BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 16 2005, pages 3352–3359
doi:10.1093/bioinformatics/bti550

Structural bioinformatics

MARNA: multiple alignment and consensus structure prediction of RNAs based on sequence structure comparisons

Sven Siebert and Rolf Backofen*

Department of Bioinformatics, Institute of Computer Science, Friedrich-Schiller-University Jena, Ernst-Abbe Platz 2, 07743 Jena, Germany

Received on April 1, 2005; revised and accepted on June 16, 2005
Advance Access publication June 21, 2005

ABSTRACT

Motivation: Due to the importance of considering secondary structures in aligning functional RNAs, several pairwise sequence–structure alignment methods have been developed. They use extended alignment scores that evaluate secondary structure information in addition to sequence information. However, two problems for the multiple alignment step remain. First, how to combine pairwise sequence–structure alignments into a multiple alignment and second, how to generate secondary structure information for sequences whose explicit structural information is missing.

Results: We describe a novel approach for multiple alignment of RNAs (MARNA) taking into consideration both the primary and the secondary structures. It is based on pairwise sequence–structure comparisons of RNAs. From these sequence–structure alignments, libraries of weighted alignment edges are generated. The weights reflect the sequential and structural conservation. For sequences whose secondary structures are missing, the libraries are generated by sampling low energy conformations. The libraries are then processed by the T-Coffee system, which is a consistency based multiple alignment method. Furthermore, we are able to extract a consensus-sequence and -structure from a multiple alignment. We have successfully tested MARNA on several datasets taken from the Rfam database.

Availability: MARNA can be used online on our webpage www.bio.inf.uni-jena.de/Software/MARNA/index.html

Contact: backofen@inf.uni-jena.de

INTRODUCTION

In recent years, RNA molecules gained increasing interest since a huge variety of functions associated with them was found. Consequently, research on small RNAs has been elected as the scientific breakthrough of the year 2002 by the readers of the Science magazine (Couzin, 2002). The function of an RNA-molecule is mainly determined by its (secondary) structure. It is assumed that the structure of an RNA is often more conserved than its sequence (even more than for proteins). Hence, one cannot use standard multiple sequence alignment techniques like e.g. Clustal W (Thompson *et al*., 1994), Dialign (Morgenstern, 1998) or T. Coffee (Notredame *et al*., 2000) since they completely neglect structural information.

*To whom correspondence should be addressed.

Multiple sequence- and structure-based alignments of RNAs can be divided into two major classes, the probabilistic and the non-probabilistic approaches. Probabilistic approaches are based on stochastic context-free grammars (SCFG) and require an initial multiple alignment as input. The quality of the outputs crucially depends on this initial alignment. They are used to model RNA-families and/or to predict a secondary structure via comparative analysis [e.g. Cove (Eddy and Durbin, 1994), RNACAD (Brown, 1999) and Pfold (Knudsen and Hein, 2003)]. A non-probabilistic, comparative approach is e.g. given by RNAlign (Corpet and Michot, 1994) that performs an alignment between a bank of aligned sequences and a new sequence.

In this paper, we propose a non-probabilistic approach to align a set of more than two RNAs with or without known conformations. The standard approach is to perform direct pairwise alignments of RNAs using sequence and (secondary) structure information and to combine the pairwise alignments into a multiple alignment. No general approach yet exists albeit there is a wealth of approaches for pairwise alignment of RNAs (see below). The reason is that the results of the pairwise sequence/structure alignments cannot simply be aligned in a progressive way (like profiles for sequence alignments). To the best of our knowledge, there are only two exceptions, namely PMcomp/PMmulti (Hofacker *et al*., 2004) and RNAforester (Höchsmann *et al*., 2003). PMcomp aligns RNA base pairing probability matrices and predicts a common folding structure between two sequences. PMmulti uses PMcomp in a progressive alignment strategy and provides multiple alignments with good qualities. However, it has a high complexity of $O(n^6)$ time and $O(n^4)$ space for the pairwise comparisons. In RNAforester, secondary structures are interpreted as trees, and a tree-based alignment is applied.

We solved the problem of combining pairwise alignments of RNAs as follows. First, alignment edges between RNAs reflecting sequence and structure similarities are generated based on an algorithm published by Jiang *et al*. (2002). In a second step, these edges are collected in a library, which is given as input to the multiple sequence alignment method T-Coffee (Notredame *et al*., 2000). Structural positions that are supported by several pairwise comparisons are strengthened. Hence, the result comprises sequence and structure similarities of RNAs albeit the progressive alignment strategy is in principle not structure-based.

We have used the algorithm of Jiang *et al*. (2002) since it provides the greatest scoring flexibility and has moderate complexity. But any other sequence- and structure-based pairwise alignment method can also be adapted to our approach. The computational problem of

pairwise alignment of RNAs was first addressed by Sankoff (1985), who proposed a dynamic programming algorithm that aligns a set of RNA sequences while predicting their common fold at the same time. Subsequently, a variety of pairwise sequence-structure alignment approaches have been developed. Lenhof *et al*. (1998) addresses the problem of optimally aligning a given RNA sequence of unknown structure to one of known sequence and structure. Local pairwise RNA-alignments using the same scoring scheme as Jiang *et al*. (2002) are considered by Backofen and Will (2004). Beside the above listed approaches, there are several approaches that work on a tree based representation of RNAs (see e.g. Jiang *et al*., 1995; Höchsmann *et al*., 2003; Shapiro and Zhang, 1990).

We tested our approach on eukaryotic SECIS-elements on tRNA-like 3′ UTR elements from *Tymovirus/Pomovirus* and on the *Hammerhead ribozyme* (type III). We compared our MARNA results with the manual alignments taken from the Rfam database and with the alignments generated by PMmulti.

METHODS

A sequence S is a word over the alphabet $\{A, C, G, U\}$. $S[i]$ denotes the i-th symbol in S. An arc a is a pair $(i, j) \in \mathbb{N} \times \mathbb{N}$ such that $i < j$. i and j are called ends of the arc a. A base is called free if it is not involved in any arc. A secondary structure P is a set of arcs such that no end of an arc appears more than once in P. Here, we consider secondary structures that are nested, i.e. for any two base pairs $(i_1, i_2) \in P$ and $(j_1, j_2) \in P$, we have either independent base pairs with $i_2 < j_1$ or $j_2 < i_1$, or nested base-pairs with $i_1 < j_1 < j_2 < i_2$ or $j_1 < i_1 < i_2 < j_2$. We call the tuple $S = (S, P)$ a sequence-structure. In the following, all RNAs are specified by their sequences and their known secondary structures. Unknown structures will be handled in a later section.

