Accurate and efficient reconstruction of deep phylogenies from structured RNAs – Supplemental Material –

Roman R. Stocsits [e], Harald Letsch [e], Jana Hertel [a], Bernhard Misof [f], Peter F. Stadler [a,c,b,d]

[a] *Bioinformatics Group, Dept. of Computer Science, Interdisciplinary Center for Bioinformatics, University of Leipzig, Härtelstraße 16-18, D-04107 Leipzig, Germany*

[b] *Department of Theoretical Chemistry, University of Vienna, Währingerstraße 17, A-1090 Wien, Austria*

[c] *RNomics Group, Fraunhofer Institut für Zelltherapie und Immunologie, Deutscher Platz 5e, D-04103 Leipzig, Germany*

[d] *Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, USA*

[e] *Zoologisches Forschungsmuseum Alexander Koenig, Bonn*

[f] *UHH Biozentrum Grindel & Zoologisches Museum, Hamburg, Germany*

1 RNAsalsa – Implementation, availability, requirements

`RNAsalsa` is written in `C`. The `C` source code and pre-compiled executables for various platforms, as well as a detailed manual and a tutorial may be downloaded from `http://www.rnasalsa.zfmk.de/` and from `http://www.bioinf.uni-leipzig.de/Software/RNAsalsa`.

The *manual* is also part of this *Supplemental Materials* section (see below). The software is distributed free of charge under the license of the `Vienna RNA Package`. The program produces extensive text and graphical output describing individual and consensus structures, the constraints used throughout the computation, as well as alignments in various machine-readable forms for further processing. The current implementation is applicable to quite various data sets of RNA sequences: `RNAsalsa` can utilize complete genes as well as any RNA sequence fragments given an applicative structural constraint. Regarding data set size, `RNAsalsa` can handle data up to an extent of several hundred LSU rRNAs on standard PC hardware. Memory consumption is mostly not critical, time consumption explicitly depends on the data setup.

RNAsalsa is command line based so far, a graphical user interface is currently work in progress.

A list of the main alignment and stringency parameter values of RNAsalsa is provided in *Table 1*. The stringency thresholds for generating and optimizing the structure constraints and the settings for gap penalties and substitution costs have pre-defined defaults that may be adapted to a certain data set by the user. Figure 1 gives a further overview about the subprocesses of RNAsalsa beside that already presented in the main text.

Table 1
Overview of the parameter values in RNAsalsa.

alignment parameters[1]		stringency setting defaults[2]	
match score	10	switch -s1[3]	0.6
mismatch score	0	switch -s2[4]	0.6
gap opening penalty	-11	switch -s3[5]	0.6
gap extension penalty	-3		

[1] Parameter values for matches, mismatches and gap penalties in RNAsalsa alignments.
[2] Default stringency values for secondary structure adoption, may be adapted by the user as well for individual constraints as for the final consensus structure.
[3] Optional switch *s*1: minimum frequency of base pairing occurrence in the first constraint adaptation.
[4] Optional switch *s*2: the stringency setting for the majority voting procedure to obtain an individual constraint by the fusion of pairwise alignment folding results.
[5] Optional switch *s*3: stringency settings for the final consensus structure extraction process.

2 Materials and methods

2.1 Structure prediction comparisons

The performance of the RNAsalsa folding algorithm on ribosomal RNA sequences was compared with that of three other structure prediction methods. MXSCARNA [1] employs the McCaskill algorithm to calculate base pairing probabilities and considers potential stem information in the subsequent alignment process. RNAfold [2] produces individual secondary structures of RNA sequences by free energy and RNAalifold [3] generates a consensus structure for aligned RNA sequences by a combination of free energy and covariation.

Structure prediction analyses of the mammalian 16S rRNA secondary structure was conducted and the resulting structures were compared to the mammalian 16S reference structure model, adopted from [4]. RNAalifold and MXSCARNA provide consensus secondary structures of the aligned data set. Accuracy of the particular programs was measured and compared to that of RNAsalsa by the occurrence of correctly predicted helices in the consensus structures. Both, the RNAalifold and MXSCARNA methods were applied by using default parameter settings. In contrast, the RNAfold algorithm produces

