*Bioinformatics*, 31(24), 2015, 3914–3921
doi: 10.1093/bioinformatics/btv471
Advance Access Publication Date: 13 August 2015
Original Paper

OXFORD

Structural bioinformatics

# Pairwise RNA secondary structure alignment with conserved stem pattern

**Jimmy Ka Ho Chiu and Yi-Ping Phoebe Chen***

Department of Computer Science and Information Technology, La Trobe University, Melbourne, Victoria 3086, Australia

*To whom correspondence should be addressed.

Associate Editor: Ivo Hofacker

Received on September 8, 2014; revised on August 4, 2015; accepted on August 7, 2015

## Abstract

**Motivation:** The regulatory functions performed by non-coding RNAs are related to their 3D structures, which are, in turn, determined by their secondary structures. Pairwise secondary structure alignment gives insight into the functional similarity between a pair of RNA sequences. Numerous exact or heuristic approaches have been proposed for computational alignment. However, the alignment becomes intractable when arbitrary pseudoknots are allowed. Also, since non-coding RNAs are, in general, more conserved in structures than sequences, it is more effective to perform alignment based on the common structural motifs discovered.

**Results:** We devised a method to approximate the true conserved stem pattern for a secondary structure pair, and constructed the alignment from it. Experimental results suggest that our method identified similar RNA secondary structures better than the existing tools, especially for large structures. It also successfully indicated the conservation of some pseudoknot features with biological significance. More importantly, even for large structures with arbitrary pseudoknots, the alignment can usually be obtained efficiently.

**Availability and implementation:** Our algorithm has been implemented in a tool called PSMAlign. The source code of PSMAlign is freely available at http://homepage.cs.latrobe.edu.au/ypchen/psmalign/.

**Contact:** phoebe.chen@latrobe.edu.au

**Supplementary information:** Supplementary data are available at *Bioinformatics* online.

## 1 Introduction

The interest in RNA research was reinvigorated after discovering its regulatory and catalytic roles in many biological processes (Couzin, 2002; Storz, 2002). Those RNAs with such roles are called the non-coding RNAs (ncRNAs). They perform a diverse range of functions, such as regulating gene expression (Serganov and Patel, 2007), modifying ribosomal RNA (Maden and Hughes, 1997), and controlling muscle differentiation (Cesana *et al.*, 2011). It is believed that a specific function of an ncRNA molecule is related to its structure (Chen and Chen, 2009; Lee *et al.*, 1997). Strong evidence is that ncRNAs are more evolutionarily conserved in their structures than their sequences (Reiter *et al.*, 2011). Therefore, the two main studies in this area are the structure determination and the comparison among different structures. The former can be subdivided into experimental analysis (Scott and Hennig, 2008) and computational prediction (Rivas and Eddy, 1999; Zuker and Stiegler, 1981) using experimentally determined thermodynamics parameters (Turner and Mathews, 2010). On the other hand, structure comparison is purely computational. Moreover, since the structure is critical to the function, RNA molecules are compared with both structures and constituent base sequences.

In this article, we are interested in the pairwise RNA secondary structure comparison problem. This problem can be further classified into the edit problem and the alignment problem (Denise and Rinaudo, 2014). The edit problem computes the minimum cost (distance) of modifying a given structure $R_1$ to another given structure $R_2$,

using a series of pre-defined edit operations. The alignment problem finds a consensus structure $R_c$ such that the total edition cost from $R_1$ and $R_2$ to $R_c$ is minimized. Early approaches model the RNA secondary structures as trees and consider the edit distance or similarity score among them (Jiang *et al.*, 1995; Zhang and Shasha, 1989). Afterwards, various tools were developed, of which some produce optimal solutions, while the others are heuristics. RNA_align (Lin *et al.*, 2001), RNAForester (Hochsmann *et al.*, 2003, 2004) and Gardenia (Blin *et al.*, 2010) are some examples that belong to the exact solution category. The average time complexity for the last two methods is $O(n^2)$, where $n$ is the length of the longer structure. (Herrbach *et al.*, 2010) and the worst case complexity is $O(n^4)$ (Denise and Rinaudo, 2014). MiGaL (Allali and Sagot, 2008) and RNAStrAT (Guignon *et al.*, 2005) belong to the heuristics category.