We use the gap symbol '–' to denote an inserted/deleted nucleotide. An alignment $\mathcal{A}$ of two sequence–structures $S_1 = (S_1, P_1)$ and $S_2 = (S_2, P_2)$ is a subset of $[1..|S_1|] \cup \{-\} \times [1..|S_2|] \cup \{-\}$, where for all pairs $(i, j), (i', j') \in \mathcal{A}$ holds

(1) $i \leq i' \Rightarrow j \leq j'$

(2) $i = i' \neq - \Rightarrow j = j'$ and

(3) $j = j' \neq - \Rightarrow i = i'$.

In addition, we require that for every $i \in [1..|S_1|]$ there is some j with $(i, j) \in \mathcal{A}$, and vice versa for $j \in [1..|S_2|]$. The pairs $(i, j) \in \mathcal{A}$ are called alignment edges. We say that $i \in [1..|S_1|]$ is *aligned* with j if $(i, j) \in \mathcal{A}$, and analogously for $j \in [1..|S_2|]$. An alignment edge $(i, j) \in \mathcal{A}$ is called realized if neither $i = -$ nor $j = -$.

Pairwise alignment scores

The scoring of an alignment $\mathcal{A}$ of two sequence-structures $S_1 = (S_1, P_1)$ and $S_2 = (S_2, P_2)$ is based on the notion of edit operations on bases as well as on arcs. We briefly recall the edit operations from Jiang *et al*. (2002), and present a slightly modified version of their distance-based scoring scheme.

Edit operations on free bases are base match, base mismatch and base deletion. A base match has cost 0, base mismatch and base deletion have positive costs. We combine these cost functions into a single cost function $w_{base}(i, j)$, where $w_{base}(i, j) = 0$ only if $S_1[i] = S_2[j]$. We will feel free to write either the positions or the nucleotides as arguments in the cost function whereever necessary.

For arcs, we have a more complex scoring scheme. Consider an arc $(i, i') \in P_1$ such that i is aligned with j and i' is aligned with j'. An arc match occurs if j, j' form an arc $(j, j') \in P_2$, $S_1[i_1] = S_2[j_1]$ and $S_1[i_2] = S_2[j_2]$. We have an arc mismatch if $(j, j') \in P_2$, but $S_1[i_1] \neq S_2[j_1]$ or $S_1[i_2] \neq S_2[j_2]$. Arc matches have cost 0, whereas arc mismatches have cost $w_{am}(i, i', j, j')$. On the other hand, if $(j, j') \notin P_2$, then we have an arc deletion with cost $w_{ad}(i, i', j, j')$. Lin *et al*., subdivided arc deletions into arc breakings, arc alterings and arc removings. An arc breaking occurs if none of j and j' equals the gap symbol –. If exactly one of j, j' equals –, then we have an arc altering. If both j, j' are equal to –, then we have an arc removing. The edit operations on arcs are summarized in Figure 1.

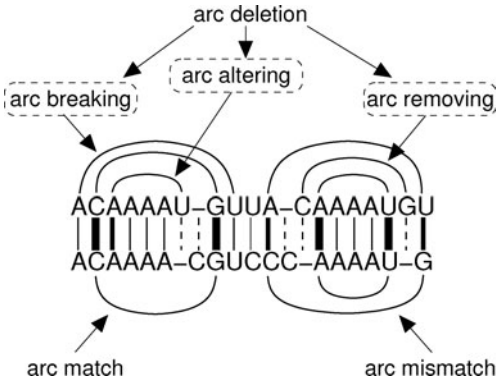


Fig. 1. An alignment of two RNAs with corresponding edit operations on arcs. Alignment edges between bases are shown as solid lines (realized edges) and dashed lines (non-realized edges). The thickness of realized edges corresponds to similarity weights between bases. Non-realized alignment edges are skipped for the multiple alignment step.

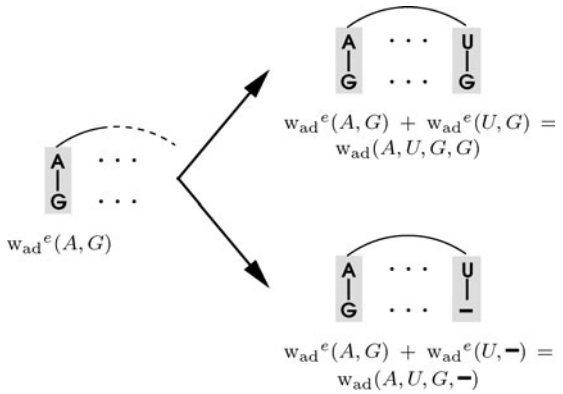


Fig. 2. Independent scoring of both arc ends.

The total score of an alignment is the sum of costs of all applied edit operations that transform one sequence–structure into the other. The complexity of finding an alignment with minimal costs is determined by the way arc deletions are scored. Jiang *et al*. (2002) presented a dynamic programming algorithm that solves this problem in $O(n^2m^2)$ time and $O(nm)$ space under certain restrictions on the scoring of arc deletions. In effect, this requires the existence of functions $w_{ad}^l(i, j)$ and $w_{ad}^r(i', j')$ for the left and right ends, respectively, such that

$$w_{ad}(i, i', j, j') = w_{ad}^l(i, j) + w_{ad}^r(i', j').$$

In the following, we will not distinguish between left and right ends of an arc, i.e. we set $\forall i, j : w_{ad}^l(i, j) = w_{ad}^r(i, j) = w_{ad}^e(i, j)$, where $w_{ad}{}^e(i, j)$ is a single function to score both ends of an arc.

The effect of this restriction is that one can evaluate both arc ends in an alignment independently, which is a necessary prerequisite for the dynamic programming algorithm. This situation is depicted in Figure 2.

In our approach, we even simplify the scoring scheme further by defining $w_{ad}{}^e(i, j)$ to be composed of a base match, base mismatch or base deletion together with a fixed cost for deleting an arc. Hence, we set

$$w_{ad}{}^e(i, j) = w_{base}(i, j) + \frac{1}{2} w_{ad}^{const},$$

where w_{ad}^{const} is the cost for deleting one arc.

