LocARNA-P Manual:
Computing Match Probabilities and Reliabilities in LocARNA's Probabilistic Mode

Sebastian Will

Computer Science, University Freiburg, Freiburg, Germany
CSAIL, MIT, Cambridge, MA, USA

LocARNA version 1.6

Overview

This document is part of the documentation of the tool package LocARNA. The package LocARNA consists of several tools for the comparative analysis of RNA based on simultaneous alignment and folding (SA&F) [4]. This document describes the features of LocARNA-P [7], i.e. the part of the package that extends the original functionality of LocARNA [8] by features based on the efficient computation of match probabilities. These features comprise

- Multiple alignment based on probabilistic consistency transformation
- Assessment of local alignment quality by reliability profiles
- Locating RNA motifs in reliability profiles
- De-novo prediction of structural, non-coding RNA
- Calculating pairwise partition functions and sequence and structure match probabilities
- Refining genome-wide screens for putative non-coding RNA

We describe the installation of the LocARNA package and the usage of the LocARNA-P functionality. Finally, we provide detailed instructions and script for refining genomic screens for the de-novo prediction of structural RNA with LocARNA-P.

1 Installation

The package LocARNA is distributed as source code under license GPLv2 and available for download at

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

We provide installation instructions for a recent GNU/Linux-based desktop operating system like Ubuntu or Fedora.

All but the very low-level functionality of LocARNA requires a recent installation of the Vienna RNA package. If the package is not available on your system, please download from http://www.tbi.univie.ac.at/~ivo/RNA/

and install this package following its installation instructions.

To install LocARNA on your system, download the latest release from

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

At this time the latest release is Version 1.6 and directly obtained from

http://www.bioinf.uni-freiburg.de/Software/LocARNA/Releases/locarna-1.6.tar.gz.

LocARNA is installed from the command line by running the commands

```
tar xzf locarna-1.6.tar.gz
cd locarna-1.6

./configure
make
make install
```

These commands will compile the package and install it under directory `/usr/local`.

Non-standard installation For installing LocARNA in a different directory hierarchy or when the Vienna programs are not in the default search path, one controls this by options of `configure`. `configure --help` provides a complete list of configuration options. Most importantly, the installation path is controlled by

```
./configure --prefix=LOCARNA_INSTALLATION_PATH
```

The installation path of the Vienna RNA package is controlled by LocARNA-specific options and environment variables for `configure`.The `configure` option

```
--with-vrna=VRNAPREFIX
```

selects the installation directory of the Vienna RNA library to `VRNAPREFIX`. Specific names and paths of executables are controlled by environment variables:

RNAfold name of executable RNAfold (def=RNAfold)

RNAplfold name of executable RNAplfold (def=RNAplfold)

RNAalifold name of executable RNAalifold (def=RNAalifold)

TCOFFEE name of executable tcoffee (def=t_coffee)

Note that T-Coffee is not used in probabilistic mode and therefore need not be installed for the functions described in this document. However, for using the tool `locarnate` [1] of the package, T-Coffee is required. It is available from http://www.tcoffee.org/Projects_home_page/t_coffee_home_page.html.

For aligning large sequences it is advised to use the option

```
   --enable-large-pf
```

when calling `configure`. This option avoids under- and overflows when computing the partition function by activating high precision floating point arithmetic.

2 Quick Start

This section describes standard functionality and usage by a small example. The distribution contains some example input in the subdirectory `Examples`. In general LocARNA is controlled from the command line. We assume that `Examples` is the current directory.

We are going to align the sequences specified in the file archaea.fa:

```
>fruA
CCUCGAGGGGAACCCGAAAGGGACCCGAGAGG
>fdhA
CGCCACCCUGCGAACCCAAUAUAAAAUAAUACAAGGGAGCAGGUGGCG
>vhuU
AGCUCACAACCGAACCCAUUUGGGAGGUUGUGAGCU
>hdrA
GGCACCACUCGAAGGCUAAGCCAAAGUGGUGCU
>vhuD
GUUCUCUCGGGAACCCGUCAAGGGACCGAGAGAAC
>selD
UUACGAUGUGCCGAACCCUUUAAGGGAGGCACAUCGAAA
>fwdB
AUGUUGGAGGGGAACCCGUAAGGGACCCUCCAAGAU
```

These sequences are multiply aligned by running

```
   mlocarna --probabilistic --consistency-transformation archaea.fa
```

Due to the options `--probabilistic --consistency-transformation` the alignment is based on match probabilities and probabilistic consistency transformation, which increases the accuracy over the default operation of `mlocarna`.

The command writes the following text output to standard out.

```
mLocARNA --- multiple Local (and global) Alignment of RNA --- LocARNA 1.6
Copyright Sebastian Will

Compute pair probs ...
Compute match probabilities ...
Consistency transform match probabilities ...
Compute pairwise alignments ...
Perform progressive alignment ...
```

```
fdhA               CGC-CACCCUGCGAACCCAAUAUAAAAUAAUACAAGGGAGCAG-GUGG-CG
fwdB               AUG-UUGGAGGGGAACCCGU-------------AAGGGACCCUCCAAG-AU
hdrA               GG--CACCACUCGAAGGCU--------------AAGCCAAAGUGGUG--CU
selD               UUACGAUGUGCCGAACCCUU------------UAAGGGAGGCACAUCGAAA
vhuD               GU--UCUCUCGGGAACCCGU------------CAAGGGACCGAGAGA--AC
vhuU               AGC-UCACAACCGAACCCAU-------------UUGGGAGGUUGUGAG-CU
fruA               CC--UC-GAGGGGAACCCGA-------------AAGGGACCCG-AGA--GG
alifold            ((..((((((((...(((.................))).))))))))..)) \
                                                 (-36.33 = -18.09 + -18.24)
reliability
- 10%              ### #########**###**            ***###*######### ##
- 20%              ##* ########***###**             **###*########* ##
- 30%              ##* ########***###**             **###*########  ##
- 40%              ##  ########***###**             **###*########  ##
- 50%              ##  ###*####***###**             **###*#### ###  ##
- 60%              ##  *# **###***###*              **###*###* *#*  ##
- 70%              ##      *##****##**              ***##**##*      ##
- 80%              #*      **********                ********       *#
- 90%              #         ********                 ******         #
-100%                           ***
fwdB               ‘(..((((((((...(((.................))).))))))))..)’
hdrA               {(..(((((((....(((.................)))..)))))))..)}
vhuD               ((..((((((((...(((.................))).))))))))..))
fruA               ((..((...(((...(((.................))).)))...))..))
vhuU               {((.((((((((...(((.................))).))))))))).)}
fdhA               {((.(((.{(((...(((..{{{‘...’.}}}...))).)))}.)))).)}
selD               ‘(.(((((((((...(((‘...............’))).))))))))).)’
```

After some progress messages, the output contains the generated alignment, the RNAalifold consensus structure for the generated alignment, and reliability information. The call will furthermore generate a directory archaea.out containing result and input files.