There are still two shortcomings impacting the usefulness and efficiency of these approaches. Firstly, most of them do not support pseudoknots since aligning structures with arbitrary pseudoknots is NP-hard (Jiang *et al.*, 2002), but pseudoknots have already been known to perform diverse functions in many biological processes (Staple and Butcher, 2005; Wadkins *et al.*, 1999). Significant comparison results could be obtained when such crossing motifs can be aligned, because they are usually highly conserved (Theimer *et al.*, 2005). An approach allowing arbitrary pseudoknots for exact alignment has its time complexity dependent on the complexity of the input structures (Möhl *et al.*, 2008). On the other hand, the polynomial time alignment algorithm is possible for a restricted set of pseudoknots (Evans, 2006). However, both of them are still inadequate to compare the complex crossing motifs that have already been identified (Chiu and Chen, 2012). More importantly, none of them are available as a tool. Second, since ncRNAs often exhibit better conservation in their structures, exact alignment methods might not indicate some biologically significant structural features. This is because they consider both base sequences and structures equally, and hence some conserved structural motifs that are dissimilar in their sequences are difficult to align. Conversely, some motifs are spuriously aligned, disrupting the overall alignment. To overcome these limitations, we devised a heuristics, targeting the pairwise alignment of large RNA secondary structures with arbitrary pseudoknots.

## 2 Methods

### 2.1 Conserved stem pattern in similar RNA secondary structure pair

Our proposition is that ncRNA secondary structures are more conserved than their sequences. Hence, a pair of similar secondary structures shares a conserved stem pattern. The stem pattern is a subset of all base pair stems in a given secondary structure, plus the structural relations among those in this subset. Our approach attempts to discover a conserved stem pattern between a pair of secondary structures, and constructs a structure alignment from it. Since an alignment function is applied to evaluate the alignment costs among stems and unpaired strands, our heuristics solves the ALIGN(CROSSING × CROSSING → CROSSING) problem.

The stem pattern is modeled by a stem graph (Hamada *et al.*, 2006). This graph illustrates three structural relationships for every pair of base pair helices (stacks) in its edges, namely parallel (P), nested (N) and pseudoknotted (K). For any base pair $(i_1, j_1)$ in stack $s_1$ and any base pair $(i_2, j_2)$ in stack $s_2$, $s_1$ and $s_2$ are parallel when $i_1 < j_1 < i_2 < j_2$. $s_2$ is nested in $s_1$ when $i_1 < i_2 < j_2 < j_1$, and $s_1$ is a nesting stack. They are pseudoknotted and are called crossing stacks when $i_1 < i_2 < j_1 < j_2$. Figure 1 illustrates these structural relationships in a stem graph of a sample secondary structure. Each vertex of the original stem graph models a stack, but we merge some nested stacks to a stem and model it by a single vertex to reduce graph order. The stack merging is introduced in the supplementary material. A stem graph $G$ is defined as $G = (V, E, l)$, where $V$ is the vertex set, $E = V \times V$ is the edge set as $G$ is complete, and $l$: $E \rightarrow \Sigma$ is a function mapping every edge to its edge label.

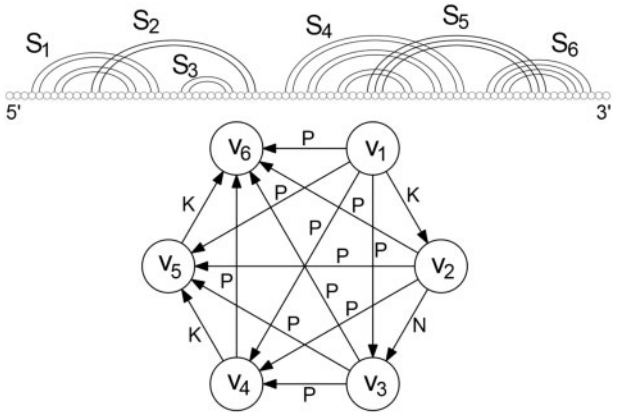


**Fig. 1.** Top: A sample RNA secondary structure. Bottom: Stem graph modeling the sample structure, where vertex $v_i$ represents stem $s_i$. The stem structural relations are: parallel (P), nested (N) and pseudoknotted (K). The secondary structure is drawn using VARNA (Darty *et al.*, 2009)

