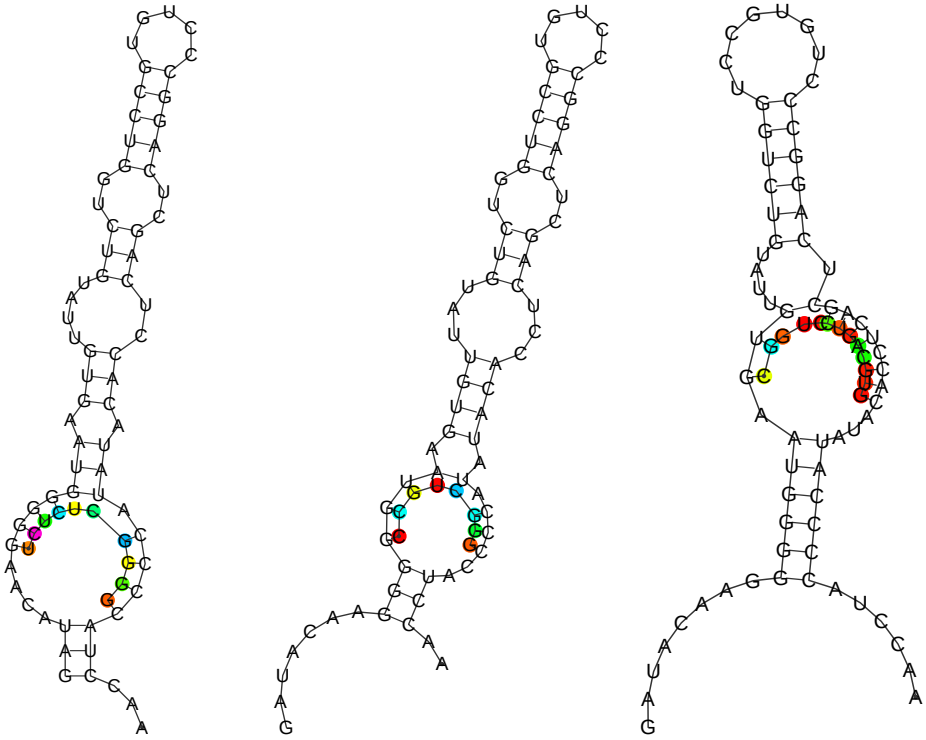

Figure 2.3: Suboptimal produced by RNAsnoop with accessibility. 28S-1196 stands for the position of the modified uridyle and the name of the target.

ing RNAsnoop you should ensure that the query and target alignments contain the same number of sequences and that the sequences in both alignments are in the same order. Possible calls of RNAsnoop with alignments are for example:

RNAsnoop with Alignment

```
RNAsnoop -A -t 28S.clw -s ACA51_s2.aln
RNAsnoop -A -t 28S.clw -s ACA51_s2.aln -N
RNAsnoop -A -t 28S.clw -s ACA51_s2.aln -e 10
RNAsnoop -A -t 28S.clw -s ACA51_s2.aln -e 10 -N
```

The output of the alignment version contains is slightly different to that of the single sequence. The interaction positions are now relative to the position in the alignments. Further the sequence below the consensus structure correspond to the IUPAC consensus sequence. Finally information about the covariance scores in the duplex and stem regions are also indicated:

RNAsnoop with Alignment

```
<.<<<.|.<<<<<<&...((.>>>>>>.((((...(((.(((((((......))))))).)))...)))))..>>>.>......)) 178,191;184 :   8,61
       (-29.09 = -4.93 + -13.10 + -14.77 + -0.38 + 4.1; duplex cov = -22.67; stem cov = 44.00 )
CGCCGCUCGUGGGG&AAMCUACCCYRURUACACCUCAGCYYAGGCCaCUGUGCCUGGUCUGUAUUGUGAAUGRGKRRACAYRG
```

The postscript output contains information about compensatory mutation and incompatible mutations. Basepair containing 0,1,2,3,4,5 compensatory mutations are represented in red, ocher, green, cyan, blue and purple respectively. The color opacity represents the number of incompatible base pairs, where a bright color stands for no incompatible base pairs and transparent stands for more than 2 incompatible base pairs (see figure **??**). Besides the structure, an postscript containing a concatenation of the target and snoRNA alignment is produced. Similar to the structure figure, the alignment figure also contains color-coded information on base pairs (see figure **??**). The naming of the structure and alignment postscripts follows the same pattern as the single sequence structure postscript, with the only difference that sno_ is replaced by snoaln_ and straln_ for the alignment and structure figures, respectively.