3 Central Functionality: Computing Multiple Alignments and Match Probabilities

This section details the probabilistic mode of the multiple alignment tool mlocarna, which is activated by the option --probabilistic. In this mode, mlocarna will compute pairwise match probabilities for all pairs of your input sequences. It computes base match *and* base pair match probabilities. These probabilities are technically defined as probabilities of a respective sequence or structure match in the Boltzmann ensemble of alignment/consensus structure pairs [7].

For fast reference, the tool mlocarna provides an overview of its various options by

```
mlocarna --man
```

In general, `mlocarna INPUT.fa` accepts a fasta file INPUT.fa and writes text output to standard out and writes result files, intermediary files and input files to a target directory. This output directory defaults to `INPUT.out` and can be specified by `mlocarna --tgtdir DIR`.

The call

```
mlocarna --probabilistic --consistency-transformation --tgtdir TGT IN.fa
```

results in the following actions and output.

- read the *input sequences* and their names from IN.fa
- compute base pair probabilities for all input sequences (runs `RNAfold -p`). Write sequences and base pair probabilities to files in subdirectory TGT/input using proprietary pp-format
- compute partition functions and (sequence and structure) match probabilities for all pairs of input sequences. Write match probabilities to files in `TGT/probs/bmprobs` (base match) and `TGT/probs/amprobs` (arc match). In this step `mlocarna` calls the tool `locarna_p`.
- consistency transform sequence and structure match probabilities. Write transformed probabilities to files in `TGT/probs/bmprobs-cbt` and `TGT/probs/amprobs-cbt`
- Compute all pairwise alignments and generate guide tree (by the pair group method algorithm). Write the guide tree to `TGT/results/result.tree`.
- Perform progressive alignment for constructing the multiple alignment. In each progressive alignment step `mlocarna` calls the low-level tool `locarna` in order to perform a maximum expected accuracy alignment based on the computed (and transformed) match probabilities. In each progressive step, write the resulting intermediary alignments to `TGT/intermediates` and the final alignment to `TGT/results/results.aln`.
- run `RNAalifold -r` on the final alignment and write output to standard output and the directory `TGT/results`.
- Compute a reliability profile and a reliability dot plot (containing reliabilities of each base pair in the consensus structure). Experimentally, compute reliabilities projected to each input sequence and predict MEA structures for each sequence. Write the results to to standard out and `TGT/results`.

We list some important options that modify `mlocarna`'s behavior.

- The actions of `mlocarna` are reported verbosely when using option `-v` and even more verbose with option `--moreverbose`.
- Omitting the option `--consistency-transformation` skips the consistency computation and runs the progressive multiple alignment on the untransformed match probabilities.
- `--max-diff=`Δ controls a heuristic in `mlocarna` that trades computation time vs. alignment accuracy. It restricts the difference $|k-(m/n)i|$ for alignment cuts (i,k), where n and m are the sequence lengths. This will, in particular, disallow matches of positions i and k where the above difference is larger than Δ. This restriction is applied to all alignment and match probability computations. $\Delta = 60$ appeared to be a conservative restriction in the Bralibase benchmark.

- `--plfold-span=`L and `--plfold-winsize=`W restrict the base pair size when computing base pair probabilities by calling `RNAplfold` with span L and window size W (default $W = 2L$).
- `--mea-beta=`β Weight of base pair match contribution in probabilistic mode. (Default $\beta = 200$, however $\beta = 400$ yields better results in Bralibase benchmark.)
- `--threads=`k. Use k threads for distributable computations in mlocarna. With this option `mlocarna` gains significant speed on multi-cores.
- `--iterate` or `--iterations=`I Perform iterative refinement for respectively 1 or at most I iterations after the progressive alignment.

For aligning up to about 15 sequences of lengths up to a few hundred nt (like most RNAs in Rfam and as tested in the Bralibase benchmark), a recommended parametrization is e.g.

```
mlocarna --probabilistic --consistency-transformation --iterations=2 \
         --max-diff=60 --mea-beta=400 --tgtdir TGT IN.fa
```

For aligning very long sequences, e.g. 10 sequences of length of a few thousand nt, we recommend the use of `--plfold-span` and multiple cores, e.g.

```
mlocarna --probabilistic --consistency-transformation --iterate -v \
         --max-diff=100 --mea-beta=400 --threads=8 --plfold-span=120 \
         --tgtdir TGT IN.fa
```

4 Reliability Profiles

A reliability profile of LocARNA-P consists of a sequence and a structure reliability for each alignment column. The weighted sum of both *column reliabilities* is called *total column reliability*. Working with LocARNA-P's reliability profiles of alignments always starts by creating a multiple alignment of the alignment sequences using `mlocarna --probabilistic` as discussed in the previous section. The sequences are given to `mlocarna` as input in fasta format.

4.1 Plotting Reliability Profiles

We provide a tool `reliability-profile.pl` that generates reliability profile plots as show in Figure 1. Note that the use of this script requires a working installation of the statistics package `R`. A text-based version of such a profile is already given in the text output of LocARNA-P.

A complete overview of the options of the tool is obtained by

```
reliability-profile.pl --man
```

For generating the plot of Figure 1, one runs

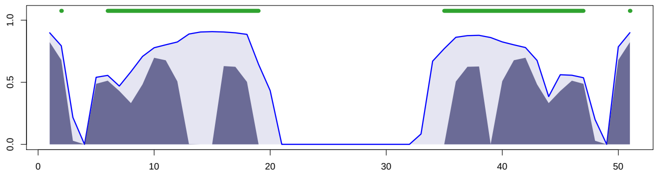

Figure 1: Reliability profile plot of the small example example.fa as generated by reliability-profile.pl

```
mlocarna --probabilistic --consistency-transformation archaea.fa
```

as in the Quick Start section followed by

```
reliability-profile.pl archaea.out
```

This generates the plot and writes it to the file rel.pdf in pdf format. Note that the script requires the target directory archaea.fa of the above mlocarna run as input. Recall that mlocarna computes and stores the profile data in the target directory.