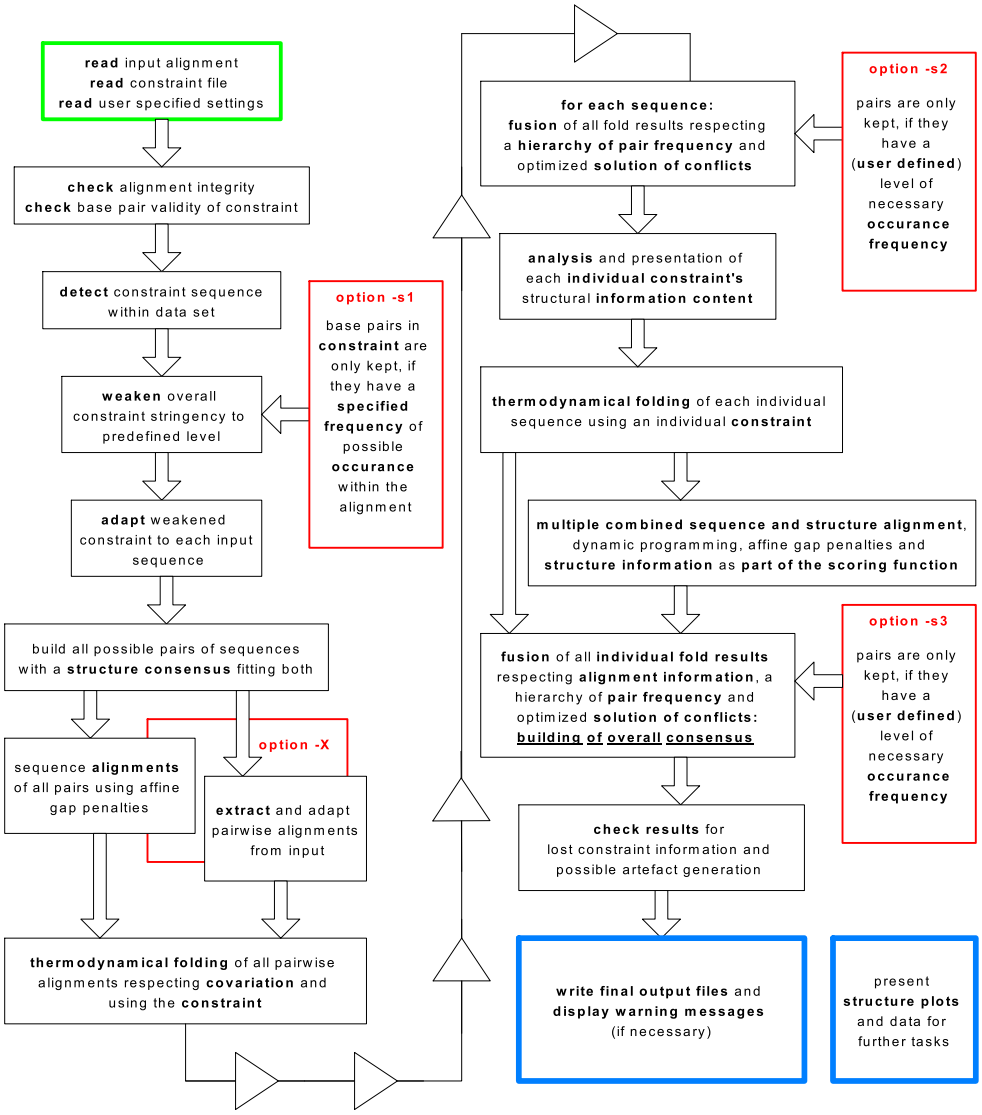

Fig. 1. The algorithmic concepts throughout the workflow of `RNAsalsa` as a graphical representation. See main text for an alternative representation.

individual secondary structures for each sequence. We compared these individual structure predictions of `RNAfold` to those of `RNAsalsa`, by measurement of the pairwise distances of each taxons folding results to the reference structure. Distance measurements were performed with the `RNAdistance` software, which is, like `RNAfold` and `RNAalifold`, part of the `Vienna RNA Package` [2].

As RNA secondary structures can be represented as trees [2,5] and differences between structures can be displayed as the tree edit distance *dt*, the minimum number of steps to change one tree into another can serve as a measure for similarity. As the nucleotide composition of the underlying primary sequence has no influence on the tree edit distance *dt*, it is a pure geometric distance

measure. Structure distance values were compared with a paired sampled t-Test. For both the `RNAfold` and `RNAdistance` analyses, default parameter settings were used.

2.2 Sequence data – applications in phylogeny reconstruction

In order to demonstrate the potential of the `RNAsalsa` workflow in phylogenetic contexts we performed short exemplary analyses. The data sets used and the corresponding references are listed in the following.

Two different datasets, comprising both nuclear and mitochondrial ribosomal RNA genes were used: The mitochondrial 12S and 16S rRNA [6] of 26 mammalian species together with the structure constraints of *Bos taurus* 12S and 16S rRNAs from the European Ribosomal Database (ERD) were used for investigating the monophyly of primates. As an independent structural model, the 16S RNA mammalian consensus model was used [4] to validate the results.

The second dataset consists of six nearly complete echinoderm 28S rRNAs covering all five major groups. As a constraint structure, we used the 28S model of *Thalia democratica* from the ERD. All sequences were downloaded from the `NCBI Genbank` database (see *Tables 2 and 2.2*).

2.3 Data set compilation – Primates

As mentioned above, we compiled rRNA data consisting of both nuclear and mitochondrial ribosomal RNA genes. The mitochondrial 12S and 16S data (26 mammalian taxa) rely on a study by [6], who tested the phylogenetic signal of the whole mitochondrion genomes for the placement of the genus *Tarsius* within primates. Additionally, we employed the mammalian 16S rRNA secondary structure model, published by [4], as a reference structure and independent source of secondary structure information to compare different structure prediction algorithms. As a structure constraint for the RNAsalsa alignment, we used the 12S and 16S structure models of *Bos taurus*, retrieved from the European Ribosomal Database [7,8]. See *Table 2* for a compilation of all data used.

2.4 Data set compilation – Echinoderms

The second data set comprises 28S rRNA sequences of six echinoderm taxa, representing all major groups of this deuterostome phylum and the urochor-

Table 2
The first data set comprises mitochondrial 12S and 16S rRNA sequences of 26 mammalian taxa.