Multiple alignment

T-Coffee Once the sequence–structure alignments have been calculated for all pairs of input sequences, we construct the so-called library. A library for a pairwise alignment of two sequence–structures consists of the set of all realized edges together with a weighting of each edge. Then, the libraries for all pairwise alignments are given to T-Coffee (Notredame *et al*., 2000) to build a single multiple alignment.

The T-Coffee system is a consistency based alignment method that combines local and global information to produce a multiple alignment in the following manner. First, an extended primary library is produced that improves all pairwise alignments by taking into consideration how all other sequences align with the current two RNAs. Edges achieve higher weights if the bases at the end points of these edges are also aligned with other sequences. Second, the improved dataset of pairwise alignments is processed by a progressive alignment strategy. A distance matrix is computed between all sequences using the improved weights of alignment edges. Subsequently, the neighbor-joining method (Saitou and Nei, 1987) provides a phylogenetic tree, which dictates the order of aligning these sequences. Since the initial libraries were generated from sequence–structure alignments, the resulting multiple alignment reflects the sequential *and* structural similarities of RNAs.

Distance and similarity The weights attached to realized edges in the libraries correspond to similarity weights. For that reason, we have to transform the distances defined in the previous section into similarity values. Smith and Waterman (1981) solved the problem of transforming distances into similarities for edit operations on bases. We extend this approach to our set of edit operations. The main observation of Smith and Waterman (1981) is that one has to consider the number of nucleotides r involved in an edit operation. We call this number the order of the edit operation. In our case, we have edit operations with $r = 4$ (arc match and arc mismatch), $r = 2$ (base match and base mismatch) and $r = 1$ (base deletion). Since we have split the arc deletion operation into two separate edit operations for the arc ends, we have an edit operation with $r = 2$ if the arc end is aligned with a nucleotide, and an edit operation with $r = 1$ if the arc end is aligned with **–**.

By enumerating all different edit operations, we can write the distance score of an alignment $\mathcal{A}$ as

$$\mathrm{dist}(\mathcal{A}) = \sum_r \sum_k w^{r,k} \lambda_{\mathcal{A}}^{r,k},$$

where $w^{r,k}$ is the cost for the k-th edit operation of order r (for $r = 4, 2, 1$), and $\lambda_{\mathcal{A}}^{r,k}$ is the number of times the k-th edit operation of order r is used in the alignment $\mathcal{A}$. Then we can rewrite the distances $w^{r,k}$ into similarities $s^{r,k}$ as follows:

THEOREM 1. *Consider a scoring scheme where $w^{r,k}$ is the cost of the k-th edit operation of order r. Let A^{MSP} be any fixed value, which is interpreted as the maximal similarity per nucleotide position we want to achieve. Define the similarity $s^{r,k}$ for the k-th edit operation of order r by*

$$s^{r,k} = r \cdot A^{\mathrm{MSP}} - w^{r,k}.$$

Then the alignment $\mathcal{A}$ which minimizes dist($\mathcal{A}$) is the alignment that maximizes sim($\mathcal{A}$) $= \sum_r \sum_k s^{r,k} \lambda_{\mathcal{A}}^{r,k}$, and vice versa.

PROOF 1. The optimal alignment for two sequence–structures $S_1 = (S_1, P_1)$ and $S_2 = (S_2, P_2)$ under the similarity score is given by

$$\begin{aligned}
\mathcal{A}_{opt} &= \operatorname*{argmax}_{\mathcal{A}\ \text{align. of } S_1,S_2} \left\{ \sum_{r,k} s^{r,k} \lambda_{\mathcal{A}}^{r,k} \right\} \\
&= \operatorname*{argmax}_{\mathcal{A}} \left\{ \sum_{r,k} (r A^{\mathrm{MSP}} - w^{r,k}) \lambda_{\mathcal{A}}^{r,k} \right\} \\
&= \operatorname*{argmax}_{\mathcal{A}} \left\{ A^{\mathrm{MSP}} \sum_{r,k} r \lambda_{\mathcal{A}}^{r,k} - \sum_{r,k} w^{r,k} \lambda_{\mathcal{A}}^{r,k} \right\}
\end{aligned}$$

Since any nucleotide position is involved in exactly one edit operation, we know that $\sum_{r,k} r \lambda_{\mathcal{A}}^{r,k}$ is the total number of nucleotide position involved in edit operations. Hence, $\sum_{r,k} r \lambda_{\mathcal{A}}^{r,k} = |S_1| + |S_2|$. Thus,

$$\begin{aligned}
\mathcal{A}_{\mathrm{opt}} &= \operatorname*{argmax}_{\mathcal{A}} \left\{ A^{\mathrm{MSP}} (|S_1| + |S_2|) - \sum_{r,k} w^{r,k} \lambda_{\mathcal{A}}^{r,k} \right\} \\
&= \operatorname*{argmax}_{\mathcal{A}} \left\{ - \sum_{r,k} w^{r,k} \lambda_{\mathcal{A}}^{r,k} \right\} = \operatorname*{argmin}_{\mathcal{A}} \left\{ \sum_{r,k} w^{r,k} \lambda_{\mathcal{A}}^{r,k} \right\}.
\end{aligned}$$

Thus, one has only to choose the maximal similarity per position A^{MSP} to transform the distance score into a similarity score without changing the global optimal alignment. Albeit it does not change the global optimal alignment, it is important for the T-Coffee system since only the realized edges are considered when combining the pairwise alignments into a multiple alignment. This implies that alignment edges containing a gap have a weight of 0. To achieve a good approximation to this, we set

