De-novo identification of homologous RNA secondary structure domains using base-pairing probabilities

Stefan E Seemann[1,2], Martin A Smith[1,3], John S Mattick[1,3]
[1]*Garvan Institute of Medical Research, 384 Victoria Street, Sydney 2010, Australia*
[2]*University of Copenhagen, Groennegaardsvej 3, Frederiksberg, Denmark*
[3]*St Vincents Clinical School, UNSW, Sydney 2010, Australia*
seemann@rth.dk

Keywords: RNA secondary structure, Basepair probability, Structure-based alignment

Abstract: Non-protein coding RNAs (ncRNAs) are the prevalent transcriptional product of higher eukaryote genomes. Their varied biological functions are governed by both their sequence composition and their higher-order structural conformation. The uncertainty of secondary structure prediction algorithms for single RNA sequences in conjunction with the limited diversity of well- characterised RNA structures have restricted the identification and annotation of novel functional ncRNA domains. Here, we present a unified computational methodology for the identification of common RNA secondary structures from a set of sequences, requiring little to no user intervention while being fully customisable. We compare the performance of several state of the art tools for pairwise secondary structure alignment with `DotAligner`, a novel algorithm we developed that considers the ensemble of sub-optimal RNA base pairings between two RNA sequences simultaneously. Through hierarchical clustering and bootstrapping analysis, our method identifies statistically significant clusters of homologous, structured RNA domains with no limitations on the sequence composition of the input. We successfully identify known RNA secondary structures mixed in with randomised controls, as well as novel structured domains from various previously published transcriptomic datasets.

1 INTRODUCTION

The structure of RNA molecules is an essential functional criteria of many non-coding RNAs (ncRNAs), such as the stem-loop of microRNAs and the double stem-loop RNA motifs of the HOTAIR long ncRNA (**?**). NcRNAs can be devided in RNA families of similar inherent functionality, structures, or composition. The largest collection of RNA families is the Rfam database with 2,208 families in its version 11.0 (**?**). However, high-througput sequencing continuously uncovers novel non-coding RNA transcripts and genome-wide RNA structure predictions have revealed hundreds of thousands putative conserved RNA secondary structures. We hypothesize that the RNA secondary structure is the scaffold for intermolecular interactions of many ncRNA-driven regulatory pathways. Protein binding domains of RNA molecules may evolve totally independent from sequence and, instead, may be solely determined by structure. It has been shown that if the sequence similarity falls below 60% sequence comparison will not find anymore domain similarities that are based on structure (**?**). In addition, competing structures and suboptimal structures may support or even drive the functionality of an RNA domain. Hence, methods are needed that find structural similarity independent from sequence conservation and freed from one single optimal RNA secondary structure.

For clustering of RNA domains a dissimilarity measurement of all pairs of query structures is needed. The dissimilarity is described through a pairwise weighted string alignment with arbitrary pairwise dependencies (for base pairings). The Needleman-Wunsch (2) algorithm solves the maximum weight string alignment problem by dynamic programming in $O(N^2)$ by preserving the sequence order and maximizing the similarity. The consideration of pairs of nucleotides in each sequence that form intra-molecular interactions extends the problem to pairwise dependencies among positions in each string. This problem variant is MAX-SNP-hard. However, the problem can be attacked by intelligent heuristics that avoid the examination of all possible aligning states.

Simultaneous alignment and folding (**?**) is the ac-

knowledged gold standard to predict the consensus structure and alignment of a set of related RNA sequences. Because the Sankoff algorithm is practically not applicable, the pre-calculation of the structure ensemble of each sequence, *e.g.* basepair probabilities in thermodynamically equilibrated RNA structure ensembles (**?**), is used by different methods to speed up the calculation of structure-based alignments. The programs `pmcomp` for pairwise and `pmmulti` for multiple alignments (**?**), as well as `LocaRNA` (**?**) score the alignment based on the notion of a common secondary structure. Despite of the usage of the basepair probability matrices extract these methods the maximum-weight common secondary structure but do not explicitely consider suboptimal structures in the alignment. The pairwise alignment of basepair probability matrices (dot plots) has been first introduced by `CARNA` (**?**; **?**). `CARNA` finds iterativelly better alignments with an effective constraint programming technique using a branch and bound scheme (propagator).