Given a predefined alignment function, the best conserved stem pattern is one such that the sum of alignment costs for all mapped stem pairs, plus the deletion cost for each unmapped stem, is minimized. Because each stem graph vertex represents a unique stem, the conserved stem pattern can be represented by a set of mapped vertex pairs between the two stem graphs. Assuming a bijective stem matching in the conserved stem pattern, this set is equivalent to a bijective function $m$: $V_{\text{map}} \rightarrow V'_{\text{map}}$ where $V_{\text{map}} \subseteq V$, $V'_{\text{map}} \subseteq V'$, and $V$, $V'$ are the vertex sets of the two stem graphs. $m$ is called the error-correcting graph matching (ECGM) which was originally proposed for pattern recognition (Bunke, 1997). So, the best conserved stem pattern is graphically equivalent to a minimum cost ECGM (mcECGM), with its cost regarded as the edit distance between the two graphs. Denote $G = (V, E, l, p)$ and $G' = (V', E', l', p')$ to be the stem graphs for the two RNA secondary structures $R$ and $R'$, respectively. The additional function $p$: $V \rightarrow S$ maps each vertex to its corresponding base pair stem in the set $S$ of all stems. For an ECGM $m$: $V_{\text{map}} \rightarrow V'_{\text{map}}$, the costs are defined for the specific edit operations suggested in (Bunke, 1997) as follows:

Vertex substitution cost, $c_{\text{vs}}(v, v') = a(p(v), p'(v'))$ where $v \in V_{\text{map}}$ and $v' \in V'_{\text{map}}$

Vertex deletion cost, $c_{\text{vd}}(v) = a(p(v), \eta)$ where $v \in V - V_{\text{map}}$ and $\eta$ is an empty base sequence

Vertex insertion cost, $c_{\text{vi}}(v') = a(p'(v'), \eta)$ where $v' \in V' - V'_{\text{map}}$

Edge substitution cost, $c_{\text{es}}(e, e') = \begin{cases} 0 & \text{if } l(e) = l'(e') \\ \infty & \text{otherwise} \end{cases}$

where $e \in V_{\text{map}} \times V_{\text{map}}$ and $e' \in V'_{\text{map}} \times V'_{\text{map}}$

Edge deletion cost, $c_{\text{ed}}(e) = 0$ where $e \in E - V_{\text{map}} \times V_{\text{map}}$

Edge insertion cost, $c_{\text{ei}}(e') = 0$ where $e' \in E' - V'_{\text{map}} \times V'_{\text{map}}$

$a$ is the predefined alignment function giving alignment cost. The best conserved stem pattern modeled by $m$ exists in both $R$ and $R'$, because the edge substitution cost $c_{\text{es}}$ ensures an identical structural relation between any two mapped stem pairs.

## 2.2 Progressive stem matching to approximate the true conserved stem pattern

On the other hand, the best conserved stem pattern defined above might be different from the true conserved stem pattern, because the edit distance of the stem pattern excludes the unpaired base regions (e.g. hairpin loops) that are also deterministic in structure alignment. Also, ECGM is in general NP-complete (Bunke, 1997). While an exact algorithm is scalable with some RNA families, for large RNA molecules, such as 16S and 23S ribosomal RNAs, the number of base pair stems and the stem graph order might exceed 100. An exhaustive search becomes intractable for such a graph order. It is then necessary to reduce the search space by selecting mapped vertex pairs that are more probable to appear in the ECGM representing the true pattern. We introduce a heuristics called progressive stem matching. It identifies similar nesting motifs and then non-nesting motifs in both input structures, and computes the mcECGM from these results afterwards. This mcECGM models the approximate true conserved stem pattern. The core algorithms for each of its phases are stated below. An example illustrating these phases is provided in the supplementary material.