In its default setting, the script reliability-profile.pl predicts the regions of highest reliability and marks them by green lines in the plot. Details of this prediction and how to control it are described in the next section. The prediction can be turned off by the option --dont-predict.

The following options of reliability-profile.pl control its output for pure profile plotting (without controlling the prediction).

--seqname=seqname Project to sequence name

--dont-predict Turn off predicting. (defaults to on)

--title=title Title of plot

--out=filename Output filename

--offset=pos Offset of sequence in genome

--signals=list List of (from,to,orientation) triples to specify signals that will be shown in the plot. One provides the list as string ”from0 to0 orientation0;from1 to1 orientation1 ...”. Its possible to specify multi-range signals by from0a to0a from0b to0b ...

--structure-weight=w Weight of structure against sequence (1.0)

--show-sw Show the influence of structure weight in the plot

--dont-plot Skip plotting, only output prediction

--write-R-script Write the R script

--revcompl Plot (and fit) the reverse complement

--output-format=f Output format (f = pdf or png, defaults to pdf)

--show-fitonoff Show the on/off values for the fit

4.2 Reliability Profiles for Locating RNA Motifs

Likely locations of RNA motifs can be predicted from the reliability profile using the script `reliability-profile.pl`, which will also show the prediction in the resulting plot. Note that the script delegates the actual prediction to the low-level tool `fit`, which never has to be called directly by most users.

The prediction of RNA motifs can be controlled in various ways due to the following options of `reliability-profile.pl`.

`--fit-penalty=`δ Penalty for on/off switching in fit

`--fit-once-on` Restrict fitting to being exactly once on

`--structure-weight=`sw Weight of structure against sequence (1.0)

`--beta=`β Inverse temperature beta

`--revcompl` Plot and fit a reverse complement

`--write-subseq` Write the subsequence of fit

The parameters $delta$, sw, and β control the actual predicting procedure that fits a two-step-function to the signal of total column reliability. δ is a penalty for switching between on and off in the two-step-function. sw is a factor that weights structure reliability against sequence reliability. Large sw result in strong emphasis of the structure signal for the prediction. In practice, it can be useful to set $sw = 2$ or $sw = 3$. β controls the optimization procedure for estimating optimal on and off values for the two-step-function. `--fit-once-on` increases the accuracy, when predicting boundaries of a single RNA motif in an alignment by using this prior knowledge. `--revcompl` will plot the profile of the reverse complement of the alignment and also predict the RNA motif region(s) for the reverse complement.

4.3 Evaluation of Existing Alignments by Reliabilities and Reliability Profiles for De-novo Prediction

LocARNA-P allows the evaluation of existing alignments due to the LocARNA alignment model. In particular, this yields reliability scores that discriminates true non-coding RNA regions from other regions with high accuracy. This can be used in refining genome-wide de-novo prediction of non-coding RNA.

Assume, an existing alignment in file `input.aln`. We further need the unaligned sequences of `input.aln` in a file `input.fa`. For evaluating the alignment in `input.aln`, one first calls `mlocarna` on the sequences in `input.fa`, like

```
    mlocarna --probabilistic input.fa --tgtdir TGT
```

The alignment of `input.aln` is now evaluated by

```
    mlocarna --evaluate input.aln input.fa --tgtdir TGT
```

The option *–tgtdir* can be omitted, in this case the target directory is named `input.out`. When the option is specified, the same name must be given in both calls.

Of course, we can use the script to evaluate existing alignments as well as LocARNA-P generated alignments. We evaluate the LocARNA-P result alignment of our example input sequences by

```
mlocarna --evaluate example.out/results/result.aln example.fa
```

This yields

```
mLocARNA --- multiple Local (and global) Alignment of RNA --- LocARNA 1.6
Copyright Sebastian Will

EVALUATION MODE

fdhA               CGC-CACCCUGCGAACCCAAUAUAAAAUAAUACAAGGGAGCAG-GUGG-CG
fruA               CC--UC-GAGGGGAACCCGA-------------AAGGGACCCG-AGA--GG
fwdB               AUG-UUGGAGGGGAACCCGU-------------AAGGGACCCUCCAAG-AU
hdrA               GG--CACCACUCGAAGGCU--------------AAGCCAAAGUGGUG--CU
selD               UUACGAUGUGCCGAACCCUU------------UAAGGGAGGCACAUCGAAA
vhuD               GU--UCUCUCGGGAACCCGU------------CAAGGGACCGAGAGA--AC
vhuU               AGC-UCACAACCGAACCCAU-------------UUGGGAGGUUGUGAG-CU

- 10%              ### #########**###**            ***###*######### ##
- 20%              ##* ########***###**             **###*########* ##
- 30%              ##* ########***###**             **###*########  ##
- 40%              ##  ########***###**             **###*########  ##
- 50%              ##  ###*####***###**             **###*#### ###  ##
- 60%              ##  *# **###***###*              **###*###* *#*  ##
- 70%              ##      *##****##**              ***##**##*      ##
- 80%              #*      **********                *********      *#
- 90%              #         ********                 ******         #
-100%                           ***

RELIABILITY 1/COL   50.27%
RELIABILITY 2/COL  56.15%
MAX REL. STRUCT.   ((..((((((((...(((.................))).))))))))..))
RELIABILITY 1/CCOL  69.29%
RELIABILITY 2/CCOL  77.40%
```

In general the evaluation outputs the following information.

- the evaluated alignment
- the reliability profile of the alignment (according to the match probabilities in TGT in text form.
- RELIABILITY 1/COL. Reliability 1 divided by the number of alignment columns.
- RELIABILITY 2/COL. Reliability 2 divided by the number of alignment columns.
- MAX REL. STRUCTURE. Structure of maximal reliability.
- RELIABILITY 1/CCOL. Reliability 1 divided by the number of alignment columns containing at least one match.