Order	Species	GenBank Acc.	Source
Monotremata	Ornithorhynchus anatinus	X83427	Janke et al. (1996)
Didelphimorphia	Didelphis virginiana	Z29573	Janke et al. (1994)
Cingulata	Dasypus novemcinctus	Y11832	Arnason et al. (1997)
Chiroptera	Artibeus jamaicensis	AF061340	Pumo et al. (1998)
	Pteropus dasymallus	NC_002612	Nikaido et al. (2000)
Artiodactyla	Bos taurus	J01394	Anderson et al. (1982)
	Sus scrofa	AJ002189	Ursing & Anderson (1998)
Cetacea	Balaenoptera physalus	X61145	Arnason et al. (1991)
Perissodactyla	Equus caballus	X79547	Xu & Arnason (1994)
Carnivora	Felis catus	U20753	Lopez et al. (1996)
	Canis familiaris	U96639	Kim et al. (1998)
	Halichoerus grypus	X72004	Arnason et al (1993)
	Phoca vitulina	X63726	Arnason & Johnsson (1992)
Rodentia	Mus musculus	J01420	Bibb et al. (1981)
	Rattus norvegicus	X14848	Gadaleta et al. (1989)
Scadentia	Tupaia belangeri	AF217811	Schmitz et al. (2000)
Primates	Nycticebus coucang	AJ309867	Schmitz et al. (2000)
	Tarsius bancanus	AF348159	Schmitz et al. (2002)
	Cebus albifrons	AJ309866	Schmitz et al. (2000)
	Macaca sylvanus	AJ309865	Schmitz et al. (2000)
	Papio hamadryas	Y18001	Arnason et al. (1998)
	Hylobates lar	X99256	Arnason et al. (1996a)
	Pongo pygmaeus	D38115	Horai et al. (1995)
	Gorilla gorilla	D38114	Horai et al. (1995)
	Pan troglodytes	D38113	Horai et al. (1995)
	Homo sapiens	X93334	Arnason et al. (1996b)

date species *Thalia democratica* as an outgroup taxon. We included only taxa, which are represented by 28S rRNA sequences extending in, at least, 3500 bp to ensure a complete or almost complete 28S rRNA gene. As a structural constraint, we used the 28S model of *Thalia democratica* from the European Ribosomal Database. The structures taken from that source are coded in a proprietary `DCSE` format and had to be translated into the dot-bracket format using the program `extractfromdcse` from the `PHASE` package [9]. *Table 2.2* gives an overview of the applied data sets.

2.5 Alignment and Maximum Likelihood analyses

`MAFFT` was used with the L-INS-i option [10] and the default settings for gap opening (1.53) and gap extension (0.23) penalties. `ClustalW` was also

Table 3
The second data set comprises 28S rRNA sequences of six echinoderm taxa.

Class/Phylum	Species[1]	GenBank Acc.	Source
Urochordata			
Thaliacea	Thalia democratica[2]	AF158725	Mallat & Sullivan (1998)
Echinodermata			
Crinoidea	Florometra serratissima	AF212168	Winchell et al (2002)
Ophiuroidea	Ophioderma cenereum	AY859643	Mallat & Winchell (2002)
Asteroidea	Asterias forbesi	AF212169	Winchell et al (2002)
Echinoidea	Strongylocentrotus purpuratus	AF212171	Winchell et al (2002)
	Arbacia punctulata	AY026367	Medina et al. (2001)
Holothuroidea	Cucumaria salma	AF212170	Winchell et al (2002)

[1] We have chosen six echinoderm taxa representing all major groups of this deuterostome phylum.
[2] The urochordate species *Thalia democratica* was used as an outgroup taxon.

used with default parameter values for gap opening (15.0) and gap extension (6.66). As pre-alignments for the `RNAsalsa` analyses, we employed the `ClustalW` alignments of the mammalian and the echinoderm data set. The `RNAsalsa` run was performed using most of all defaults exept some threshold values that effect the generation of the first overall and later individual structure constraints.

- option $-s1$ was set to 0.51
- option $-s2$ has been let at the default setting (0.6)
- option $-s3$ was also set to 0.51

Optional switches $-s1$ and $-s3$ have been relaxed in our case to obtain optimized structure information for that certain data set.

After the alignments, the 12S and 16S sequences of the mammalian data set were concatenated and the quality of both the echinoderm and mammalian alignments was evaluated with the program `Aliscore` [11], a new method to identify ambiguously aligned regions in multiple sequence alignments. The algorithm and recommended settings of `Aliscore` are described in detail in [11]. To put it in a nutshell: within a sliding window, sequences are assumed unrelated if the observed score is not better than 95 % of scores of random sequences of similar window size and character composition generated by a MC resampling process. All positions within the window receive a positive sign in case of non-random similarity or otherwise a negative sign. For each pairwise comparison a quality profile is generated and finally a consensus quality profile is calculated. Pairwise comparisons can be done randomly or guided by a tree. `Aliscore` generates a list of all putative randomly similar sections. No distinction is made between random similarity caused by mutational saturation and alignment ambiguity. Both have effects on tree reconstruction and exclusion of the identified characters is recommended. As `Aliscore` is currently not able to detect base pairings, ambiguously aligned positions which are part

of a helix have been omitted and the corresponding nucleotides have been retained and handled as single characters in further tree reconstruction. For gap treatment, window size and pairwise comparisons, the following settings of `Aliscore` were used:

- window size was six positions
- gaps were treated as ambiguous characters
- pairwise comparisons were guided by a neighbour joining tree representing the distances of the included taxa

The guiding tree distances were corrected by a General Time Reversible (GTR), [12] model and a gamma distribution [13], using PAUP* 4.10b [14]. Maximum Likelihood analyses were conducted with the Pthreads-parallelized version of `RAxML` 7.0.4 [15,16]. Nucleotide substitution was displayed by the GTR model with all model parameters estimated from the data and four categories of gamma distributed rates across sites. In the mammalian data set, 12S and 16S rRNA sequences were handled as individual partitions. Maximum Likelihood bootstrap percentages were obtained after 1000 replications.