$$A^{\mathrm{MSP}} = \max_{r,k} \left\{ \frac{w^{r,k}}{r} \right\}.$$

The above theorem can also be extended to vary the contribution from structural and sequential positions for the generation of the multiple alignment. Obviously, the distance score is flexible enough to strengthen either structural or sequential positions. Structural positions are strengthened by rising the constant cost for arc deletion (i.e. $\mathrm{w}_{\mathrm{ad}}^{\mathrm{const}}$). But this is somewhat lost if we have the same maximal similarity for structural and sequential positions. This leads to the following modification of the theorem. We say that that a position i in the sequence–structure $S = (S, P)$ is a structural position if there is an i' with $(i, i') \in P$ or $(i', i) \in P$. The position i is defined to be sequential otherwise. The order of an edit operation is now defined by two values r_{str} and r_{seq}, which are the numbers of structural and sequential positions in the edit operation, respectively. For an alignment $\mathcal{A}$ the value $\lambda_{\mathcal{A}}^{r_{\mathrm{str}},r_{\mathrm{seq}},k}$ denotes again the number of times the k-th edit operation of order $r_{\mathrm{str}}, r_{\mathrm{seq}}$ is used in $\mathcal{A}$. Then we can write the distance score of $\mathcal{A}$ as $\mathrm{dist}(\mathcal{A}) = \sum_{r_{\mathrm{str}},r_{\mathrm{seq}},k} w^{r_{\mathrm{str}},r_{\mathrm{seq}},k} \lambda_{\mathcal{A}}^{r_{\mathrm{str}},r_{\mathrm{seq}},k}$.

THEOREM 2. *Let $w^{r_{\mathrm{str}},r_{\mathrm{seq}},k}$ be the cost of the k-th edit operation of order $r_{\mathrm{str}}, r_{\mathrm{seq}}$. Let $A_{\mathrm{str}}^{\mathrm{MSP}}$ be the maximal similarity for structural positions, and let $A_{\mathrm{seq}}^{\mathrm{MSP}}$ be analogously defined for sequential positions. Define the similarity for the k-th edit operation of order $r_{\mathrm{str}}, r_{\mathrm{seq}}$ by*

$$s^{r_{\mathrm{str}},r_{\mathrm{seq}},k} = r_{\mathrm{str}} \cdot A_{\mathrm{str}}^{\mathrm{MSP}} + r_{\mathrm{seq}} \cdot A_{\mathrm{seq}}^{\mathrm{MSP}} - w^{r_{\mathrm{str}},r_{\mathrm{seq}},k}.$$

Then the alignment $\mathcal{A}$ which minimizes dist($\mathcal{A}$) is the alignment that maximizes the similarity sim($\mathcal{A}$) $= \sum_{r_{\mathrm{str}},r_{\mathrm{seq}},k} s_{\mathcal{A}}^{r_{\mathrm{str}},r_{\mathrm{seq}},k} \lambda_{\mathcal{A}}^{r_{\mathrm{str}},r_{\mathrm{seq}},k}$, and vice versa.

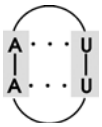
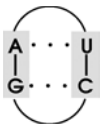
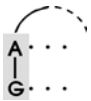
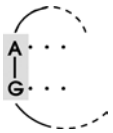
The resulting scoring scheme is depicted in Table 1. As discussed above for A^{MSP}, a good choice for $A_{\mathrm{str}}^{\mathrm{MSP}}$ (resp. $A_{\mathrm{seq}}^{\mathrm{MSP}}$) is to use the maximal cost for edit operations involving only structural (resp. sequential) positions. Another possibility is to choose $A_{\mathrm{str}}^{\mathrm{MSP}}$ such that the maximal weight for a single edge (namely $2A_{\mathrm{str}}^{\mathrm{MSP}}$) equals the maximal value allowed in T-Coffee. This is a reasonable choice if there is a high confidence in the structures selected for the sequences, and one wants to ensure that the structural positions are aligned. Note that in the current implementation of MARNA, we use the same values for $A_{\mathrm{seq}}^{\mathrm{MSP}}$ and $A_{\mathrm{str}}^{\mathrm{MSP}}$.

Combining several structures

The previously described approach uses one given structure for each sequence, which could be for example an experimentally confirmed structure. Usually, the structure is not known and has to be computed by secondary structure prediction programs like Mfold (Zuker, 1994) or RNAfold (Hofacker, 2003). Here, we are confronted with the problem that very often the real motif is not found in the minimum free energy structure, but in some sub-optimal structures.

A better strategy is to assign several structures to each sequence covering different possible folds of the sequence. To generate an ensemble of

Table 1. Edit operations on arcs together with the associated distances and their similarity values given to the T-Coffee system

Edit-Op	Name	Distance	Similarity
	Arc match	0	$4 \cdot A_{\text{str}}^{\text{MSP}}$
	Arc mismatch	$\text{w}_{\text{am}}(\text{A, U, G, C})$	$4 \cdot A_{\text{str}}^{\text{MSP}} - \text{w}_{\text{am}}(\text{A, U, G, C})$
	Arc breaking Arc altering (realized edge)	$\text{w}_{\text{base}}(\text{A, G}) + \frac{1}{2}\,\text{w}_{\text{ad}}^{\text{const}}$	$A_{\text{str}}^{\text{MSP}} + A_{\text{seq}}^{\text{MSP}} - \text{w}_{\text{base}}(\text{A, G}) - \frac{1}{2}\,\text{w}_{\text{ad}}^{\text{const}}$
	Arc breaking Arc altering (realized edge, two arcs)	$\text{w}_{\text{base}}(\text{A, G}) + \text{w}_{\text{ad}}^{\text{const}}$	$2 \cdot A^{\text{MSP}} - \text{w}_{\text{base}}(\text{A, G}) - \text{w}_{\text{ad}}^{\text{const}}$
	Arc breaking Arc removing (non-realized edge)	$\text{w}_{\text{base}}(\text{A, }-) + \frac{1}{2}\,\text{w}_{\text{ad}}^{\text{const}}$	no realized edge

Note that for arc-match and arc-mismatch, we assign half of the total similarity value to each alignment edge when building the library.

low energy structures, we have used the stochastic backtracking version of RNAsubopt (Vienna RNA package) as well as RNAshapes (Giegerich *et al*., 2004). The latter avoids the production of a large number of similar structures. The result is a usually small set of different structures $\mathcal{E}_S = \{P_S^1 \dots P_S^{n_S}\}$ for a sequence S. In the following, we call $\mathcal{E}_S$ the ensemble of the sequence S. Since the structures P_S^k in $\mathcal{E}_S$ occur with different frequencies in the low energy spectrum, they have to be weighted. The weight for each structure is given by the conditional probability $\Pr(P_S^k|\mathcal{E}_S)$ of seeing this structure under the condition that only structures of the ensemble $\mathcal{E}_S$ are considered. Thus, we have

$$\Pr(P_S^k|\mathcal{E}_S) = \frac{\Pr^b[P_S^k]}{\sum_{1\le l\le n} \Pr^b[P_S^l]}, \tag{1}$$

where $\Pr^b[P_S^l]$ is the Boltzmann probability that S forms the structure P_S^l. Since RNAshapes often returns structures with similar energies, we approximate $\Pr(P_S^k|\mathcal{E}_S)$ by the uniform distribution in our current implementation of MARNA, thus avoiding the calculation of the Boltzmann probabilities.