Beside of `LocaRNA` and a method based on directed acyclic graph kernels (**?**), the alignment-free approach `ClustGraph` (**?**) has been used to cluster RNA structure in common domains. Here, we propose an alternative heuristic for the pairwise weighted string alignment with arbitrary pairwise dependencies that can deliver dissimilarity scores of dot plots in time close to an Needleman-Wunsch alignment which makes the approach applicable for clustering of large numbers of putative RNA domains.

2 IMPLEMENTATION

The proposed algorithm makes the computationally complex problem of aligning two dotplots available through a two step approach: (1) find dissimilarity (distance) of basepair probabilities of each nucleotide in sequence S_a to each nucleotide in sequence S_b; and (2) find best path through the distance matrix generated in 1. This algorithm runs in $O(N^2)$, hence, in the same time complexity as the sequence-based alignments. In the following we discuss in detail how the algorithm works.

As described in (**?**) the weight Z of alignment A of two arc-annotated sequences (S_a, P_a) and (S_b, P_b) is defined by

$$Z(A) = \sum_{(i,i')\in A} \sigma(i,i') + \sum_{\substack{(i,j)\in Pa,\\ (i',j')\in P_b,\\ (i,i')\in A,(j,j')\in A}} \tau(i,j,i',j') + \gamma \times L, \tag{1}$$

where S is a sequence and P is a base pairing probability matrix, $\sigma(i,i')$ is the similarity of sequence positions $S_a[i]$ and $S_b[i']$, $\tau(i,j,i',j')$ is the similarity of arcs $(i,j) \in P_a$ and $(i',j') \in P_b$, and γ is the gap cost associated with each sequence position that is not matched ($L = |S_a| + |S_b| - 2|A|$). The alignment problem finds the maximal $Z(A)$. As its solution is MAX-SNP-hard we implemented a heuristic of the alignment problem in `DotAligner` which is summarized in the following pseudocode:

Pseudocode

Get Alignment A of the two dotplots P_a and P_b

Require: (S_a, P_a) of length N, (S_b, P_b) of length M
  {STEP 1: global alignment of pairing probabilities of each base in S_a and S_b}
  for $i = 1$ to N **do**
    for $i' = 1$ to M **do**

$$Z(A|i,i') = \kappa \times \sigma(i,i') + \frac{1-\kappa}{min(N,M)} \sum_{\substack{(i,j)\in P_a,\\ (i',j')\in P_b,\\ (j,j')\in A}} \tau(i,j,i',j') + \gamma \times L \tag{2}$$

    end for
  end for
  {STEP 2: local alignment of pairwise weights $Z(A|i,i')$}
  for $i = 1$ to N **do**
    for $j = 1$ to M **do**

$$H_{ij} = max \begin{cases} 0 \\ H_{i-1j} + \gamma \\ H_{i-1j-1} + Z(A|i-1,j-1) \\ H_{ij-1} + \gamma \end{cases} \tag{3}$$

    end for
  end for
  $A(S_a, S_b) = BACKTRACKING(H)$

In step 1 we calculate weights $Z(A|i,i')$ as defined in equation 2 for all combinations of fixed positions i in sequence S_a and i' in sequence S_b. The only difference between equations 2 and 1 is the fixation of i and i', and the introduction of parameter κ setting the impact of sequence conservation. Thus, we globally align two vectors of probabilities instead of two matrices which can be done by the Needleman-Wunsch algorithm. The actual alignments aren't needed in step 2, instead the two sequences are locally aligned by using the weights $Z(A|i,i')$ from step 1 as similarity scores, see equation 3. This can be done by the Smith-Waterman algorithm. The final similarity of the two dotplots is calculated by

$$Z(A) = \frac{1}{|A|}\left(\sum_{\substack{(i,i')\in A,\\ i=\neg gap, i'=\neg gap}} Z(A|i,i') + \gamma \times L\right) \quad (4)$$

where $Z(A|i,i')$ is equal to equation 2 without the term for gaps, $|A|$ is the length of the local alignment and $L = |S_a| + |S_b| - 2|A|$.