2.6 Model comparisons

To test, if specific RNA substitution models influence the topology and node support of phylogenetic trees, we additionally compared application of these models with simple DNA model application on the mammalian rRNA data. Recently, RNA substitutions models are not implemented in any Maximum Likelihood software, but in specific Bayesian inference programs. Consequently, we performed an additional Bayesian analysis with incorporation of site specific interdependencies with a parallel version of MrBayes 3.1.2 [17,18], which employs a deviation of the Schoeniger & von Haeseler model [19] to account for character covariance. The concatenated mammalian `RNAsalsa` alignment was subdivided into four partitions (12S loops, 12S stems, 16S loops and 16S stems). For each subset, the GTR model was used and different substitution rates were covered by a gamma distribution. In loop regions, the $GTR + \Gamma$ model was employed in the standard DNA (4by4) mode, whereas in stem regions, the doublet RNA mode framework was applied. Analyses were run with two different Metropolis coupled Markov runs (four chains, 12.000.000 generations and every 100th generation sampled), which resulted in 240.000 sampled trees. A total of 12.000 trees were discarded as "burn-in" trees for each run separately. Posterior probabilities were calculated, using a 50% majority-rule consensus tree, from the concatenated set of trees, generated in all MCMC runs. Subsequently, we analyzed the mammalian data set, aligned with RNAsalsa and applied the same tree reconstruction setup, without considering doublet models and employing in all regions the $GTR+\Gamma$ model in the standard DNA

mode.

3 Results

3.1 Secondary structure prediction

We compared the performance of structure prediction functionality to three other relevant methods: `MXSCARNA` [1] computes pairing probabilities and considers potential stem information in the subsequent alignment process, `RNAfold` [2] produces individual secondary structures of RNA sequences, and `RNAalifold` [3] generates the consensus structure for a given input alignment. We further compared `RNAfold` predictions with `RNAsalsa`'s individual predictions ψ^i, while the `MXSCARNA` and `RNAalifold` results are compared with `RNAsalsa`'s final consensus structure ω.

The `RNAsalsa` secondary structure model for the mammalian 16S rRNA sequences is highly congruent to the *Bos taurus* reference model proposed by [4], see the graphical representation of the reference structure (*Fig. 2*).

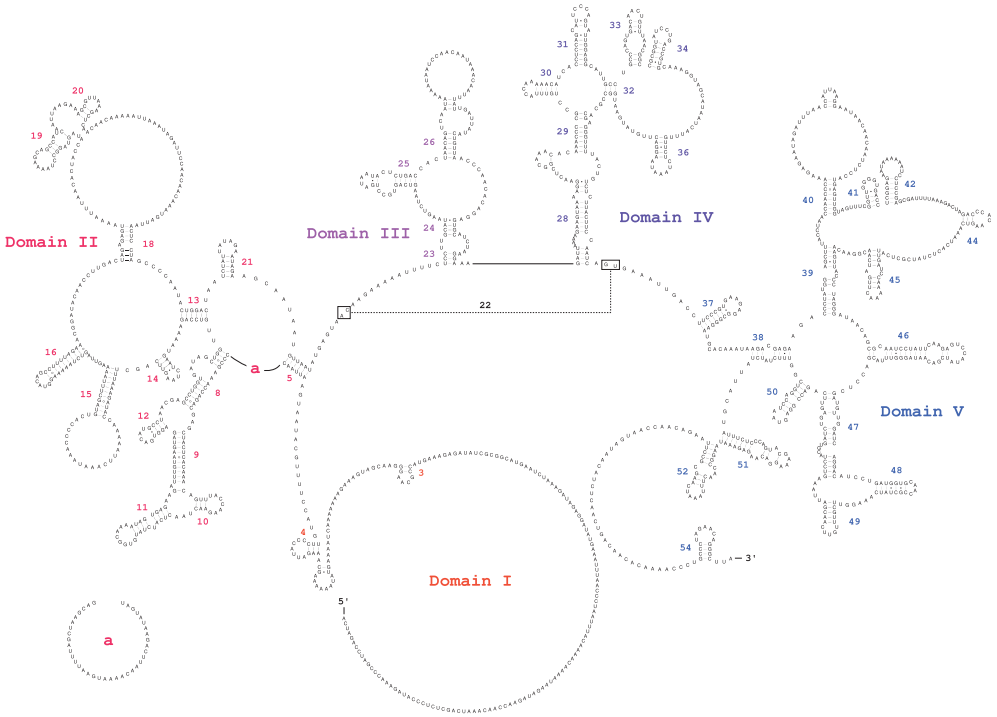

Fig. 2. Graphical representation of the mammalian 16S rRNA secondary structure model, generated by `RNAsalsa`. The structure string is plotted on the *Bos taurus* sequence.

44 of the 52 helices within the conserved structure core are correctly predicted by RNAsalsa. The remaining discrepancy is likely not a weakness of RNAsalsa but reflects a greater variability of mammalian 16S rRNA structures than present in the data set used to construct the reference model. MXSCARNA and RNAalifold capture only 27 and 23 helices, resp. In contrast to RNAsalsa, they fail in particular to detect long range interactions.

Furthermore, to demonstrate the capabilities of constraints in thermodynamical folding algorithms we compared the results of unconstrained foldings by the RNAfold software [2] only with those generated by RNAsalsa. In both cases, we used the same thermodynamical parameter sets and algorithms as implemented in RNAfold, but the only difference is the usage of folding constraints in the case of RNAsalsa. These structure constraints are automatically generated and optimized for each RNA sequence within the data set.