- `RELIABILITY 2/CCOL`. Reliability 2 divided by the number of alignment columns containing at least one match.

Reliability 1 is defined as the sum of all total column reliabilities (without weighting structure against sequence reliability). *Reliability 2* and the *structure of maximal reliability* are computed as maximal score

$$\sum_{i \text{ unpaired in } P} \text{sequence reliability of column } i + \sum_{(i,j) \text{ paired in } P} 3 \times \text{reliability of base pair } (i,j)$$

over all consensus structures P of the alignment and the maximal structure respectively.

Use for de-novo prediction of non-coding RNA A reliability score that discriminates true and false non-coding RNA alignments particularly well, when obtained from the LocARNA-P alignment is computed by the script reliability-profile.pl. The call

```
reliability-profile.pl example.out --dont-plot --fit-once-on
```

outputs the line

```
SCORE 0.578128400248413 0.366049392832053
```

The first value is called *hit score*, the second *outside score*. In our benchmark on Fly non-coding RNAs, good discrimination was achieved by the score difference hit score-outside score. These scores are computed after predicting the boundaries of the potential non-coding RNA. The hit score is the sum of total column reliabilities in the predicted region, whereas the outside score is this sum over the column outside of the predicted region.

The same call to reliability-profile.pl outputs the line

```
FIT 1 20
```

The first value denotes the start column of the predicted non-coding RNA region, whereas the second is the end column.

5 Pairwise Match Probabilities and Partition Function

The previously described functionality is based on the efficient computation of pairwise sequence and structure match probabilities. Those probabilities are computed by the low-level tool `locarna_p`.

The tool is called on input files `in1` and `in2` that specify the sequences and their structure ensemble (in the form of base pair probabilities) as

```
locarna_p in1 in2
```

The input files `in1` and `in2` are either dot plot postscript files as produced by `RNAfold -p` or files in the LocARNA's proprietary pp format. Files in the latter format are e.g. found in the target directory of a `mlocarna` run.

The following example shows how to use the program on two example sequences, where we use `RNAfold -p` to predict the base pair probabilities and write them to files `seq1_dp.ps` and `seq2_dp.ps`.

```
    printf ">seq1\nACGGACGUAGGGCACGACGUGGGU" | RNAfold -p
    printf ">seq2\nAGCCGACGUAACGGGGCACGUGACU" | RNAfold -p
    locarna_p seq1_dp.ps seq2_dp.ps
```

Without further options the program outputs the partition function of the alignment-consensus structure pairs of the input sequences. Match probabilities can be written to files specified by the options

```
    --write-basematch-probs=<file>
    --write-arcmatch-probs=<file>
```

for writing sequence (base) match probabilities and structure (arc) match probabilities respectively.

A complete list of the options of `locarna_p` is obtained by calling

```
    locarna_p --man
```

6 HOWTO: Refining De-novo Prediction of Structural RNA

In this section, we describe how to refine predictions from a genome-wide screen for non-coding RNAs. We provide scripts that call the previous commands for performing the reliability computation and prediction. Specifically we outline and provdie scripts for the refinement of an `RNAz` [6, 5] screen. Nevertheless, the method can be applied to screens by other programs, e.g. by `EvoFold` [2], as well.

We assume that the reader is familiar with the general setup of a genome-wide screen by `RNAz`. Such a procedure is described in [3] for *Drosophila melanogaster* using a *Drosophilids* whole genome alignment generated by PECAN. The results from this particular screens are available online and can be downloaded from

`http://www.bioinf.uni-leipzig.de/publications/supplements/07-001`.

The candidate annotation table from this screen contains positions in the *D. melanogaster* genome for each putative non-coding RNA locus. Furthermore, the table contains the RNAz max P score for each locus, which is used to identify high and low confidence predictions.

The goal of our refinement is to predict precise boundaries for the putative non-coding RNAs in each locus and to assign reliability scores that improve the discrimination between true and false predictions over the RNAz max P score.

We provide three scripts that in combination perform the refinement of the Fly RNAz screen in the hope that they are useful for other de-novo non-coding RNA screens too. The first script `locarnap-revisit-RNAz-hits.pl` reads the annotation from the RNAz screen, extracts the relevant locus annotation, and, for each locus, writes the corresponding slice of the PECAN whole genome alignment to disk.

Extracting locus alignment slices In this first step, we prepare the input for LocARNA-P by extracting alignments of the RNAz loci with genomic context from the whole genome aligmnent and writing the alignments and their sequences to files. We will also produce a table of meta information on the loci. The script is called as

```
locarnap-revisit-RNAz-hits.pl --all 2>/dev/null >annotation
```

to write the annotation to file `annotation`, the alignment slices to directory `Alignment-Slices`, and the sequences of the alignments to `Realign-Sequences`. The script expects to find all input data in subdirectory `Data`.

The directory `Data` contains the directories

- `Annotation` containing annotation files from the Fly RNAz screen,
- `Alignments` containing the PECAN whole genome alignment. The alignment is organized in chromosomes, such that there is one directory *CHR*`-pecan-CAF1` for each Drosophila chromosome *CHR* in 2L, 2R, 3L, 3R, 4, and X, and
- `Dmel-r4.3` containing the sequences of the *D. melanogaster* chromosomes in fasta format, organized in files `dmel-`*CHR*`-chromosome-r4.3.fasta.gz`. The assembly of this data and the alignment match (here release 4.3).

See `locarnap-revisit-RNAz-hits.pl --man` for full documentation of the script.

Realigning the loci In the second step, we use `mlocarna` to realign all loci sequences that were extracted in the step one. The script `locarnap-realign-all.pl` realigns all loci by calling

```
mlocarna --probabilistic --consistency-transformation --max-diff=100
--struct-weight=200 --mea-beta 400 locusXY.mfa
```

for each file of locus sequences `locusXY.mfa`. By default, the script submits the jobs to a sun grid engine (SGE) queue and expects to be started on a head node. Alternatively, the script can be run locally (option `--run-locally`) and provides support for multicore machines (option `--threads`. The script expects the directories from step one in the current directory. The results are written to directory `Alignment-Results`, which has to exist already. Several constants controlling the behavior of the script can be adapted in the code. An option `--test` allows to control the SGE script and submission command before submission to the grid engine. By default, the script realigns in the forward direction only. If also realignment of the reverse complements is wanted, the script has to be called a second time with option `--revcompl`. See `locarnap-realign-all.pl --man` for full documentation of the script.