Next, we have to use the different structures to form a single library for a pair of sequences. So let S_1 and S_2 be two sequences. Assume that we have selected n_1 structures for the first sequence and n_2 structures for the second one. In this setting, $n_1 = 1$ (resp. $n_2 = 1$) means that we have a unique known structure for S_1 (resp. S_2). Thus, we are able to mix sequences having known structures with sequences where we do not know the structures. Let $\mathcal{E}_{S_1} = \{P_{S_1}^1 \dots P_{S_1}^{n_1}\}$ and $\mathcal{E}_{S_2} = \{P_{S_2}^1 \dots P_{S_2}^{n_2}\}$ be the ensembles of structures selected for sequences S_1 and S_2, respectively. Then we perform $n_1 \times n_2$ sequence–structure alignments for $(S_1, P_{S_1}^k)$ and $(S_2, P_{S_2}^l)$ $(1 \le k \le n_1,\ 1 \le l \le n_2)$. All realized edges from these alignments are then collected into a single library. For edges that are common to several alignments, the weights are summed up. In order to achieve weights that are consistent with other libraries, the combined similarity values of the realized edges are normalized by multiplying them by $\Pr(P_{S_1}^k \mid \mathcal{E}_{S_1}) \cdot \Pr(P_{S_2}^l \mid \mathcal{E}_{S_2})$.

Consensus structure

Once we have computed the final alignment, we are ready to calculate a consensus structure from this alignment. The standard approach is to estimate possibly conserved bonds by means of the mutual information content [e.g. Luck *et al*. (1999), Gutell and Woese (1990), Chiu and Kolodziejczak (1991) and Gutell *et al*. (1992)] between all columns i and j in a given alignment. The keynote is that if there is not very much sequence conservation in these columns, but the columns show a high correlation measured by the mutual information content, then this must be due to a conserved bond. Hofacker *et al*. (2002) extended this approach by considering the probabilities of forming these base-pairs.

In our case, the situation is different since we explicitly use structure information for the calculation of the alignment. Hence, the calculation of the consensus structure should be based on these given structures.

To exemplify the basic idea, suppose that exactly one structure per sequence is given. Thus, each structure must then be interpreted as the 'real' known structure. A conserved base pair between two columns in the alignment is found if the majority of sequences have a base pair at the corresponding sequence positions. The remaining problem is that the resulting set of conserved base pairs alone does not form a nested secondary structure and is thus not a valid consensus structure. This is a problem common to all approaches for calculating a consensus structure. The usual solution is to calculate a secondary structure that maximizes base pair conservation. So let c, c' be two columns with $1 \le c < c' \le m$, where m is the number of columns of the multiple alignment. Furthermore, let $\text{bp\_cons}(c, c')$ be the number of sequences that have a base pair between the corresponding sequence positions. The consensus structure is then defined to be a secondary structure $P \subseteq [1..m] \times [1..m]$ such that

$$\sum_{(c,c')\in P} \text{bp\_cons}(c, c')$$

is maximized. This can be calculated using a variant of the Nussinov algorithm (Nussinov *et al*., 1978). For this purpose, we define a matrix $(N_{i,j})$ with $1 \le i, j \le m$, where

$$N_{i,j} = \max_P \sum_{\substack{(c,c')\in P \\ i\le c<c'\le j}} \text{bp\_cons}(c, c')$$

is the maximal base pair conservation for all columns between i and j. The corresponding recursion equation for this matrix is

$$N_{i,j} = \max \left\{ \begin{array}{c} N_{i+1,j}, \\ N_{i,j-1}, \\ N_{i+1,j-1} + \text{bp\_cons}(i,j), \\ \max_{i<k<j} \left\{ N_{i,i+k} + N_{i+k+1,j} \right\} \end{array} \right\}.$$

It is a dynamic programming approach, where the traceback reports the consensus structure of the alignment.

Finally, we have again to consider the case where we are given structure ensembles for some (or all) sequences. The overall structure of the approach is the same, only the definition of conserved base pairs has to be adapted, i.e. the definition of bp_cons(c, c'). If we have several structures for one sequence, then the probability of seeing a particular base pair depends on the probabilities of the structures that contain this base pair. Hence, we can only calculate the expected number of occurrences of base pairs for two columns c and c'. Thus, we redefine bp_cons(c, c') as follows. Consider a multiple alignment of K sequences. For each sequence S_k, let $\mathcal{E}_k$ be the ensemble of structures calculated for S_k. For each column c, let i_c^k be either the position that corresponds to column c in sequence S_k (if aligned), or **–** otherwise. Furthermore, let $\delta_P(c, c')$ be the index function of P, i.e. $\delta_P(c, c')$ is 1 if $(c, c') \in P$, and 0 otherwise. Then

$$\text{bp\_cons}(c, c') = \sum_{k=1}^{K} \sum_{P \in \mathcal{E}_k} \delta_P \left(i_c^k, i_{c'}^k \right) \cdot \Pr\left[P \mid \mathcal{E}_{S_k} \right],$$

where $\Pr[P \mid \mathcal{E}_{S_k}]$ is defined as given in Equation (1).

Complexity

Here, we assume that all sequences have nearly the same length L and that we generate an ensemble of E structures for each sequence. Note that by using RNAshapes, E is typically small (up to three sequences). The running time of one pairwise alignment is $O(E^2L^4)$. We have to make $N(N-1)/2$ comparisons in a set of N RNAs. Therefore, the pairwise comparison step needs $O(E^2N^2L^4)$ computation time. The most time consuming part of the multiple alignment step consists of building the extended library, which takes $O(N^3L^2)$ steps in the worst case (Notredame *et al*., 2000). Altogether, the dominating alignment complexity is given by

$$O(E^2N^2L^4) + O(N^3L^2)$$

APPLICATION

Our algorithmic approach of multiple alignment of RNAs can be used online at our MARNA server. The maximal sequence length of one RNA is restricted to 500 bases. The maximal number of RNAs depends on the sequence lengths. The sum of all sequence lengths is restricted to 10 000 bases. MARNA is capable of aligning RNA sequences with known as well as unknown secondary structures. In the latter case, the user can choose whether to assign for every sequence a known structure or an ensemble of several structures automatically generated by RNAshapes or by stochastic backtracking (as part of RNAsubopt, implemented in the Vienna RNA Package).