RNAsalsa's predictions of the individual structures always use an individual structure constraint that was optimized by RNAsalsa's procedure itself up to that point. Therefore, they can match the reference model much better than thermodynamic folds by RNAfold, which have been calculated without supporting constraints. See *Fig. 3* for an illustrating set of comparisons performed on 16S rRNA.

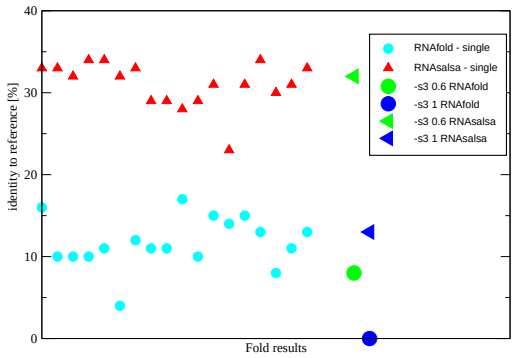

Fig. 3. We compared RNAfold predictions with RNAsalsa's individual predictions ψ^i (done on 16S rRNA). The larger dots and triangles to the right represent consensus structures under filtering conditions analogous to those of RNAsalsa's switch $-s3$ (see main text). RNAsalsa's ability to predict secondary structures outperforms RNAfold. The only difference in folding algorithms and parameter sets is the addition of automatically generated and optimized individual constraints.

3.2 Benchmarks of Alignments

Simulated data were generated as reference alignments using the RNAsim [20]. We used RNAsim's various values between 10 and 100000 for the "branch scaling" option -s, whereas the default was used for all other options. Since the

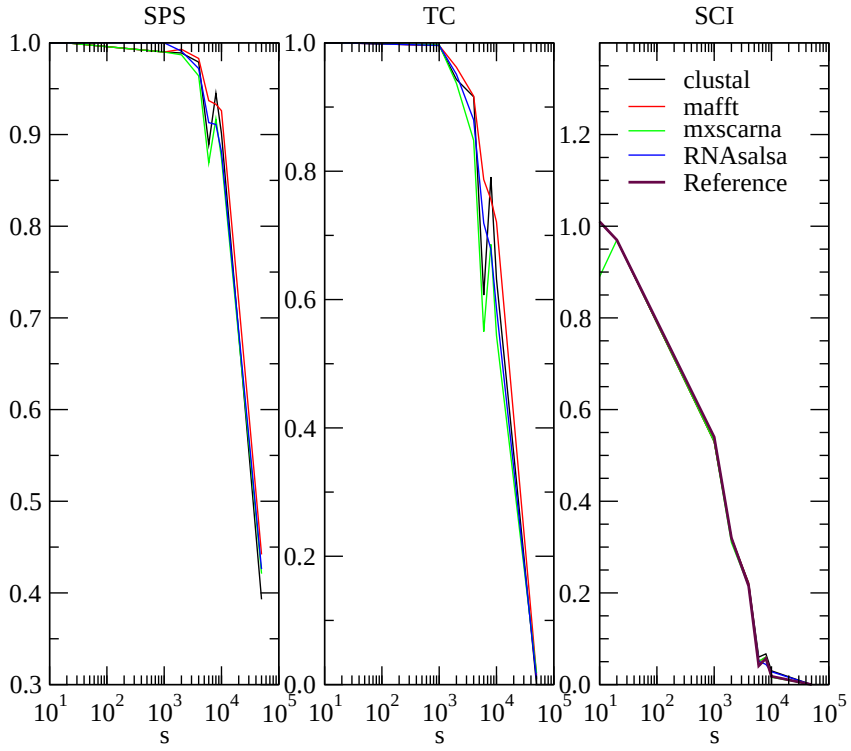

Fig. 4. Results of `RNAsim` simulations for different branch scaling parameters s. The performance of the different methods does not differ substantially between the programs. The sequence diverengence has a dominating influence on the alignment quality.

branch scaling parameter has a strong influence on both sequence and structure, we systematically analysed how various alignment algorithms behave as a function of divergence, see Fig. 4.

The `BRAliBase-II` set of structural alignments (`http://projects.binf.ku.dk/pgardner/bralibase/bralibase2.html`) [21] was used for comparing the results of different alignment programs. It covers group II introns, 5S rRNA, tRNA, U5, and SRP RNA [1] . The results are provided in Fig. 5. We observe no major differences in the performance metrics. As in the case of the simulated data, the alignments become more divergent between methods as the problems become harder.

[1] Following [21], we did not use the SRP RNA alignments.