Predicting boundaries and locus reliability The script `locarnap-predict-and-plot.pl` is provided to perform the actual prediction of boundaries and reliability of each putative noncoding RNA locus. Furthermore, the script generates reliability profile plots with optional annotation (in directory `Relplots`). The script is called as

```
locarnap-predict-and-plot.pl annotation
```

and expects the results from the previous script in directory `Alignment-Results`. See

```
    locarnap-predict-and-plot.pl --man
```

for full documentation of the script.

7 Reporting bugs.

Please report bugs to Sebastian Will (*will at informatik.uni-freiburg.de*). It is much appreciated to include the complete input data, the program call with complete parameters, and the program output in the bug report. For tracking bugs, it is of further help to compile the package with `configure` option `--enable-debug` and report potential error messages (and `gdb` stack trace).

References

[1] Wolfgang Otto, Sebastian Will, and Rolf Backofen. Structure local multiple alignment of RNA. In *Proceedings of German Conference on Bioinformatics (GCB'2008)*, volume P-136 of *Lecture Notes in Informatics (LNI)*, pages 178–188. Gesellschaft für Informatik (GI), 2008.

[2] J. S. Pedersen, G. Bejerano, A. Siepel, K. Rosenbloom, K. Lindblad-Toh, E. S. Lander, J. Kent, W. Miller, and D. Haussler. Identification and Classification of Conserved RNA Secondary Structures in the Human Genome. *PLoS Comput Biol*, 2(4):e33, 2006.

[3] Dominic Rose, Jorg Hackermuller, Stefan Washietl, Kristin Reiche, Jana Hertel, Sven Findeiss, Peter F. Stadler, and Sonja J. Prohaska. Computational RNomics of drosophilids. *BMC Genomics*, 8:406, 2007.

[4] David Sankoff. Simultaneous solution of the RNA folding, alignment and protosequence problems. *SIAM J. Appl. Math.*, 45(5):810–825, 1985.

[5] Stefan Washietl, Ivo L. Hofacker, Melanie Lukasser, Alexander Huttenhofer, and Peter F. Stadler. Mapping of conserved RNA secondary structures predicts thousands of functional noncoding RNAs in the human genome. *Nat Biotechnol*, 23(11):1383–90, 2005.

[6] Stefan Washietl, Ivo L. Hofacker, and Peter F. Stadler. Fast and reliable prediction of noncoding RNAs. *Proc. Natl. Acad. Sci. USA*, 102(7):2454–9, 2005.

[7] Sebastian Will, Tejal Joshi, Ivo L. Hofacker, Peter F. Stadler, and Rolf Backofen. Locarna-P: Accurate boundary prediction and improved detection of structured rnas for genome-wide screens. LocARNA-P manuscript, submitted.

[8] Sebastian Will, Kristin Reiche, Ivo L. Hofacker, Peter F. Stadler, and Rolf Backofen. Inferring non-coding RNA families and classes by means of genome-scale structure-based clustering. *PLOS Computational Biology*, 3(4):e65, 2007.

NAME

locarna − manual page for locarna (LocARNA 1.6)

SYNOPSIS

locarna [*-h,--help*] [*-V,--version*] [*-v,--verbose*] *-m,--match=<score> -M,--mismatch=<score> --ribosum-file=<f> --use-ribosum=<bool> -i,--indel=<score> --indel-opening=<score> -s,--struct-weight=<score>* [*-e,--exp-prob=<prob>*] *-t,--tau=<factor> -E,--exclusion=<score>* [*--stacking*] *--struct-local=<bool> --sequ-local=<bool> --free-endgaps=<spec>* [*--normalized=<L>*] *-w,--width=<columns>* [*--clustal=<file>*] [*--pp=<file>*] [*-L,--local-output*] [*-P,--pos-output*] [*--write-structure*] *-p,--min-prob=<prob> -D,--max-diff-am=<diff> -d,--max-diff=<diff> --max-diff-aln=<aln file> --max-diff-pw-aln=<alignment>* [*--max-diff-relax*] *-a,--min-am-prob=<amprob> -b,--min-bm-prob=<bmprob>* [*--kbest=<k>*] [*--better=<t>*] [*--mea-alignment*] [*--probcons-file=<file>*] *--match-prob-method=<int> --temperature=<int> --pf-struct-weight=<weight>* [*--mea-gapcost*] *--mea-alpha=<weight> --mea-beta=<weight> --mea-gamma=<weight> --probability-scale=<scale>* [*--write-match-probs=<file>*] [*--read-match-probs=<file>*] [*--write-arcmatch-scores=<file>*] [*--read-arcmatch-scores=<file>*] [*--read-arcmatch-probs=<file>*] [*--noLP*] *--anchorA=<string> --anchorB=<string>* [*--ignore-constraints*] [*<file 1>*] [*<file 2>*]

DESCRIPTION

locarna − a tool for pairwise (global and local) alignment of RNA.

LocARNA 1.6

OPTIONS

−h,−−help

Help

−V,−−version

Version info

−v,−−verbose

Verbose

Scoring parameters:

−m,−−match=<score>(50)

Match score

−M,−−mismatch=<score>(0)

Mismatch score

−−ribosum−file=<f>(RIBOSUM85_60)

Ribosum file

−−use−ribosum=<bool>(true)

Use ribosum scores

−i,−−indel=<score>(**−350**)

Indel score

−−indel−opening=<score>(**−500**)

Indel opening score

−s,−−struct−weight=<score>(200)

Maximal weight of 1/2 arc match

−e,−−exp−prob=<prob>

Expected probability

−t,−−tau=<factor>(0)

Tau factor in percent

−E,−−exclusion=<score>(0)

Exclusion weight

−−stacking
Use stacking terms (needs stack−probs by RNAfold **−p2**)

Type of locality:

−−struct−local=<bool>(false)
Structure local

−−sequ−local=<bool>(false)
Sequence local

−−free−endgaps=<spec>(−−−−)
Whether and which end gaps are free. order: L1,R1,L2,R2