### 2.2.1 Phase 1: Initialization

We adapt the definitions of stack and stem proposed in (Rødland, 2006), where a base pair stem consists of one or more base pair stacks joined by bulges or internal loops. Given a secondary structure, this phase first merges some of its nested stacks into stems (the merging mechanism is described in the supplementary material), and then generates its stem graph $G=(V, E, l, p)$. By determining whether its underlying stem is nesting for every vertex, $V$ is partitioned into $V_{\text{nest}}$ and $V_{\text{non-nest}}$ such that a stem represented by $v$ in $V_{\text{nest}}$ is nesting; otherwise $v$ is in $V_{\text{non-nest}}$. A vertex in $V_{\text{nest}}$ is a nesting vertex, and a vertex in $V_{\text{non-nest}}$ is a non-nesting vertex. $V'$ of another structure is partitioned into $V'_{\text{nest}}$ and $V'_{\text{non-nest}}$.

### 2.2.2 Phase 2: Identifying highly similar non-nesting motifs

The underlying stem for a vertex in $V_{\text{non-nest}}$ or $V'_{\text{non-nest}}$ forms a non-nesting motif, which consists of the stem itself and the loop segment enclosed by it. The green boxes of Supplementary Figure S3A provide some examples. We identify highly similar non-nesting motifs in both structures, and then generate the mapped vertex pairs accordingly. These pairs are utilized to search similar nesting motifs in phase 3. The algorithm appended at the end of this sub-section is described as follows: The motif alignment cost ratio $r_b$ for every possible non-nesting motif alignment pair (line 1) is first calculated, where $r_b$ is defined as:

$$r_b(v,v') = \frac{a(h(p(v)),h'(p'(v')))}{\min\{a(h(p(v)),\eta),a(h'(p'(v')),\eta)\}} \quad (1)$$

$a$ is the alignment function returning the cost, $h$ and $h'$ extract non-nesting motifs bounded by the stem obtained from $p(v)$ and $p'(v')$ respectively. $\eta$ is a null motif. $r_b$ is the alignment cost averaged by the lower motif removal cost. For each non-nesting motif in both structures, its lowest $r_b$ value (indicating best alignment) is kept as $r_{b\_\min}$ (lines 2 – 3). Its best and suboptimal aligned counterparts are added to $L$ (line 4) such that their $r_b$ values are at most $f$ times of $r_{b\_\min}$. So, $f$ controls the selection of the best and suboptimal aligned counterparts into $L$. mcECGMs are computed from $L$ to remove outliers. For efficiency reasons, mcECGMs are obtained progressively with a selection range $b=(b_{\text{start}}, b_{\text{end}}]$. $b$ first selects highly similar motif pairs in $L$ (lines 5 – 6) to build initial partial mcECGMs. Those with at least μ pairs are kept (line 6), and μ is the minimum size requirement of the initial mcECGMs for robust expansion to the final mcECGMs. When at least one qualified initial mcECGM is found, $b$ is shifted to the next range of width Δ (line 7) to select the next less similar pairs in $L$ for iterative expansion (line 10). Otherwise, $b_{\text{end}}$ is incremented by the step size Δ (line 7) to repeat the initial mcECGM finding (line 8). The one with the most vertex pairs in the final mcECGMs is reported as $m_b$. The vertex edit operation costs in this phase are motif based, i.e. $c_{vs}(v, v')=a(h(p(v)), h'(p'(v')))$, $c_{vd}(v)=a(h(p(v)), \eta)$ and $c_{vi}(v')=a(h'(p'(v')), \eta)$, and the edge edit operation costs are identical to Section 2.1. When a non-nesting stem is crossing, the other stems with which it crosses are not considered during its motif alignment.