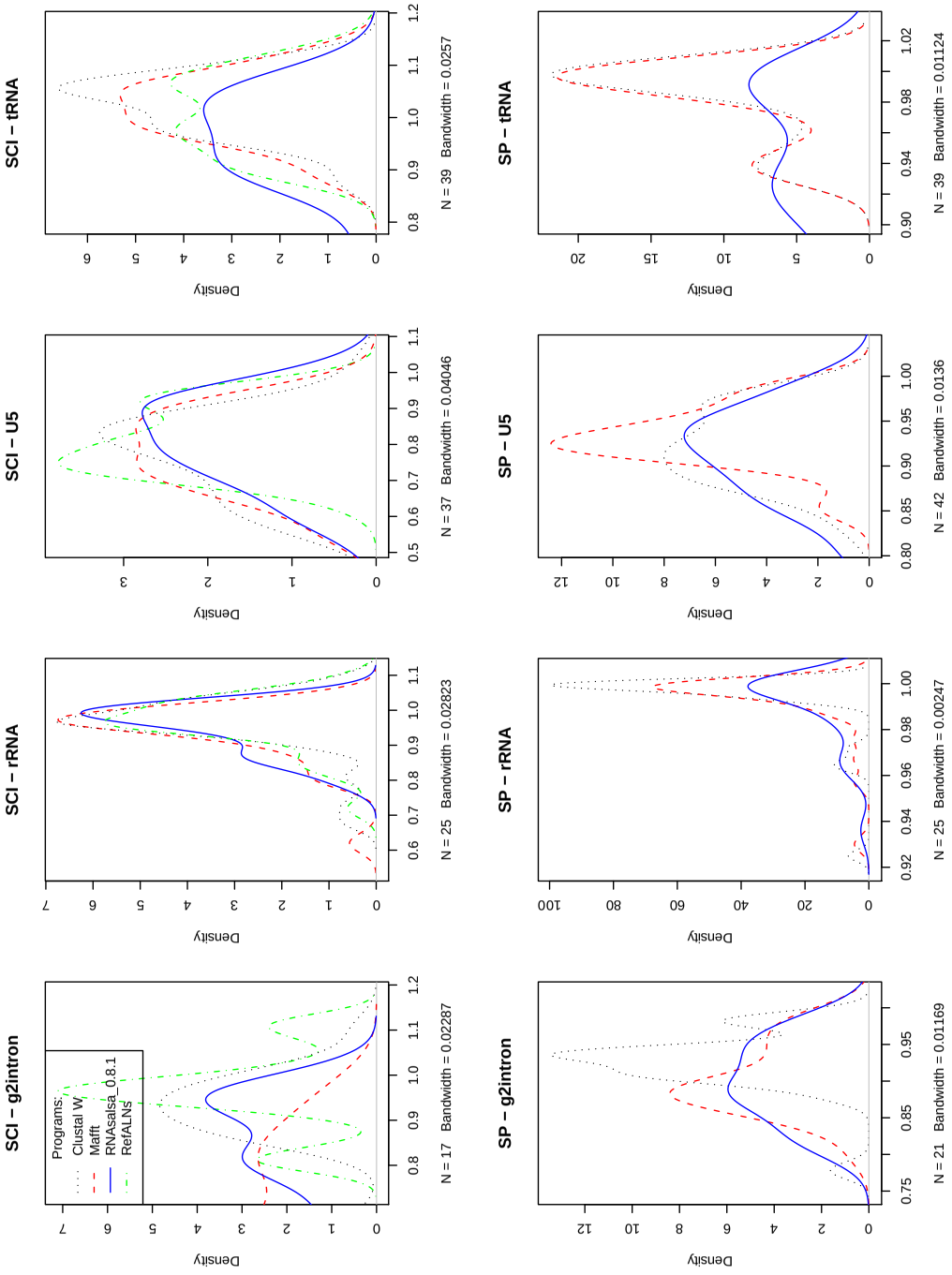

Fig. 5. Results of SCI and SP scores for the four different programs compared to the BRAliBase-II structural alignments as reference.

This manual documents version 0.8.1 of the RNAsalsa software.

4 Installation

The software may be downloaded from

www.rnasalsa.zfmk.de or

www.bioinf.uni-leipzig.de/Software/RNAsalsa

For installing from source code unpack and compile the downloaded .tar.gz or .zip file. For instance, in the case of Linux based operating systems, just type

```
tar zxvf RNAsalsa_xxx.tar.gz
cd RNAsalsa_xxx
make
```

That's all. Note, that you need an adequate compiler and the make tool pre-installed. Downloadable pre-compiled binaries for Win32 and MacOS are ready for usage.

5 The input requirements ...

(1) a known structure constraint that can serve as an external source of structure information and as a fundament of the structure building process.
(2) an input alignment of your data. The sequence underlying the structure constraint must be part of the alignment.

... and in detail:

- The input alignment is a standard `FASTA` file. Please note that all sequences within the data set must have an unique initial identifier. Thus, two sequences named '`> Animal 1 (the cute one)`' and '`> Animal 2 (the ugly one)`' will give an error, because the determining initial identifier '`Animal`' is the same for both. Additionally, taxon names must not contain any characters that are not covered by the alphanumeric ASCII table.
- The constraint file is a text file consisting of exactly 3 lines: the name, the nucleotide sequence, and the structure encoding string in dot-bracket format. The dot-bracket conventions mean that each base pair is indicated by a left-handed bracket and a corresponding right-handed bracket, a dot is not obligatory structural but might be, and 'x' means that the considered nucleotide must never be paired. All structure encoding characters lie exactly under the sequence position they encode (in other words, there cannot be any line breaks within either the sequence or the structure string).

6 The structural constraint file

An arbitrary example of a functional constraint text file looks like the following:

```
my_constraint_sequence
AGGCCUAUGCAAACCGUUUGCGGGACGGCU
...((....(((....)).)..))..xx..
```

Please note that the file must consist of **exactly 3 lines**, regardless of the length of the sequence, and that the lines with sequence and structure information must match each other. There must be not any further line breaks or empty lines. Further, the sequence which defines the structure constraint, must be part of your initial input FASTA alignment.