−−normalized=<L>(0)
Normalized local alignment with parameter L

Controlling output:

−w,−−width=<columns>(120)
Output width

−−clustal=<file>
Clustal output

−−pp=<file>
PP output

−L,−−local−output
Output only local sub−alignment

−P,−−pos−output
Output only local sub−alignment positions

−−write−structure
Write guidance structure in output

Heuristics for speed accuracy trade off:

−p,−−min−prob=<prob>(0.0005)
Minimal probability

−D,−−max−diff−am=<diff>(**−1**)
Maximal difference for sizes of matched arcs

−d,−−max−diff=<diff>(**−1**)
Maximal difference for alignment traces

−−max−diff−aln=<aln file>()
Maximal difference relative to given alignment (file in clustalw format))

−−max−diff−pw−aln=<alignment>()
Maximal difference relative to given alignment (string, delim=&)

−−max−diff−relax
Relax deviation constraints in multiple aligmnent

−a,−−min−am−prob=<amprob>(0.0005) Minimal Arc−match probability

−b,−−min−bm−prob=<bmprob>(0.0005) Minimal Base−match probability

Special sauce options:

−−kbest=<k>(**−1**)
Enumerate k−best alignments

−−better=<t>(**−1000000**)
Enumerate alignments better threshold t

Options for controlling MEA score:

−−mea−alignment
Do MEA alignment

−−probcons−file=<file>
Probcons parameter file

−−match−prob−method=<int>(0)
Method for computation of match probs

−−temperature=<int>(150)
Temperature for PF−computation

−−pf−struct−weight=<weight>(200)
Structure weight in PF−computation

−−mea−gapcost
Use gap cost in mea alignment

−−mea−alpha=<weight>(0)
Weight alpha for MEA

−−mea−beta=<weight>(200)
Weight beta for MEA

−−mea−gamma=<weight>(100)
Weight gamma for MEA

−−probability−scale=<scale>(10000) Scale for probabilities/resolution of mea score

−−write−match−probs=<file>
Write match probs to file (don’t align!)

−−read−match−probs=<file>
Read match probabilities from file

−−write−arcmatch−scores=<file>
Write arcmatch scores (don’t align!)

−−read−arcmatch−scores=<file>
Read arcmatch scores

−−read−arcmatch−probs=<file>
Read arcmatch probabilities (weight by mea_beta/100)

Constraints:

−−noLP
No lonely pairs

−−anchorA=<string>()
Anchor constraints sequence A

−−anchorB=<string>()
Anchor constraints sequence B

−−ignore−constraints
Ignore constraints in pp−file

RNA sequences and pair probabilities:

<file 1>
Basepairs input file 1 (alignment in eval mode)

<file 2>
Basepairs input file 2 (dp dir in eval mode)

REPORTING BUGS

Report bugs to <will (at) informatik.uni−freiburg.de>.

NAME

locarna_deviation − manual page for locarna_deviation (LocARNA 1.6)

SYNOPSIS

deviation *<aln-file> <ref-aln-file>*

DESCRIPTION

locarna_deviation − compute the deviation of an alignment to a reference alignemnt

OPTIONS

<aln−file>

alignment file in clustalw format

<aln−ref−file>

reference alignment file in clustalw format

The sequences of <aln−file> have to be contained in the alignment of <aln−ref−file>.

NAME

locarna_p − manual page for locarna_p (LocARNA 1.6)

SYNOPSIS

locarna_p [*-h,--help*] [*-V,--version*] [*-v,--verbose*] *-m,--match=<score> -M,--mismatch=<score> --ribosum-file=<f> --use-ribosum=<bool> -i,--indel=<score> --indel-opening=<score> -s,--struct-weight=<score>* [*-e,--exp-prob=<prob>*] *-t,--tau=<factor> --temperature=<int> --pf-scale=<scale> -p,--min-prob=<prob> -a,--min-am-prob=<amprob> -b,--min-bm-prob=<bmprob>* [*--include-am-in-bm*] [*--write-arcmatch-probs=<file>*] [*--write-basematch-probs=<file>*] *-w,--width=<columns> -d,--max-diff=<diff> -D,--max-diff-am=<diff> --max-diff-aln=<aln file> --max-diff-pw-aln=<alignment> --fragment-match-probs=<"i j k l">* [*<bps-file 1>*] [*<bps-file 2>*]

DESCRIPTION

locarna_p − a tool for pairwise partition function alignment of RNA. Computes the partitition function and sequence and structure match probabilities.

LocARNA 1.6

OPTIONS

−h,−−help
Help

−V,−−version
Version info

−v,−−verbose
Verbose

Scoring parameters:

−m,−−match=<score>(50)
Match score

−M,−−mismatch=<score>(0)
Mismatch score

−−ribosum−file=<f>(RIBOSUM85_60)
Ribosum file

−−use−ribosum=<bool>(true)
Use ribosum scores

−i,−−indel=<score>(**−350**)
Indel score

−−indel−opening=<score>(**−500**)
Indel opening score

−s,−−struct−weight=<score>(180)
Maximal weight of 1/2 arc match

−e,−−exp−prob=<prob>
Expected probability

−t,−−tau=<factor>(0)
Tau factor in percent

−−temperature=<int>(150)
Temperature for PF−computation

−−pf−scale=<scale>(1.0)
Scaling of the partition function. Use in order to avoid overflow.

−p,−−min−prob=<prob>(0.0005)
Minimal probability

−a,−−min−am−prob=<amprob>(0.0005) Minimal Arc−match probability

−b,−−min−bm−prob=<bmprob>(0.0005) Minimal Base−match probability

−−include−am−in−bm
Include arc match cases in computation of base match probabilities

Controlling output:

−−write−arcmatch−probs=<file>
Write arcmatch probabilities

−−write−basematch−probs=<file>
Write basematch probabilities

−w,−−width=<columns>(120)
Output width

Heuristics for speed accuracy trade off:

−d,−−max−diff=<diff>(**−1**)
Maximal difference for alignment traces

−D,−−max−diff−am=<diff>(**−1**)
Maximal difference for sizes of matched arcs

−−max−diff−aln=<aln file>()
Maximal difference relative to given alignment (file in clustalw format))

−−max−diff−pw−aln=<alignment>()
Maximal difference relative to given alignment (string, delim=&)

Computed probabilities:

−−fragment−match−probs=<"i j k l">() Requests probabilities for the match of fragments [i..j] and [k..l].
Accepts a ';' separated list of ranges.