2.4 Generating interaction figures based on RNAsnoop output

Depending on the command-line options used with RNAsnoop, the user may end up with many hundred interactions. Producing a postscript output for each of them with the -N option is a time and space consuming task. It might be wiser to produce postscript only for chosen suboptimals. This

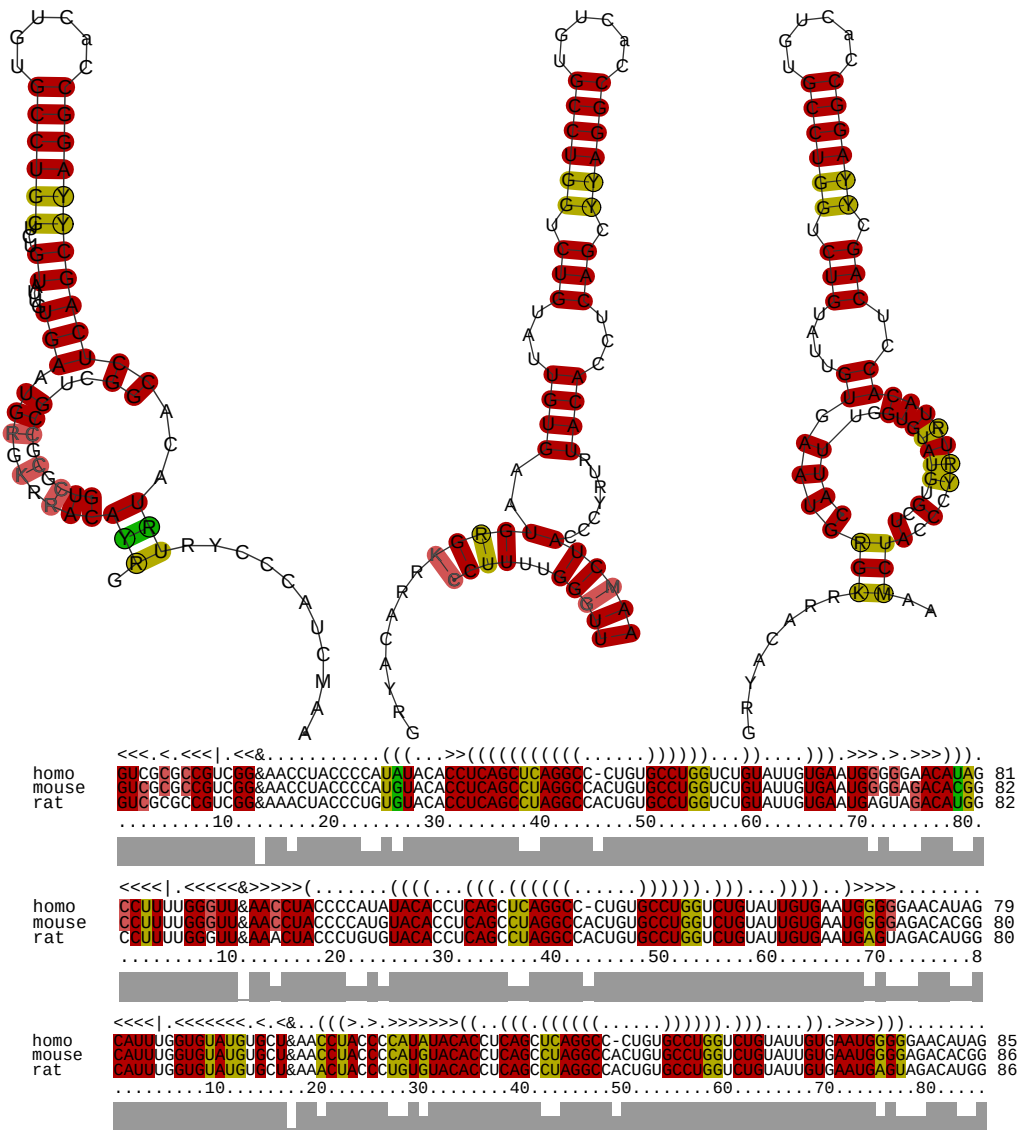

Figure 2.4: Suboptimal of alignment interactions produced by RNAsnoop **top** Annotated structure figure. **bottom** Annotated alignment figure. The leftmost structure corresponds to the topmost alignment, while the rightmost structure corresponds to the bottomost alignment.