7 The set of switches and options

Typing just 'RNAsalsa' or 'RNAsalsa.exe' (on MS Windows systems) without any further input will display a short help:

```
Usage: RNAsalsa [-h] [-v] [-X] [-p] [-s1] [-s2] [-s3] -c FILE -i FILE
   or: RNAsalsa [-h] [-v] [-s3] -a FILE

-c FILE  Necessary switch and input file containing the constraint.
             1st line name, 2nd line sequence, 3rd line structure.
-i FILE  Necessary switch and input alignment file.
             Input alignment in CLUSTALW or FASTA format.
-p      Switch OFF all PostScript output of structures (default is ON).
-s1      Stringency setting for initial constraint weakening.
-s2      Stringency setting for merging of pairwise consensus structures.
-s3      Stringency setting for building the final consensus structure.
             Stringency values lie between 0.0 (lowest) and 1.0 (highest).
-X       All pairwise alignments from input, no internal sequence alignments.
-a FILE  Re-read existing RNAsalsa output, do multiple alignment only.
             Folding output file format must equal 'SALSA_fold_results.txt'.
-h       Show this help message and exit.
-v       Show version information and exit.
```

8 The process of RNAsalsa – some more details

A typical `RNAsalsa` run using the default instruction set is started by

```
RNAsalsa -i <input_alignment> -c <constraint_file>
```

`RNAsalsa` generates graphical `PostScript` output of all individual structure predictions and the adapted/extended constraint for each sequence in the data set, and the consensus structure based on the final alignment. By adding the switch `-p` to the `RNAsalsa` command line you can inhibit the generation of those graphics.

The switches `s1`, `s2`, and `s3` are very important control parameters with high impact on the `RNAsalsa` run and your results. Therefore, the setting of those `-s` switches should be done always with caution and might need some testing by

trial and error. Virtually, in most cases it will be possible to improve the results by specific adaption of the -s settings for a certain data set (compared to the defaults settings). There are no common rules for the stringency settings, their meaning and impact will always depend on the analyzed data. The default settings for the three switches `s1`, `s2`, and `s3` are 0.6.

The switch `-s1` becomes operative during the initial adaptation (weakening) of the constraint. If a certain region of the input alignment is covered by initial structural constraints, then 60 percent of the alignment must be in a condition to fulfill the constraint (in case the default setting of 0.6 is not changed). Note that base pairs are always handled as an entity, and therefore one position always influences the corresponding second position.

The switch `-s2` defines the stringency in a similar way when `RNAsalsa` merges one certain sequence's subset within the set of all possible pairwise alignment foldings of the data to one definite structure model. That model will then serve as an individual constraint for the subsequent thermodynamical folding.

The switch `-s3` finally is operative as a stringency factor also in a similar way during the final calculation of the consensus structure of the complete data set based on the final structure guided alignment.

For all `-s` switches is valid that they can be set to anything between 0 and 1 in 100 steps (2 post decimal positions). If the value is set below (or equals) 0.5 then it might happen that conflicts between equally scored base pairs cannot be solved. In that case `RNAsalsa` posts a warning and the resulting `RNAsalsa` calculations need suspiciousness. Delicate tasks should be repeated with changed `-s` settings.

However, the `-s` values should be in almost all cases higher than 0.5 (special applications might be an exception).

When the parameter `-X` is set, then all pairwise alignments for each sequence are extracted from the input alignment and no internal alignments are done. In some cases this may improve the results, but note that any initial alignment deficiencies cannot be overruled anymore and will lead to systematic errors throughout the complete `RNAsalsa` run.

Sometimes it may be necessary to repeat the final structure guided alignment and the generation of the consensus, e.g. with different `-s3` settings. The switch `-a` allows that by re-reading the folding output of a former run and restarting those final steps.

The switches `-h` and `-v` give a short help and the version information.

9 What else is worth knowing?

The answers to forthcoming *Frequently Asked Questions*.

(1) All alignments during an `RNAsalsa` run are calculated by dynamic programming and use affine gap penalties.
(2) Thermodynamic foldings are minimum free energy driven.
(3) The folding algorithm is taken from the `Vienna RNA package (RNAfold)`, also is the `PostScript` output routine.
(4) Constraints are fulfilled as long as they are thermodynamically possible.
(5) Memory consumption is highest during the optimization and merging steps that lead to the individual constraints.
(6) The internal pairwise folding steps need the most time.
(7) For bug reports, eulogies, and for the purpose of communicating funny experiences please contact: `rs@uni-bonn.de` or `RNAsalsa@gmail.com`

10 The output of RNAsalsa

RNAsalsa produces various (that is to say lots of) output files:

- `SALSA_structaln_sequ.aln`
  is the final structure guided multiple alignment in `clustalW` format.
- `SALSA_structaln_struct.aln`
  is the corresponding alignment file for the same sequences containing the structure (dot-bracket) strings instead of letters.
- `SALSA_structaln_comb_typeA.fas`
  is the corresponding alignment file for the same sequences containing **both** the sequence and the structure (dot-bracket) strings one superimposed on the other.
- `SALSA_structaln_comb_typeB.fas`
  is again the same corresponding alignment file containing both the sequence and the structure strings one superimposed on the other. The difference is an additional ID line above the structure string; this might improve the data compatibility with other software in some cases.
- `SALSA_used_constr.txt`
  is a compilation of all adapted individual constraints that were used to intialize and guide the folding process.
- `SALSA_fold_results.txt`
  is the compilation of all individual thermodynamic foldings in dot-bracket and the calculated minimum free energies.
- `SALSA_consensus.txt`
  contains the consensus structure string for the multiple `RNAsalsa` alignment.
- `SALSA_guide_tree.txt`
  gives an overview about the guide tree that was used internally for the multiple structure based alignment.
- `SALSA_consensus_ss.ps`
  is a PostScript representation of the consensus structure.
- `CONS_*_ss.ps`
  is the file name template for a lot of files containing graphical representations of the used individual constraints.
- `STRUC_*_ss.ps`
  is the file name template for all graphical representations of the individual folding results.
- `SALSA_weakened_constr.txt`
  is the temporary result after the initial constraint has underwent first adaptive steps, e.g. after the switch `-s1` has taken effect. This file can be helpful for troubleshooting, especially to check if the constraint information is lost due to a faulty setup of the `RNAsalsa` run. Such a faulty or unsuitable setup often leads to an empty constraint string, namely all the brackets in

the initial constraint are lost, and only dots (this means **no** constraint) remain for further folding processes.