**Algorithm 1:** Find consistent similar non-nesting motif pairs

1 Calculate $r_b(v, v')$ using Equation (1) for each $(v, v') \in V_{\text{non-nest}} \times V'_{\text{non-nest}}$
2 Obtain $r_{b\_\min}(v)$ by $r_{b\_\min}(v)=min\{r_b(v, v') \mid \forall\, v' \in V'_{\text{non-nest}}\}$ for each $v \in V_{\text{non-nest}}$
3 Repeat line 2 to obtain $r_{b\_\min}(v')$ for each $v' \in V'_{\text{non-nest}}$
4 Create a list $L=\{(v, v') \mid r_b(v, v') \le f \times r_{b\_\min}(v)$ or $r_b(v, v') \le f \times r_{b\_\min}(v')\ \forall\, (v, v') \in V_{\text{non-nest}} \times V'_{\text{non-nest}}\}$
5 Initialize a selection range $b=(b_{\text{start}}, b_{\text{end}}]$ where $b_{\text{start}}=-1$
6 Obtain all mcECGMs with the elements in $L$ whose $r_b(v, v')$ are within the $b$. Add those mcECGMs with at least μ mapped vertex pairs to $M_b$
7 Modify $b$ such that $b_{\text{start}}=b_{\text{end}}$ when $M_b \neq \emptyset$, and $b_{\text{end}} += \Delta$
8 Repeat from line 6 when $M_b == \emptyset$
9 Expand all mcECGMs in $M_b$ with the elements in $L$ whose $r_b(v, v')$ are within the updated selection range $b$
10 Repeat from line 7 until all elements in $L$ have been examined
11 Return the largest size mcECGM (denoted by $m_b$) in $M_b$

### 2.2.3 Phase 3: Obtaining partial mcECGMs by identifying similar nesting motifs

The underlying stem for a vertex in $V_{\text{nest}}$ or $V'_{\text{nest}}$ forms a nesting motif, which consists of the stem itself, and any other motifs and loops nested in it. The red box of Supplementary Figure S3A shows an example. This phase starts building mcECGMs from $V_{\text{nest}}$ and $V'_{\text{nest}}$ by identifying similar nesting motifs. However, such motifs are sometimes large, and nested pseudoknots may also be present, hence it is inefficient to align such motifs. Alternatively, a structure profile $Q(v)$ ($Q(v')$) is created for each $v$ in $V_{\text{nest}}$ ($v'$ in $V'_{\text{nest}}$) to estimate motif similarity using $m_b$ obtained in phase 2. The first set in $Q(v)$, $N(v)$, represents non-nesting stems nested in the underlying stem $s$ of $v$. The second set $C_{\text{pv}}(v)$ represents non-nesting stems crossing $s$ (i.e. they precede $s$ in 5′ to 3′ direction), and the last set $C_{\text{sq}}(v)$ represents non-nesting stems crossed by $s$ (i.e. $s$ precedes them) (line 1). A match score and a mismatch score are evaluated for every nesting motif pair. The match score $z_{\text{match}}(v, v')$ is the number of mapped vertex pairs $(u, u')$ in $m_b$ such that the structural relation (limited to the three relations defined in $Q(v)$) between $u$ and $v$ is identical to that between $u'$ and $v'$ (line 4). This score indicates how many non-nesting motifs in the profiles are conserved between the two nesting motifs. On the other hand, a mismatch between $v$ and $v'$ occurs when a non-nesting vertex $u$ (or $u'$) appearing in a set of $Q(v)$ (or $Q(v')$) belongs to a mapped pair $(u, u')$ in $m_b$, but $u'$ (or $u$) is not

in same set of $Q(v')$ (or $Q(v)$). Because such mismatch can occur twice for a mapped vertex pair $(u, u')$ in $m_h$, 0.5 is multiplied to obtain the mismatch score $z_{\text{mismatch}}(v, v')$ (line 5). The net similarity score ($z$ score) is then calculated (line 6). The more positive $z$ score the more likely the underlying motif of $v$ is similar to that of $v'$. A nesting vertex pair $(v, v')$ is added to the candidate list $T$ when its $z$ score is in top $K$ for both $v$ and $v'$ (lines 8–10). mcECGMs $M_{\text{nest}}$ are obtained from $T$ (line 11). The ECGM edit operation costs applied in this phase are defined in Section 2.1.