RNA sequences and pair probabilities:

<bps−file 1>
Basepairs input file 1

<bps−file 2>
Basepairs input file 2

REPORTING BUGS

Report bugs to <will (at) informatik.uni−freiburg.de>.

NAME

locarnap_fit − manual page for locarnap_fit (LocARNA 1.6)

SYNOPSIS

locarnap_fit [*-h,--help*] [*-V,--version*] [*-v,--verbose*] *-d,--delta=<float>* *-b,--beta=<float>* [*--once-on*] [*--all-values*] *<file>*

DESCRIPTION

locarnap_fit − Fit a two step function to a data series.

OPTIONS

−h,−−help
: This help

−V,−−version
: Version info

−v,−−verbose
: Verbose

−d,−−delta=<float>(0.5)
: Penalty for state change

−b,−−beta=<float>(6)
: Inverse temperature

−−once−on
: Fit a signal that is on only once

−−all−values
: Show all function values of signal (instead of only ranges)

<file>(profile.dat)
: Input file with sequence of numbers

REPORTING BUGS

Report bugs to <will (at) informatik.uni−freiburg.de>.

NAME

locarnap−predict−and−plot.pl

SYNOPSIS

locarnap−predict−and−plot.pl [options]

DESCRIPTION

Performs boundary and reliability prediction and draws all reliability plots according to annotation file. The script is usually used after generating alignments with locarnap−realign−all.pl as third step in a pipeline for refining RNAz hits with LocARNA-P.

OPTIONS

−−help
Brief help message

−−man
Full documentation

−−test
Test

−−output−dir=d
Output directory (def=Relplots)

−−dont−plot
Skip plotting, only output

−−show−sw
Show the structure weight in the plot

−−revcompl
Draw for reverse complement (3’−5’)

−−write−subseq
Write the subsequence of fit

−−output−format=f
Output format (f = pdf or png, def=pdf)

By default plots are written to directory Relplots. The predictions are written to standard out as a table. A line of the table contains of the locus name, start,end, and orientation of the RNAz prediction, the LocARNA prediction and the first annotation, the on and off value of the fit, and the background and hit reliability.

NAME

locarnap−realign−all

SYNOPSIS

locarnap-realign-all [options] <annotation−file>

DESCRIPTION

Calls mlocarna on sequence sets in Realign-Sequences as generated by a call to locarnap−revisit−RNAz−hits.pl. The script is usually used as second step in a pipeline for refining RNAz hits with LocARNA-P.

OPTIONS

−−help
Brief help message

−−man
Full documentation

−−test
Test only. Jobs are not run or submitted to SGE!

−−revcompl
Realign reverse complement

−−run−locally
Run the realignment on the local machine (without the use of SGE).

−−threads=k
Use <k> threads for multicore support.

Writes result files to Alignment-Results, takes alignment jobs from annotation file

Unless option −−run−locally is given, distribute jobs to SGE-cluster, where we assume that the script is run on a submission host!

NAME

locarnap−revcomp.pl

SYNOPSIS

locarnap−revcomp.pl [options] <fasta files>

DESCRIPTION

locarnap−revcomp.pl generates a file of reverse complement of sequences. The script is intended for the use in the LocARNA-P pipeline for refining RNAz hits.

OPTIONS

−−help
Brief help message

−−man
Full documentation

Produce reverse complement of sequences in the given fasta files and write a corresponding file with −rc suffix.

The script is used to reverse-complement the sequences in Realign-Sequences as generated by locarnap-revisit-RNAz-hits. Assume simple fasts format (only one line per sequence after sequence header). Reverse left_context/right_context annotation in sequence names.

NAME

locarnap−revisit−RNAz−hits

SYNOPSIS

locarnap-revisit-RNAz-hits [options]

DESCRIPTION

Revisits the hits of the fly RNAz screen and prepares input data for realigning all loci with locarnap−realign−all.pl. The script is usually used as first step in a pipeline for refining RNAz hits with LocARNA-P.

OPTIONS

−−help
Brief help message

−−man
Full documentation

−−all
Extract all loci (in contrast, the default mode selects only loci that overlap with Flybase ncRNA annotation).

−−random <n>
Special mode: draw n random instances

−−context <c>
Extract with maximal context of c columns upstream and downstream.

In default mode, determine the RNAz hits with dmel flybase annotation. In random mode, draw n random hits. With flag −−all select all hits. For the selected hits get the annotation and the position in the pecan alignment. Determine the sequences that are well-aligned to the dmel sequence, obtain these sequences with genomic context (option −−context, default=100).

Goal: use these sequence for realining by locarna. From the locarna reliability profile, determine boundaries of the ncRNA. Compare to the flybase annotation and RNAz boundaries.

NAME

locarnate − manual page for locarnate version 0.9

SYNOPSIS

locarnate [*OPTIONS*] *INPUT*

DESCRIPTION

Calculates a multiple local RNA sequence structure alignment of the sequences given by INPUT

OPTIONS

Generic options:

−−help display this help and exit

−−version
output version information and exit

−v, **−−verbose**
explain what is being done

Output:

−R, **−−results**
directory for results (all formats and intermediate data)

−C, **−−clustal**
filename for results in clustal format

−F, **−−fasta**
filename for results in fasta format

−S, **−−stockholm**
filename for results in stockholm format

Configuration:

Locarna:

−m, **−−match**
Match score

−M, **−−mismatch**
Mismatch score

−−ribosum−file
Ribosum file for base and arc match scores [/home/will/share/locarna/Matrices/RIBOSUM85_60]

−i, **−−indel**
Indel score

−−indel−opening
Indel opening score

−s, **−−struct−weight**
Maximal weight of 1/2 arc match

−t, **−−tau**
Tau factor in percent

−E, **−−exclusion**
Exclusion weight

−−no−stacking
Turn of stacking terms

−−no−struc
Turn of structure locality

−−no−seq
Turn of sequence locality

−p, **−−min−prob**
Minimal probability

−D, **−−max−diff−am**
Maximal difference for sizes of matched arcs

T−coffee:

−d, **−−double**
double weights for edges with basepairs

AUTHOR

Written by Wolfgang Otto.