We have tested MARNA on three datasets from the Rfam database (Griffiths-Jones *et al*., 2003). We compared the alignments as well as the consensus structures given from the Rfam database with the output of the two different alignment tools T-Coffee and PMmulti. The manual alignment and the consensus structure from the Rfam database serve as the reference. For MARNA and PMmulti, we have compared the consensus structures as proposed by the programs. If RNAalifold (Hofacker *et al*., 2002) yielded a better consensus structure, we have also displayed this one. For T-Coffee, we have used RNAalifold (Hofacker *et al*., 2002) to predict the consensus structure.

The first dataset consists of seven randomly chosen eukaryotic SECIS-elements (selenocysteine insertion sequence, Rfam accession number RF00031). SECIS-elements are necessary for the incorporation of selenocysteine into a protein sequence directed by an in-frame UGA codon (usually a stop codon) within the coding region of the mRNA. Selenoprotein mRNAs contain a conserved secondary structure in the 3′ UTR that is required for the distinction of UGA stop from UGA selenocysteine. The sequences are ∼60 nt in length and adopt a hairpin structure that is sufficiently well-defined and conserved, but the primary sequences differ. This dataset is especially hard for sequence–structure alignment programs since it contains four non-standard base pairs (U–U, G–A, A–G, C–U) in the lower part of the stem.

The manual alignment was made from 25 SECIS-elements out of a total set of currently 65 SECIS-elements. We have chosen 7 out of the 25 sequences randomly. The manual alignment is shown in Figure 3 and serve as the 'true' alignment. An alignment with T-Coffee reveals some nucleotide similarities among sequences, as expected, but are not suited for the prediction of a consensus structure. Here, MARNA detects the long stem structure with the characteristic bulged A's in the upper loop (Lambert *et al*., 2002). For this test case, we used the predicted minimum free energy (mfe) structure for each sequence.

We also aligned all the 65 SECIS-elements using MARNA with mfe structures and with ensembles calculated by RNAshapes and by stochastic backtracking (as part of RNAsubopt, Vienna RNA Package). We have compared these results with the results of PMmulti. The results are summarized in Table 2.

The second dataset consists of 22 tRNA-like structures, found in the 3′ UTR of Tymoviruses and Pomoviruses. They were also taken from the Rfam database. The family is thought to be involved in the initiation of minus-strand synthesis and the disruption of the pseudoknot gives rise to a 50% drop in transcription efficiency. MARNA (both with mfe structures as well as with structure ensembles) is able to detect the four stems as well as the single G between the first and second stem. For PMmulti, the four stems are detected when using RNAalifold in addition (Fig. 4).

Finally, we have used the objective function given by Bali Base benchmark program (Thompson *et al*., 1999) to compare all generated alignments, again taking the Rfam alignments as a reference. The benchmark program returns two scores, namely SP (sum of pairs) and TC (total columns). SP measures the ratio of the number of correctly aligned pairs, whereas TC measures the number of correctly aligned columns. Since conservation of columns is different in sequence alignments and sequence–structure alignments, we have used only the SP-score. The results are summarized in Table 2.

DISCUSSION

We have presented a multiple alignment method for RNAs considering both the primary sequences and the secondary structures. It generates pairwise sequence–structure alignments and combines them using T-Coffee. Hence, MARNA is not only a structure alignment tool, but also considers sequence similarities. The main advantage is to set individual parameter values capable of weighting either sequence or structure properties. Concerning structures, one can use either user-defined structures, or let MARNA predict an ensemble of low energy structures.

MARNA can be tested online on our website. Although a pairwise comparison needs time complexity of $O(n^2m^2)$ for two RNAs of