11 Appendix

alignment parameters			stringency setting defaults	
match score	10		switch -s1	0.6
mismatch score	0		switch -s2	0.6
gap opening penalty	-11		switch -s3	0.6
gap extension penalty	-3			

Table 4
Some parameter values for matches, mismatches and gap penalties in `RNAsalsa` alignments. Default stringency values for secondary structure adoption may be adapted by the user as well for individual constraints as for the final consensus structure: Optional switch $s1$: minimum frequency of base pairing occurrence in the first constraint adaptation. Optional switch $s2$: the stringency setting for the majority voting procedure to obtain an individual constraint by the fusion of pairwise alignment folding results. Optional switch $s3$: stringency settings for the final consensus structure extraction process.

References

[1] Y Tabei, H Kiryu, T Kin, and K Asai. A fast structural multiple alignment method for long RNA sequences. *BMC.Bioinformatics*, 9:33, 2008.

[2] Ivo L Hofacker, Walter Fontana, Peter F Stadler, L Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster. Fast folding and comparison of RNA secondary structures. *Monatsh. Chem.*, 125:167–188, 1994.

[3] Ivo L. Hofacker, Martin Fekete, and Peter F. Stadler. Secondary structure prediction for aligned RNA sequences. *J. Mol. Biol.*, 319:1059–1066, 2002.

[4] A Burk, E J Douzery, and M S Springer. The secondary structure of mammalian mitochondrial 16S rRNA molecules: Refinements based on a comparative phylogenetic approach. *J. Mammalian Evol.*, 9:225–252, 2002.

[5] Walter Fontana, Peter F Stadler, Erich G Bornberg-Bauer, Thomas Griesmacher, Ivo L Hofacker, Manfred Tacker, Pedro Tarazona, Edward D Weinberger, and Peter Schuster. RNA folding landscapes and combinatory landscapes. *Phys. Rev. E*, 47:2083–2099, 1993.

[6] J. Schmitz, M. Ohme, and H. Zischler. The complete mitochondrial sequence of *Tarsius bancanus*: evidence for an extensive nucleotide compositional plasticity of primate mitochondrial DNA. *Mol. Biol. Evol.*, 19:544–553, 2002b.

[7] P De Rijk, J Wuyts, Y Van de Peer, T Winkelmans, and R De Wachter. The European Large Subunit Ribosomal RNA database. *Nucleic Acids Res.*, 28:177–178, 2000.

[8] Y Van de Peer, P De Rijk, J Wuyts, T Winkelmans, and R De Wachter. The european small subunit ribosomal RNA database. *Nucleic Acids Res.*, 28:175–176, 2000.

[9] C Hudelot, V. Gowri-Shankar, H Jow, M Rattray, and Paul G. Higgs. RNA-based phylogenetic methods: application to mammalian mitochondrial RNA sequences. *Mol. Phylogenet. Evol.*, 28:241–252, 2003.

[10] K Katoh, K Kuma, T Miyata, and H Toh. Improvement in the accuracy of multiple sequence alignment program MAFFT. *Genome Inform*, 16:22–23, 2005.

[11] B. Misof and K. Misof. A Monte Carlo approach successfully identifies randomness in multiple sequence alignments: A more objective means of data exclusion. *Syst. Biol.*, 58:21–34, 2009.

[12] Z.H. Yang. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites approximate methods. *J. Mol. Evol.*, 39:306–314, 1994.

[13] Z H Yang, N Goldman, and A Friday. Comparison of models for nucleotide substitution used in maximum-likelihood phylogenetic estimation. *Mol Biol Evol*, 11:316–324, 1994.

[14] D. L. Swofford. *PAUP*: Phylogenetic Analysis Using Parsimony (and Other Methods) Version 4*. Sinauer Associates, Sunderland, MA, 2003. Handbook and Software.

[15] A. Stamatakis. RAxML-VI-HPC: maximum likelihood based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22:2688–2690, 2006.

[16] M. Ott, J. Zola, S. Aluru, and A. Stamatakis. Largescale maximum likelihood based phylogenetic analysis on the IBM BlueGene/L. In *Proc. IEEE/ACM Supercomputing Conf.*, 2007.

[17] F. Ronquist and J. P. Huelsenbeck. `MrBayes 3`: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19:1572–1574, 2003.

[18] J.P. Huelsenbeck and F. Ronquist. MRBAYES: Bayesian inference of phylogenetic trees. *Bioinformatics*, 17:754–755, 2001.

[19] Schoeniger M. and A. von Haeseler. A stochastic model for the evolution of autocorrelated DNA sequences. *Mol. Phylogenet. Evol.*, 3:240–247, 1994.

[20] Sheng Guo, Li-San Wang, and Junhyong Kim. Large-scale simulation of RNA macroevolution by an energy-dependent fitness model. *Systematic Biology*, 2009. accepted; `http://kim.bio.upenn.edu/software/rnasim.shtml`.

[21] P. P. Gardner, A. Wilm, and S. Washietl. A benchmark of multiple sequence alignment programs upon structural RNAs. *Nucleic Acids Res*, 33:2433–2439, 2005.