The robustness of the alignment is improved by applying log-odds scores of having a specific base pairing against the null model of a random pairing (**?**). Here, we replace $P(i,j)$ with

$$\Psi_{i,j} = max\left(0, log\frac{P(i,j)}{p_0} / log\frac{1}{p_0}\right) \quad (5)$$

where p_0 is the expected probability for a pairing to occur at random. The term $log\frac{1}{p_0(i,j)}$ is a normalization factor that transforms the scores to a maximum of 1. $P == 1$ results in $\Psi = 1$, $P > p_0$ results in $\Psi > 0$, and $P \leq p_0$ results in $\Psi = 0$. This transformation gives weaker similarities if low basepair probabilities are compared, but stronger similarities for high basepair probabilities. The similarity τ is then calculate by

$$\tau(i,j,i',j') = \begin{cases} 0 & \text{if } \Psi(i,j) == 0 \\ & \text{and } \Psi(i',j') == 0 \\ 1 - \delta(i,j,i',j') & \text{else} \end{cases} \quad (6)$$

where $\delta(i,j,i',j') = |\Psi(i,j) - \Psi(i',j')|$ so that $\tau = (0,1)$.

Unpaired probabilities are handled in a similar way by

$$\omega(i) = max\left(0, log\frac{1 - \sum_k P(i,k)}{p_0} / log\frac{1}{p_0}\right) \quad (7)$$

where $p_0(i)$ is the expected probability for an unpaired base to occur at random. Our model is based on structure similarity, however, the sequence similarity σ may be especially important in unpaired regions, *e.g.* as accessible sequence-specific binding motif. Therefore, we weight matching nucleotides by the similarity of their unpaired probabilities:

$$\sigma(i,i') = \begin{cases} 0 & \text{if } \omega(i) == 0 \\ & \text{and } \omega(i') == 0 \\ 1 - \delta(i,i') & \text{else} \end{cases} \quad (8)$$

where $\delta(i,i') = |\omega(i) - \omega(i')|$ so that $\sigma = (0,1)$. By doing so, sequence similarity gets highest weight if the base in both sequences is likely to be unpaired.

Finally, the proposed algorithm can be optimized by different parameters which will be evaluated in the result section:

1. weight of sequence similarity (optimize κ)
2. replace $\gamma \times L$ with affine gap costs $l \times \alpha + k \times \beta$ where l is number of initiation gaps and k is the number of all gaps (optimize α and β)

2.1 Speed up

To achieve a very fast method that can be applied on a large amount of pairwise comparisons, *e.g.* a set of 2,000 RNA sequences requires $2 \times 10^6 - 1,000$ comparisons, we implemented two complementary strategies:

1. set maximal allowed shift of two input sequences in the final alignment
2. random seed alignments

The first strategy reduces the amount of comparisons in step 1 by ignoring pairs of distant nucleotides which will normally never align. This restriction will always find the best global alignment but may miss local alignments of long input sequences if the maximal shift is set to small. The second strategy randomly selects short seed sequences (5 nucleotides) in sequence 1 and aligns them to sequence 2. As soon as the first seed alignment has a gap-free alignment above a given threshold then the entire sequences are aligned around the already calculated similarities. If all seed alignments fail the program is stopped and a similarity of 0 is returned.

3 RESULTS

The accuracy of the proposed algorithm is assessed using the specificity (SP) and the sensitivity (SN), which are defined as follows:

$$SP = \frac{TN}{TN+FP}, \quad SN = \frac{TP}{TP+FN} \quad (9)$$

where TP is the number of correctly predicted positives, FP is the number of incorrectly predicted positives, TN is the number of correctly predicted negatives, and FN is the number of incorrectly predicted negatives. Furthermore, the area under the receiver operating characteristic (ROC) curve was used to optimize κ, α, and β. The ROC curve plots the true positive rates (SN) as a function of the false positive rates (1 - SP) for varying parameters.

As benchmark data set we selected 300 sequences of 10 H/ACA-box snoRNA families from Rfam version 11.0 seed alignments with average pairwise sequence identity (APSI) $< 90\%$ and sequence lengths of $>$ 130bp and $<$ 140bp: *SNORA1*, *SNORA13*, *SNORA14*, *SNORA15*, *SNORA16*, *SNORA17*,

SNORA18, *SNORA19*, *SNORA2*, *SNORA22*. We chose only sequences of similar length because step 1 of DotAligner performs global alignments.

Martins benchmark for different APSIs. Compare with reference alignments by the SPS measure introduced in Bralibase 2.1 (see CARNA paper). Compare Rfam families with significant clusters generated by pvclust.