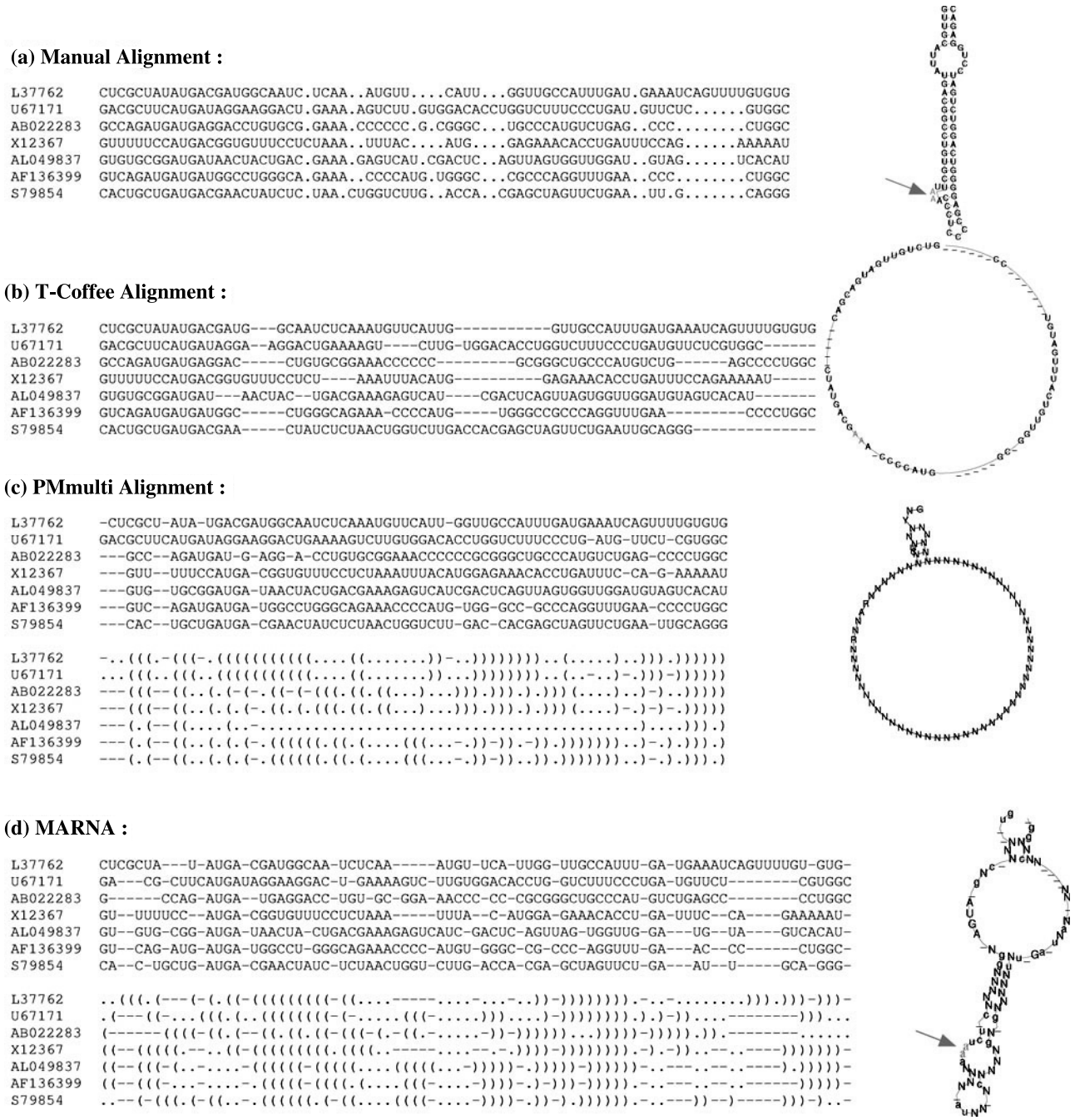


Fig. 3. Comparison of multiple alignments and consensus structure predictions of seven randomly chosen SECIS-elements taken from the Rfam database. (**a**) The restriction of the manual alignment in Rfam to the seven sequences. The consensus structure is proposed by Rfam, i.e. generated of all 65 SECIS elements. (**b**) The T-Coffee alignment is based on nucleotide similarities and thus disregards structural conformations. It is not surprising that the consensus structure contains no base-pairing interactions. (**c**) PMmulti detects some bonds, but does not identify the hairpin. (**d**) MARNA finds the hairpin-like structure as well as the buldged A's in the upper loop of the motif (indicated by a red arrow), which are required for the SECIS-element.

lengths n and m, and thus limits the input sequence lengths to 500 bases, MARNA has been tested successfully on many RNAs like tRNAs, rRNAs and ncRNAs.

This paper is based on a previous version published in the German Conference of Bioinformatics (Siebert and Backofen, 2003). It has been extended in several ways. We have added the use of structure ensembles and provided a formal description for the calculation of the alignment weights. Furthermore, we added the calculation of a consensus structure and compared our results with PMmulti using the Balibase benchmark test.

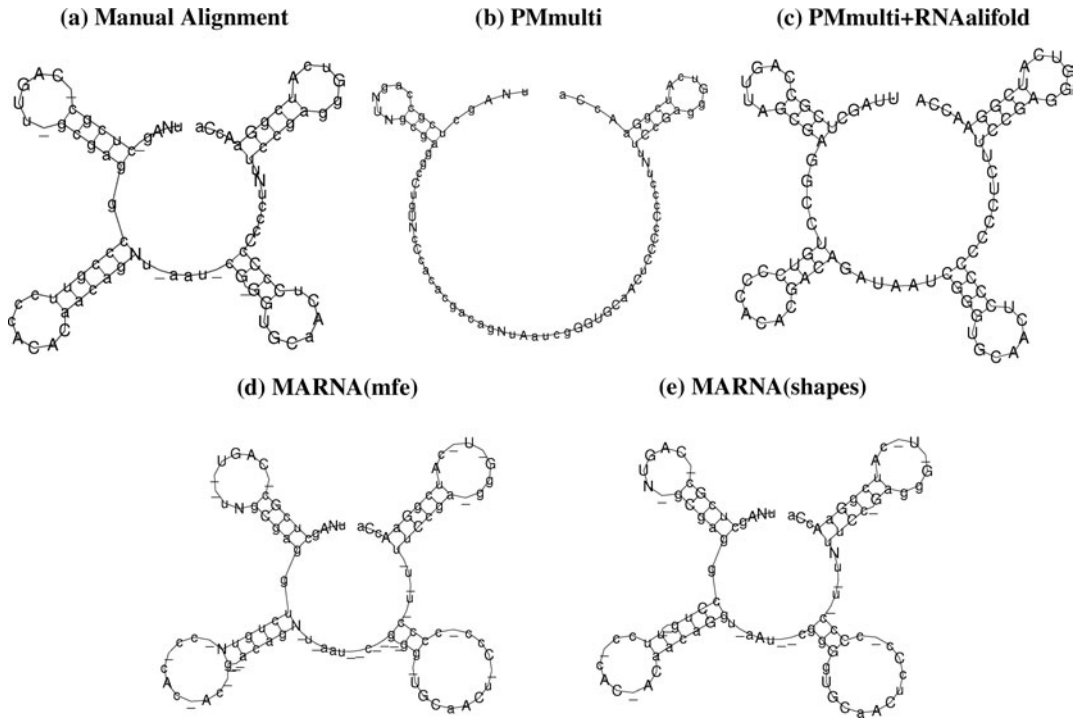


Fig. 4. Consensus structure predictions for 22 tRNA-like structures, found in Tymovirus and Pomovirus. (**a**) The consensus structure of the manual alignment contains four distinct stems. (**b**) The proposed structure of PMmulti detects two of the four stems. (**c**) An improvement of the structure prediction applied to the same computed alignment can be achieved by RNAalifold (part of the Vienna RNA package). (**d** and **e**) MARNA is able to detect all four stems in case of assigning the mfe structure as well as the shape structure to each sequence. Both consensus structures are very close to the Rfam consensus structure.