is easily achieved with RNAsnoop by using the command-line argument $-I$. This can be done both for single sequences, single sequences with accessibility and multiple sequences alignemnt. It should be noted that in this mode RNAsnoop does not check for the correctness of the structure. It only produces postscript figures based on a RNAsnoopformated output. If desired the output can be saved in an existing directory specified by the $-O$-option. The results input file must be in a file that is than piped to RNAsnoop The results file for single sequences needs the header information (here >ACA51 and >homo), while it is not needed for the multiple sequences approach.

Given the following two interactions:

selected single sequence interaction

```
>ACA51
>homo
<<.<<<<<|.<<<.<<<&.>>>.>>>.((.((((...(((..(((((.....)))))..)))...)))))..))>>>>>.>>.... 806,822;814:
      3,63  (-34.50 = -13.30 + -11.80 + -13.50 +  0.00 + 4.1 )
GUCCUCCCUCGGGAGGG&ACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
```

multiple sequences interaction

```
<.<<<.|.<<<<<<&...((.>>>>>>.(((((...(((.((((((.....)))))).)))...)))))..>>>.>......)) 178,191;184 :
      8,61  (-29.09 = -4.93 + -13.10 + -14.77 + -0.38 + 4.1; duplex cov = -22.67; stem cov = 44.00 )
```

The corresponding postscripts are generated with the following command (see figure **??**):

generating postscript

```
RNAsnoop -I < results.single
RNAsnoop -I -S .  -U u1-to-30.out < results.access
RNAsnoop -I -A -t 28S.clw -s ACA51_s2.aln < results.alignment
```

It should be noted that the output are named similarly to the postscript files produced with the $-N$ option.