3.1 Parameter optimization

We tried gap costs γ of 3, 4, 5 and 6, which are used unweighted in step 1 of the algorithm (equation 2) and weighted in step 2 of DotAligner (equation 3) by the factor

$$\frac{1}{N \times M} \sum_{i=1}^{N} \sum_{i'=1}^{M} Z(A|i,i') \qquad (10)$$

where N and M are lengths of S_a and S_b, respectively. The ROC curve in Figure 1 shows the lowest γ as most sensitive (SN = 0.61) and the highest γ as most specific (SP = 1.0) for correctly clustering the selected Rfam families. In the following we choose $\gamma = 4$, whereas the optimal gap cost lies somewhere between 3 and 4.

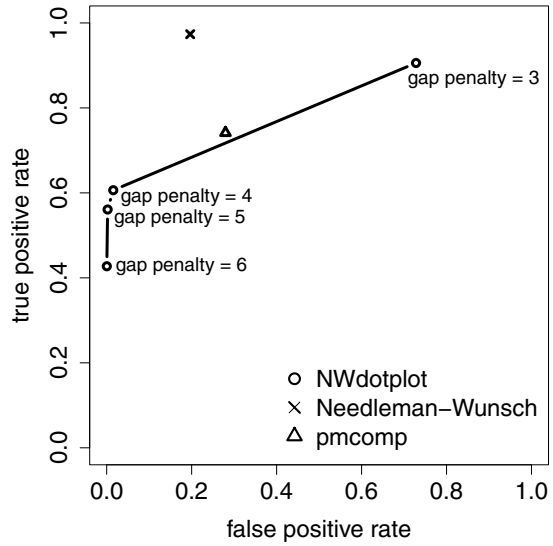

Figure 1: Performance comparison of hierarchical cluster analysis: Degree of agreement between the 10 tested Rfam families and the automated clustering based on distance scores from DotAligner with different gap penalties, Needleman-Wunsch algorithm and pmcomp.

3.2 Comparison with other methods

We compare DotAligner with sequence alignments (in-house implementation of Needleman-Wunsch algorithm with the blastn parameters match = 2, mismatch = -3 and gap penalty = 5 which are optimized for sequence identity of 90%) and the structure alignment tools pmcomp (using default parameters or larger values for parameter *-D* if the length difference of two sequences is $>$ 5 bp), CARNA, and LocaRNA. Figure 1 shows that the sequence aligner (SP = 0.80, SN =0.97) performs very well on our benchmark set with a very high sensitivity which is most likely due to the fact that the input sequences have some degree of sequence information. pmcomp (SP = 0.72, SN = 0.74) performed with a medium sensitivity and specificity. With DotAligner we are able to find very well defined clusters (SP = 0.99), however, at the cost of sensitivity (SN = 0.61), see Figure 2.

3.3 Clustering methodology

The reliability of our pairwise structure alignment algorithm at clustering homologous RNA structures was tested on a curated database of RNA structure families (cf RFAM). This enables both qualitative and quantitative performance evaluation using a gold-standard reference. We compared DotAligner to other RNA structure alignment and clustering tools using the following framework:

1. Generate dissimilarity matrix dM_A from $\frac{n(n-1)}{2}$ pairwise structure comparisons with each algorithm
2. Hierarchical clustering of RNA secondary structures and significance testing with pvclust (Suzuki R and Shimodaira H. Bioinformatics 2006).
3. Generate dissimilarity matrix dM_R from scoring metric of (1.) from curated RFAM alignments (constrained alignment).
4. Calculate the correlation coefficient between dM_A and dM_R using the Mantel correlation statistic (the cross-product between the standardised distances).

Benchmarking was performed on both complete RNA structures (global alignment) and randomly selected subsequences (local alignment) for various RFAM families, as described below.

3.4 Benchmark data generation

xx RNA families were manually selected from the seed alignments of RFAM 11 (REF). *How should we limit the mean pairwise identity? All structures must be within a given range and perform several*