Table 2. Evaluation of MARNA and PMmulti alignments using the SP-score of the Bali Base benchmark program

Sequences	RFam Accession-No.	MARNA (mfe)	MARNA (shapes)	MARNA (ens)	PMmulti
SECIS-elements (7 rand.)	RF00031	0.327	0.351	**0.545**	0.286
SECIS-elements (all 65)	RF00031	0.463	**0.487**	0.447	0.162
Tymovirus/ Pomovirus	RF00233	0.715	0.782	**0.837**	0.730
Hammerhead	RF00008	0.785	**0.811**	0.742	0.696[a]

The first and third dataset have been already analyzed in Figures 3 and 4. Additionally, we compared the whole dataset of the 65 SECIS-elements and the Hammerhead ribozymes (type III). MARNA has been tested in different combinations, namely with mfe structures and structure ensembles generated by RNAshapes and by the stochastic backtracking of RNAsubopt (using three structures per sequence). Values in the table indicate the similarity to the reference manual alignment. They vary between 0 and 1. The maximum value for each test set is highlighted. They are reached by MARNA alignment with shape and ensemble structures.

[a]Note that for the Hammerhead set, PMmulti aligned only 78 of the 85 sequences. Here, the reference alignment is the subalignment of Rfam corresponding to these 78 sequences.

ACKNOWLEDGEMENTS

The authors would like to thank Anke Busch, Michael Hiller and Sebastian Will for reading the manuscript and for their helpful comments. This work was partially supported by the DFG within the national project ‘Selenoproteine’.

Conflict of Interest: none declared.

REFERENCES

Backofen,R. and Will,S. (2004) Local sequence-structure motifs in RNA. *Journal of Bioinformatics and Computational Biology (JBCB)*, **2**, 681–698.

Brown,M.P. (1999) *RNA Modeling Using Stochastic Context-Free Grammars*. Ph. D. thesis, University of California, Santa Cruz.

Chiu,D.K. and Kolodziejczak,T. (1991) Inferring consensus structure from nucleic acid sequences. *Comput. Appl. Biosci.*, **7**, 347–352.

Corpet,F. and Michot,B. (1994) RNAlign program: alignment of RNA sequences using both primary and secondary structures. *Comput. Appl. Biosci.*, **10**, 389–399.

Couzin,J. (2002) Breakthrough of the year. Small RNAs make big splash. *Science*, **298**, 2296–2297.

Dowell,R.D. and Eddy,S.R. (2004) Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction. *BMC Bioinformatics*, **5**, 71.

Eddy,S.R. and Durbin,R. (1994) RNA sequence analysis using covariance models. *Nucleic Acids Res.*, **22**, 2079–2088.

Giegerich,R. *et al*. (2004) Abstract shapes of RNA. *Nucleic Acids Res.*, **32**, 4843–4851.

Griffiths-Jones,S. *et al*. (2003) Rfam: an RNA family database. *Nucleic Acids Res.*, **31**, 439–441.

Gutell,R.R. *et al*. (1992) Identifying constraints on the higher-order structure of RNA: continued development and application of comparative sequence analysis methods. *Nucleic Acids Res.*, **20**, 5785–5795.

Gutell,R.R. and Woese,C.R. (1990) Higher order structural elements in ribosomal RNAs: pseudo-knots and the use of noncanonical pairs. *Proc. Natl Acad. Sci. USA*, **87**, 663–667.

Höchsmann,M. *et al*. (2003) Local similarity in RNA secondary structures. In *Proceedings of Computational Systems Bioinformatics (CSB 2003)*, Stanford, CA, pp. 159–168.

Hofacker,I.L. (2003) Vienna RNA secondary structure server. *Nucleic Acids Res.*, **31**, 3429–3431.

Hofacker,I.L. *et al*. (2004) Alignment of RNA base pairing probability matrices. *Bioinformatics*, **20**, 2222–2227.

Hofacker,I.L. *et al*. (2002) Secondary structure prediction for aligned RNA sequences. *J. Mol. Biol.*, **319**, 1059–1066.

Jiang,T. *et al*. (2002) A general edit distance between RNA structures. *J. Comput. Biol.*, **9**, 371–388.

Jiang,T. *et al*. (1995) Alignment of trees—an alternative to tree edit. *Theore. Comp. Sci.*, **143**, 137–148.

Knudsen,B. and Hein,J. (2003) Pfold: RNA secondary structure prediction using stochastic context-free grammars. *Nucleic Acids Res.*, **31**, 3423–3428.

Lambert,A. *et al*. (2002) A survey of metazoan selenocysteine insertion sequences. *Biochimie*, **84**, 953–959.

Lenhof,H.P. *et al*. (1998) A polyhedral approach to RNA sequence structure alignment. *J. Comput. Biol.*, **5**, 517–530.

Lin,G.-H., Ma,B. and Zhang,K. (2001) Edit distance between two RNA structures. In *Proceedings of the 5th Annual International Conferences on Compututational Molecular Biology (RECOMB'01)*. ACM Press, Montreal, Canada.

Luck,R. *et al*. (1999) Construct: a tool for thermodynamic controlled prediction of conserved secondary structure. *Nucleic Acids Res.*, **27**, 4208–4217.

Morgenstern,B. *et al*. (1998) DIALIGN: finding local similarities by multiple sequence alignment. *Bioinformatics*, **14**, 290–294.

Notredame,C. *et al*. (2000) T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.*, **302**, 205–217.

Nussinov,R. *et al*. (1978) Algorithms for loop matchings. *SIAM J. Appl. Math.*, **35**, 68–82.

Saitou,N. and Nei,M. (1987) The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, **4**, 406–425.

Sankoff,D. (1985) Simultaneous solution of the RNA folding, alignment and protosequence problems. *SIAM J. Appl. Math.*, **45**, 810–825.

Shapiro,B.A. and Zhang,K.Z. (1990) Comparing multiple RNA secondary structures using tree comparisons. *Comput. Appl. Biosci.*, **6**, 309–318.

Siebert,S. and Backofen,R. (2003) Marna: a server for multiple alignment of rnas. In *GCB 2003*, Neremberg, Gatching near Munich, Germany, pp. 135–140.

Smith,T. and Waterman,M. (1981) Comparison of biosequences. *Adv. Appl. Math.*, **2**, 482–489.

Thompson,J.D. *et al*. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.

Thompson,J.D. *et al*. (1999) A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res.*, **27**, 2682–2690.

Zuker,M. (1994) Prediction of RNA secondary structure by energy minimization. *Meth. Mol. Biol.*, **25**, 267–294.