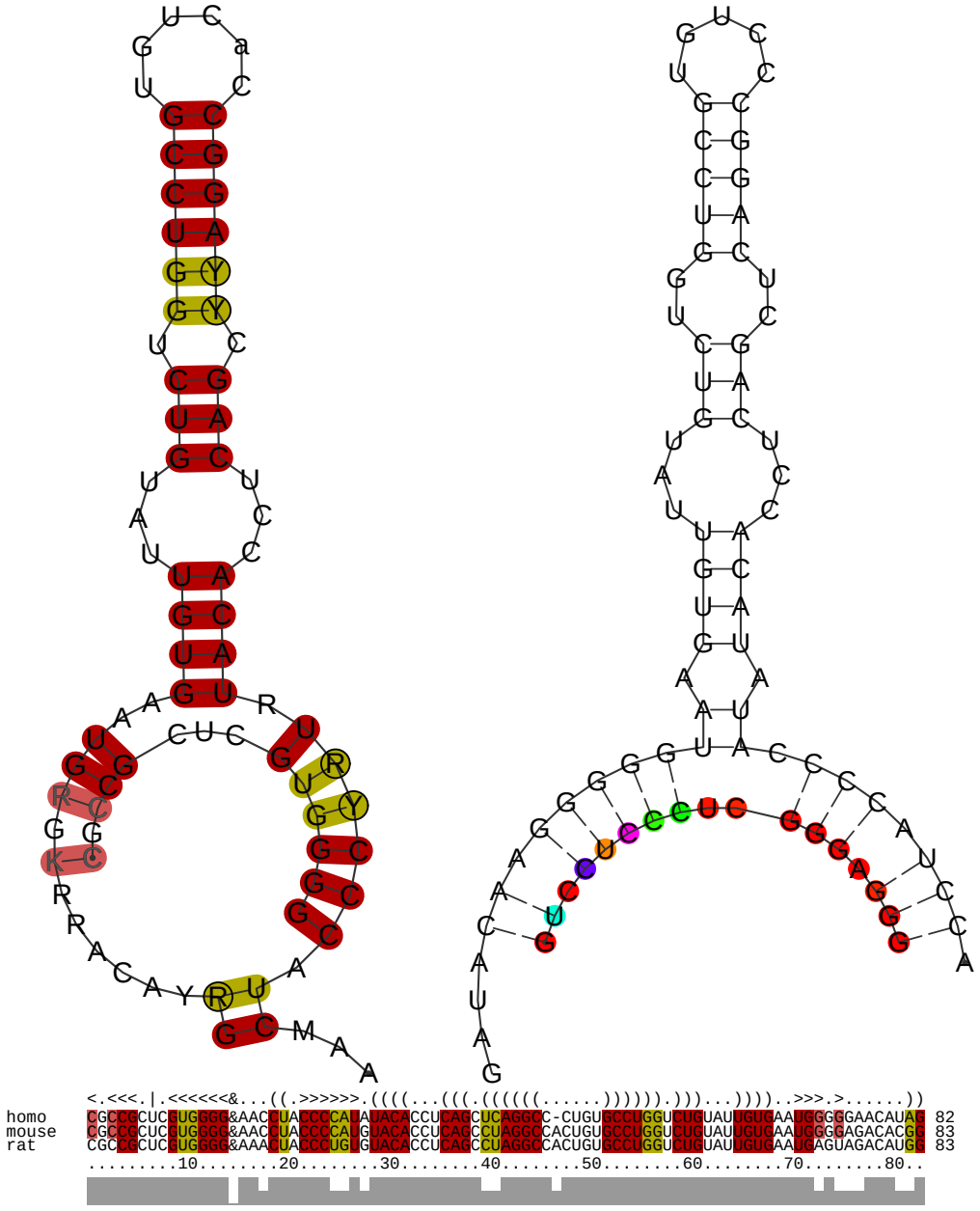

Figure 2.5: Postscript figures produced with the $-I$ option and the interactions listed above. **top** multiples sequences structure and single sequence structure with accessibility annotation. **bottom** multiple sequences alignment.

Chapter 3

Single sequence prediction and svm filtering

A helper perlscript named `svm_apply.pl` allows to filter a large number of hits returned by `RNAsnoop` `svm_apply.pl` uses a support-vector machine trained on experimentally confirmed interactions to decide what looks like a real target and what is probably a false positive. In order to work, `svm_apply.pl` only needs the `libsvm` program package. The command-line options are:

svm_apply.pl command-line options

```
-t : fasta file containing the target
-s : fasta file containing the snoRNA stem
-m : svm model
-r : scaling range
-P : directory containing accessibility files in \texttt{RNAplfold} format
-U : directory containing accessibility files in \texttt{RNAup} format
-S : \texttt{RNAup} suffix
-o : directory where results are stored
```

Command-line options are similar to that used for `RNAsnoop` the main difference being the need to indicate where the svm-model $-m$ and scaling-file $-r$ are located. Those files are found in the `http://www.tbi.univie.ac.at/~htafer/RNAsn` archive as well as in the examples directory accompaying this manual. Finally a directory where the result files are stored must be given $-o$. Based on the file in the examples directory, a typical run would look like this:

svm_apply.pl with accessiblity

```
perl ./perlscript/apply_svm.pl
-t 28S_homo.fa
-s ACA51_s2_homo.fa
-U . -S "u1_to_30.out"
-m svmfiles/yeast.svm.scale.12.model
-r svmfiles/yeast.svm.range
-o pred/
```

svm_apply.pl wo. accessibility

```
perl ./perlscript/apply_svm.pl
-t 28S_homo.fa
-s ACA51_s2_homo.fa

-m svmfiles/yeast.svm.scale.5.model
-r svmfiles/yeast.svm.range
-o pred/
```

`svm_apply.pl` returns on the command-line then all interactions that are classified as real interactions. In our example when considering accessibility they are four:

svm_apply.pl output

```
>1.0.163826.0.836174ACA51
>homo
<<<|.<<<<<<<<<<&......(((>>>>>>>>>>(((...((((((........))))))).....)))>>>))).......  4595,4609;4598 :
      10,56  (-26.60 = -3.50 + -16.70 + -7.90 + -8.40 +  5.80 + 4.10)
CAUUUGGUGUAUGUG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
>1.0.322568.0.677432ACA51
>homo
<<<<<|.<<<<<<&...(((>>>>>>((.(((..((((..(((((.....)))))..)))....))).))>>>>>....))) 1042,1054;1047 :   7,60
       (-22.70 = -9.00 + -13.10 + -12.80 + -3.49 + 11.59 + 4.10)
UCUCCUCGUGGGG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
>1.0.370556.0.629444ACA51
>homo
<<<<|.<<<&...(((>>>(((.((((...(((..(((((.....)))))..)))...)))))).)))>>>>....))) 4319,4327;4323 :   7,60
      (-20.18 = -5.30 + -6.60 + -15.60 + -3.49 +  6.71 + 4.10)
CCUUUUGGG&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
>1.0.406855.0.593145ACA51
>homo
<<<<|.<<<<&...((>>>>(((.((((...(((..(((((.....)))))..)))...)))))).)))>>>>.....)) 4319,4328;4323 :   6,60
      (-20.39 = -5.30 + -8.20 + -15.60 + -2.10 +  6.71 + 4.10)
CCUUUUGGGU&AACCUACCCCAUAUACACCUCAGCUCAGGCCCUGUGCCUGGUCUGUAUUGUGAAUGGGGGAACAUAG
```

For each interactions the topmost header contains the information on the output of the classification and the name of the snoRNA stem. The second header contains the name of the target. The rest is formated like the standard `RNAsnoop` output. Besides the command-line output, `svm_apply.pl` also produces postscript figures of the interactions (see figure **??**).

Figure 3.1: Postscript figures produced with `svm_apply.pl`, for the four selected interactions between the second stem of human snoRNA ACA51 and human 28S rRNA. `28S-4323` and `28S-4598` are real pseudouridylated uridine.

Chapter 4

SNOOPY

The SNOOPY perlscript is intended to simplify the search for snoRNA-RNA interactions. It is a roughly a two steps approach where first `svm_apply.pl` is used to select putative interactions in a reference organism. These selected interactions are then checked for conservation.

SNOOPYrequires different external program in order to be functional. Those are:

libsvm `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`

locarna `http://www.bioinf.uni-freiburg.de/Software/LocARNA/`

Bioperl `http://bioperl.org/wiki/Main_Page`

RNAsnoop `http://www.tbi.univie.ac.at/~htafer/RNAsnoop/`

svm_apply.pl `http://www.tbi.univie.ac.at/~htafer/RNAsnoop/`

libsvm is a machine-learning program package used to filter the results out. **locarna** is used to structurally realign the snoRNA alignments given as input. **Bioperl** is necessary to parse the input sequence alignments. Finally **svm_apply.pl** is a wrapper script to **libsvm**. It can also be used as a standalone script to filter single sequence interaction predictions (see previous chapter). The command-line options are:

SNOOPY command-line options

```
-S : clustalw formated files containing the snoRNA-stems sequences
-T : directory containing target alignments in clustalw format
-O : Organism for which the single sequence target search should be done
-P : directory where the RNAplfold profiles are
-U : directory where the RNAup profiles are
-s : suffix for RNAup profile files.
-r : normalization file for  svm-scale
-m : classification model for svm-predict
```

```
-a : sort hit by energy instead of the svm-score.
-R : the number of desired hits
-N : minimal number of sequences in the aligmnents passed to \prog\
```

Based on the file in the examples directory, a typical run would look like this:

SNOOPY with accessiblity

```
perlscript/snoopy.pl
-S ACA51_s2.aln
-T target/28S.aln
-U . -s "u1_to_30.out"
-r svmfiles/yeast.svm.range
-m svmfiles/yeast.svm.scale.12.model
-R 3
-N 3
-O homo
-w snoopy_pred
-a 1
```

SNOOPY wo. accessiblity

```
perlscript/snoopy.pl
-S ACA51_s2.aln
-T target/28S.aln

-r svmfiles/yeast.svm.range
-m svmfiles/yeast.svm.scale.5.model
-R 3
-N 3
-O homo
-w snoopy_pred
-a 1
```

SNOOPY saves all results into the directory specified by $-w$. In this directory the tabulator separated file $predictions$ contains the following important entries:

snoRNA snoRNA name

svm-target Ψ-position for the selected interaction

svm-score svm-score

energy total energy of interaction

TargetSequence target sequence involved in the interaction

structure structure of the snoRNA-rRNA hybrid

file name of the containing the selected interaction

figure name of the single sequence postscript file

alignmentPos position of interaction in the alignment

Seq in Alignment number of sequence in alignment for the interaction

energy energy of the multiple sequences interactions

figures name of the multiple sequences postscript files

In our example, SNOOPY finds, thanks to `apply_svm.pl`, four putative single sequence interactions (see figure **??**). From these four interactions, only one, `28S-4323` is conserved in 3 species (see figure **??**)