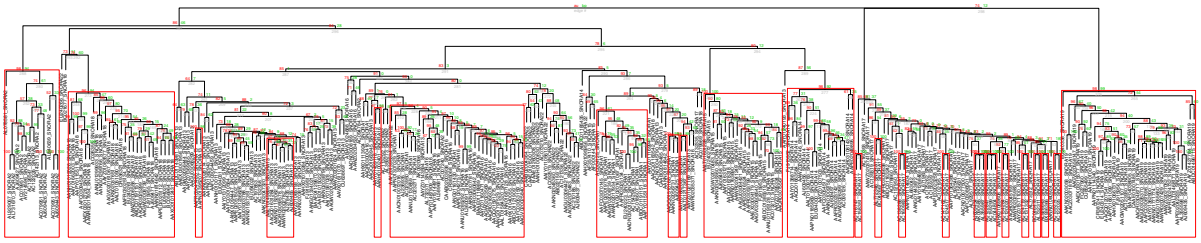

Figure 2: Automated hierarchical clustering of 300 sequences from 10 H/ACA snoRNA families. The dissimilarity matrix was calculated through `DotAligner` with gap penalty 4. The clustering was conducted by the R-package `pvclust` with multiscale bootstrap resampling with number of bootstrap 1000. We define clusters (red rectangles) as Approximately Unbiased (AU) p-values > 0.95 rejecting the hypothesis that "the cluster does not exist" with significance level 0.05.

independent comparisons, i.e. one per SeqID range? Then compare the individual SeqID ranges to a sample of variable SeqIDs (without selection)?

We employed the BuildRfamBenchmark JAVA program from (Smith M et al. NAR 2013) to generate the sample alignments for the RFAM entries listed in TableXX. The tRNA sample includes special tRNAs, like ser-tRNA with a 5th hairpin to see how the latter gets clustered by the algorithms.

RFAM ID	RNA class	average length
	5s rRNA	
	SRP	
	tRNA	
	HaCa snoRNA	
	pre-miRNA	

3.5 Complete RFAM sequences

Global alignment. More emphasis on quantitative clustering, accuracy, and correlation with control.

	SeqId 10 . . . 55		
	SP	SN	Time [s]
`DotAligner`	84.1	64.8	7.2
`CARNA`	?	?	?
`LocaRNA`	96.9	54.0	?
`FOLDALIGN`	88.6	73.7	34.8
`pmcomp`	97.9	35.7	289.9
`Needleman-Wunsch`	92.6	54.6	0.002
	SeqId 56 . . . 95		
	SP	SN	Time [s]
`DotAligner`	100	86.4	7.1
`CARNA`	?	?	?
`LocaRNA`	?	?	?
`FOLDALIGN`	97.2	79.7	37.5
`pmcomp`	100	64.5	338.4
`Needleman-Wunsch`	91.8	90.2	0.002

3.6 Fragmented RFAM sequences

Local alignment, simulating genomic screens. More emphasis on qualitative clustering

4 DISCUSSION

The proposed method does not intent to find the optimal alignment of basepair probability matrices because in step 1 pairing probabilities are aligned for each pair of nucleotides and these alignments may differ between nucleotide pairs. However, the application of `DotAligner` is a fastly calculated similarity score between two probability matrices to enable their subsequent clustering. Compared to `CARNA`, which also does not garanty the optimal solution, `DotAligner` is deterministic. Previously we repeated step 1 and 2 of `DotAligner` until no new gaps were included in the alignment, however, we neglected this strategy to improve speed because the alignment changes were marginal.

We plan to integrate the proposed method in a pipeline that screens regions of interest for structured RNA domains in a collection of RNA molecules. The so far presented approach finds only semi-local alignments, meaning the heuristic in the first step of the algorithm gives global alignments, whereas the second step provides a final local alignment. This strategy is applicable for input sequences of similar lengths, however, a local alignment is favorable if input sequences are very long or have different lengths. Hence, a possible screening pipeline may comprise window based thermodynamic folding, *e.g.* by `RNAplfold` (**?**), the identification of regions of high intra-molecular binding probabilities, *e.g.* `RNAlocal` (**?**), followed by the presented alignment tool `DotAligner`. The pre-selection of local structural potential is necessary because `DotAligner` finds only semi-local alignments but local alignments may improve the boundaries of common structured

RNA domain.

DotAligner can also be extended for multiple alignments, similar to the strategy implemented in pmmulti (**?**), and the generation of phylogenetic trees. This may replace or support the hierarchical clustering approach used here. In addition, both may serve as input for RNA secondary structure predictors, such as PETfold (**?**) unifying thermodynamic and evolutionary information.

ACKNOWLEDGEMENTS

I thank the Carlsberg foundation for my travel grant. Skål!
MAS is funded in part by a Cancer Council NSW project grant and