**Algorithm 2:** Discover partial mcECGMs from nesting stems

1 Create a structure profile $Q(v) = (N(v), C_{\text{pv}}(v), C_{\text{sq}}(v))$, where $N(v) = \{u \mid u \in V_{\text{non-nest}}$ and $l((v, u)) = =$ 'N'$\}$, $C_{\text{pv}}(v) = \{u \mid u \in V_{\text{non-nest}}$ and $l((u, v)) = =$ 'K'$\}$ and $C_{\text{sq}}(v) = \{u \mid u \in V_{\text{non-nest}}$ and $l((v, u)) = =$ 'K'$\}$ for each $v \in V_{\text{nest}}$
2 Repeat line 1 to obtain $Q(v')$ for each $v' \in V'_{\text{nest}}$
3 For each $(v, v') \in V_{\text{nest}} \times V'_{\text{nest}}$
4 $z_{\text{match}}(v, v') = |\{(u, u') \in m_h \mid (u \in N(v)$ and $u' \in N(v'))$ or $(u \in C_{\text{pv}}(v)$ and $u' \in C_{\text{pv}}(v'))$ or $(u \in C_{\text{sq}}(v)$ and $u' \in C_{\text{sq}}(v'))\}|$
5 $z_{\text{mismatch}}(v, v') = |\{u \mid (u, u') \in m_h$, $(u \in N(v)$ and $u' \notin N(v'))$ or $(u \in C_{\text{pv}}(v)$ and $u' \notin C_{\text{pv}}(v'))$ or $(u \in C_{\text{sq}}(v)$ and $u' \notin C_{\text{sq}}(v'))\}| \times 0.5 + |\{u' \mid (u, u') \in m_h$, $(u \notin N(v)$ and $u' \in N(v'))$ or $(u \notin C_{\text{pv}}(v)$ and $u' \in C_{\text{pv}}(v'))$ or $(u \notin C_{\text{sq}}(v)$ and $u' \in C_{\text{sq}}(v'))\}| \times 0.5$
6 $z(v, v') = z_{\text{match}}(v, v') - z_{\text{mismatch}}(v, v')$
7 End for
8 Select $(v, v^\circ)$ with top $K$ highest positive $z$ scores (only among those involving $v$) into $Z_{\text{top}}(v)$ for each $v \in V_{\text{nest}}$
9 Repeat line 8 to obtain $Z_{\text{top}}(v')$ for each $v' \in V'_{\text{nest}}$
10 Add $(v, v')$ to $T$ when $(v, v') \in Z_{\text{top}}(v)$ and $(v, v') \in Z_{\text{top}}(v')$ for each $(v, v') \in V_{\text{nest}} \times V'_{\text{nest}}$
11 Return all mcECGMs $M_{\text{nest}}$ obtained with the elements in $T$

### 2.2.4 Phase 4: Expanding mcECGMs with $V_{\text{non-nest}}$ and $V'_{\text{non-nest}}$

The mismatch case in the last phase suggests that some partial mcECGMs in $M_{\text{nest}}$ can be in conflict with $m_h$ obtained in phase 2, so they cannot be merged. This phase expands each mapping $m_{\text{nest}}$ in $M_{\text{nest}}$ with non-nesting vertex pairs identified from similar non-nesting motifs. Using $m_{\text{nest}}$, the original problem can be broken into smaller problems that are solved more efficiently. The algorithm at the end of this sub-section is explained below.

The $|m_{\text{nest}}|$ vertices mapped in each of $V_{\text{nest}}$ and $V'_{\text{nest}}$ correspond to $|m_{\text{nest}}|$ nesting stems, as well as $2|m_{\text{nest}}|$ stem (upstream and downstream) regions in each structure. The sequence backbones of $R$ and $R'$ can then be divided into $2|m_{\text{nest}}| + 1$ non-overlapping backbone segments $q_t$ and $q'_t$, respectively (line 3), some of which can be of length zero. The two stem regions of a non-nesting stem can be contained in the same or different segments. Hence, any vertex in $V_{\text{non-nest}}$ ($V'_{\text{non-nest}}$) can be assigned to partition $V_q(i, j)$ ($V'_q(i, j)$), where $i$ and $j$ stand for the segment numbers containing the upstream and downstream regions of the non-nesting stem, respectively (line 4), with $i \leq j$. Some non-nesting stems might cross with each other, and so their own partitions, say $V_q(i, j)$ and $V_q(k, n)$, as well as their counterparts $V'_q(i, j)$ and $V'_q(k, n)$ are merged (line 6). Figure 2 shows the six possible scenarios requiring partition merging. Afterwards, any vertex $u$ in a partition can only have parallel relationship with any vertex $v$ in another partition (i.e. $l(u, v) =$ 'P' or $l(v, u) =$ 'P') in both graphs. As a result, any mcECGM from a candidate space $V_q(i, j) \times V'_q(i, j)$ does not depend on the mcECGM outcomes in another candidate space. They are said to be independent. Moreover, the partitioning ensures the consistency between the mcECGMs with $m_{\text{nest}}$. They are then merged as the outputs of this phase (line 7). The ECGM edit operation costs applied are identical to phase 2. Again, any interactions of the nested loop bases with other bases are ignored.