NAME

MLocarna − multiple alignment of RNA (LocARNA 1.6)

SYNOPSIS

mlocarna [options] <fasta file>

DESCRIPTION

MLocarna computes a multiple sequence-structure alignment of RNA sequences.

OPTIONS

Controlling Output

−−tgtdir

Target directory. All output files are written to this directory. Per default the target directory is generated from the input filename by replacing suffix fa by (or appending) out.

−v, −−verbose

Turn on verbose ouput.

−−moreverbose

Be even more verbose

−q, −−quiet

Be quiet.

−−keep−sequence−order

Preserve sequence order of the input in the final alignment. Affects output to stdout and results/result.aln.

Controlling pairwise alignments

−−noLP / −−LP

Disallow/Allow lonely pairs (default: Disallow).

−−free−endgaps

Allow free endgaps. (Corresponds to pairwise locarna option −−free−endgaps ‘‘++++’’.)

other locarna options

Many of the options of locarna, the program for pairwise alignment, will work as well. This allows controlling scoring, locality, and speed/acccuracy trade off (heuristics). **Please see** `locarna −h` or the manpage of locarna.

Controlling guide tree construction

−−treefile

Guide tree file. If given, the computation of the guide tree is skiped and the given one is used.

−P, −−tree−min−prob=<f>

Minimal prob for constructing guide tree.This probability can be set separately for the all−2−all comparison for constructing the guide tree and the progressive/iterative alignment steps.

−−skip−pp

Skip computation of pair probs if the probabilities are already existing. Non-existing ones are still computed.

Controlling multiple alignment

−−max−diff−aln=<file>

Restrict maximal difference to the alignment in <file> in clustalw format (difference given by −−max−diff). Use this option for constrained re-aligning.

−−probabilistic

Score alignments using match probabilities that are computed by a partition function approach. This makes possible to consistency-transform the probabilities (option −−consistency−transform) and to compute reliabilities. Reliabilities can also be used for iterating the alignment with reliably aligned base pairs as structural constraints (option −−it−reliable−structure).

−−consistency−transformation
Apply probabilistic consistency transformation (only possible in probabilistic mode).

−−iterate
Refine iteratively after progressive alignment. Currently, iterative refinement optimizes the SCI or RELIABILITY (not the locarna score)! Iterative refinement realigns all binary splits along the guide tree.

−−iterations=<num>
Refine iteratively for given number of iterations (or stop at convergence).

−−extlib
Use library extension for base pair probabilities (experimental/not functional).

−−it−reliable−structure=<num>
Iterate alignment <num> times with reliable structure. This works only in probabilistic mode, when reliabilities can be computed.

Options for probabilistic mode

−−pf−only−basematch
Use only base match probabilities (no base pair match probabilities).

−−pf−scale=<scale>
Scale of partition function; use for avoiding overflow in larger instances.

−−fast−mea
Compute base match probabilities using Gotoh PF-algorithm.

−−mea−alpha
Weight of unpaired probabilities in fast mea mode.

−−mea−beta
Weight of base pair match contribution in probabilistic mode.

−−mea−gamma
Reserved parameter for fast-mea mode.

−−mea−gapcost
Turn on gap penalties in probabilistic/mea mode (default: off).

−−no−write−bm−probs
Don’t write base match probabilities to files in target dir.

−−no−write−am−probs
Don’t write arc match probabilities to files in target dir.

Special modes of operation

−−dp−cache=<dir>
Use directory <dir> as cache for dp files.

−−only−dps
Compute only the missing dp files, don’t align.

−−evaluate=<file>
Evaluate the given multiple alignment (clustalw aln format, or use −−eval−fasta)

−−eval−fasta
Assume that alignment for evaluation is in fasta format

Constraints

−−ignore−constraints
Ignore constraints even if given.

Rna folding (RNAfold/RNAplfold)

−−plfold−span=span
Use RNAplfold with span

−−plfold−winsize=ws
Use RNAplfold with window of size ws (default=2*span)

−−rnafold−parameter=<file>
Parameter file for RNAfold (RNAfold's −P option)

Multithreading

−−threads=<num> or **−−cpus=<num**>
Use <num> threads in parallel (support multicore/processor).

Getting Help

−−help
Brief help message

−−man
Full documentation

The sequences are given in input file <file> in mfasta format. All results are written to a target directory <dir>. If the file tree is given, contained tree (in NEWICK-tree format) is used as guide tree for the progressive alignment. The final results are collected in <tgtdir>/results. The final multiple alignment is <tgtdir>/results/result.aln.

Whenever parameters are not specified explicitly, we use the locarna defaults (please see `locarna −h` or the manpage of locarna).

AUTHOR

Sebastian Will

NAME

reliability−profile.pl

SYNOPSIS

reliability−profile.pl [options] <locarna−output−dir>

DESCRIPTION

reliability−profile.pl generates a reliability profile plot of a multiple alignment generated by mlocarna −−probabilistic.

OPTIONS

−−help
Brief help message

−−man
Full documentation

−−seqname=seqname
Project to sequence name

−−dont−predict
Turn off predicting. (def=on)

−−fit−penalty=penalty
Penalty for on/off switching in fit

−−fit−once−on
Restrict fitting to being exactly once on

−−title=title
Title of plot

−−out=filename
Output filename

−−offset=pos
Offset of sequence in genome

−−signals=list
List of (from,to,orientation) triples. Show signals in plot and compared infered signal to them. Give list as string ‘‘from0 to0 orientation0;from1 to1 orientation1 ...’’ Specify multi-range signals by from0a to0a from0b to0b ...

−−structure−weight=w
Weight of structure against sequence (1.0)

−−show−sw
Show the influence of structure weight in the plot

−−beta=f
Inverse temperature beta

−−dont−plot
Skip plotting, only output

−−write−R−script
Write the R script

−−revcompl
Plot and fit a reverse complement

−−write−subseq
Write the subsequence of fit

−−output−format=f
Output format (f = pdf or png, def=pdf)

−−show−fitonoff
Show the on/off values for the fit

The target directory of mlocarna is required for obtaining the reliability information. Optionally, the reliability plot will be projected to the sequence with given sequence name (or containing ‘‘seqname’’ as prefix).