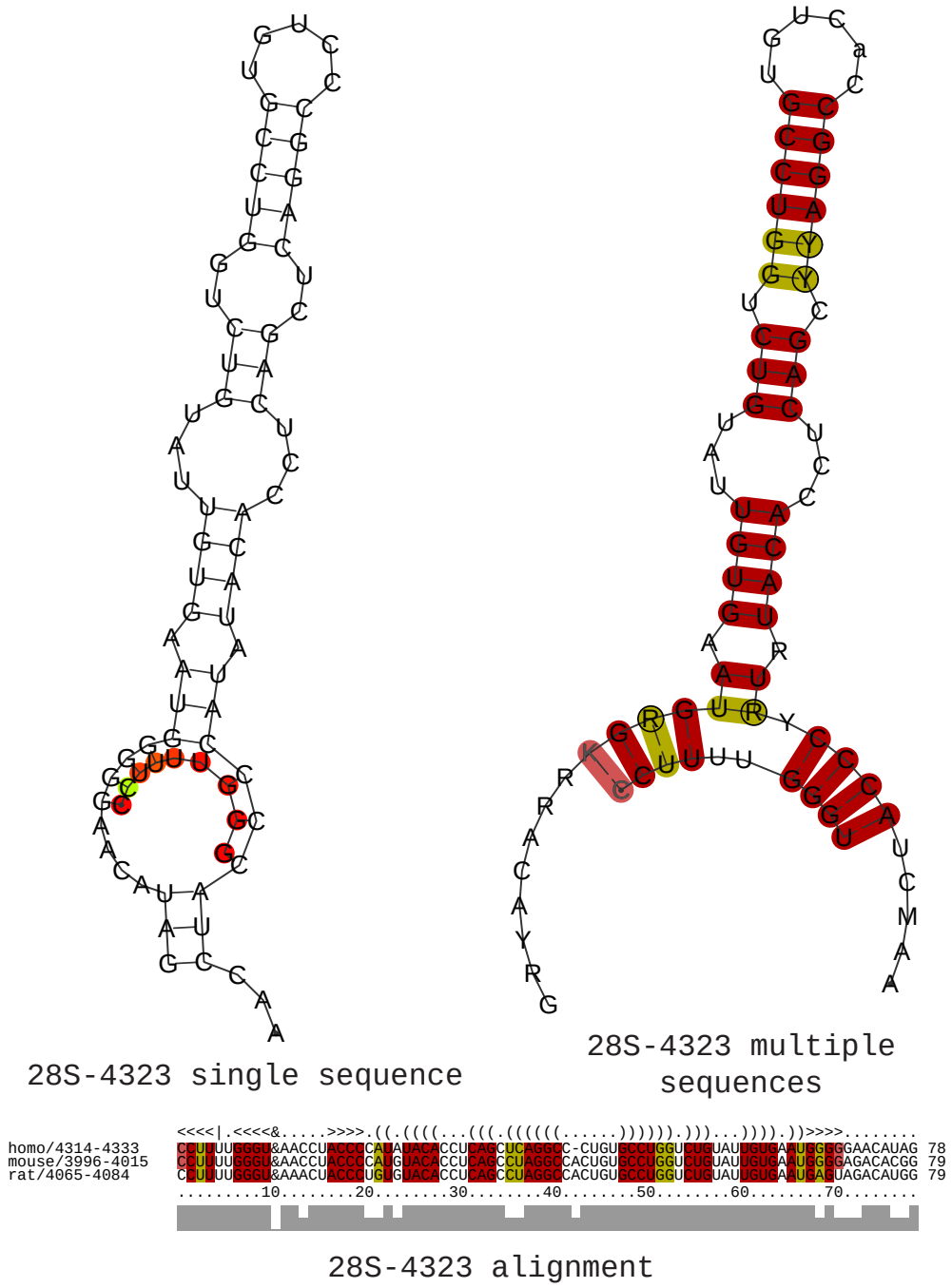

Figure 4.1: **l.f.s** Single sequence interactions the second stem of human snoRNA ACA51 and human 28S Ψ_{4323}. **r.h.s** This interaction is well conserved over rat, mouse and human. **bottom** Alignment view of the multiple sequences structure.