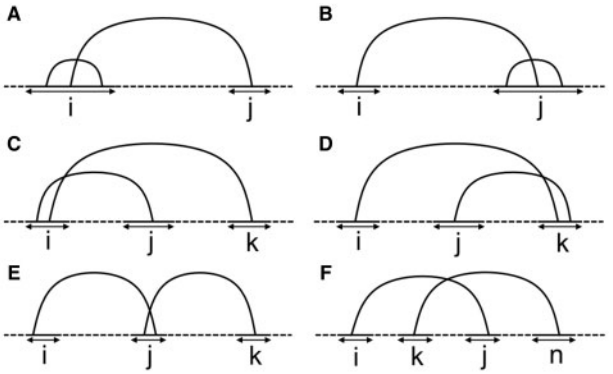


**Fig. 2.** Base pair crossings between non-nesting vertex partitions. **(A)** $V_q(i, i)$ and $V_q(i, j)$; **(B)** $V_q(i, j)$ and $V_q(j, j)$; **(C)** $V_q(i, j)$ and $V_q(i, k)$; **(D)** $V_q(i, k)$ and $V_q(j, k)$; **(E)** $V_q(i, j)$ and $V_q(j, k)$; **(F)** $V_q(i, j)$ and $V_q(k, n)$. $i$, $j$, $k$ and $n$ are backbone segments indicated by the arrows. The two partitions involved in each of these six scenarios are merged to a single partition

**Algorithm 3:** Expand partial mcECGMs from non-nesting stems

1 $M_{\text{all}} = \emptyset$
2 For each $m_{\text{nest}} \in M_{\text{nest}}$
3 Using the stem terminal regions of $m_{nest}$, divide the backbones of $R$ and $R'$ into non-overlapping segments $\{q_t \mid t = 1, 2, \ldots, 2|m_{\text{nest}}| + 1\}$ and $\{q'_t \mid t = 1, 2, \ldots, 2|m_{\text{nest}}| + 1\}$, respectively, with $t$ increasing from the 5′ end towards the 3′ end
4 Based on stem $p(v)$ for each $v \in V_{\text{non-nest}}$, add $v$ to $V_q(i, j)$ when the upstream and downstream regions of $p(v)$ are in segments $q_i$ and $q_j$ respectively ($i \leq j$)
5 Repeat line 4 to obtain all sets $V'_q$ for each $v' \in V'_{\text{non-nest}}$
6 Merge all $V_q(i, j)$ with $V_q(k, n)$ and $V'_q(i, j)$ with $V'_q(k, n)$ if there exists $u \in V_q(i, j)$ and $v \in V_q(k, n)$ such that $l(u, v) = =$ 'K', or there exists $u' \in V'_q(i, j)$ and $v' \in V'_q(k, n)$ such that $l'(u', v') = =$ 'K', for $i \leq j \leq k \leq n$
7 Merge $m_{\text{nest}}$ with all mcECGMs obtained from every non-empty $V_q(i, j) \times V'_q(i, j)$, and add all merged results to $M_{\text{all}}$
8 End for
9 Return $M_{\text{all}}$

### 2.2.5 Phase 5: Recovering missing vertex pairs in mcECGMs

In phase 3, a small $K$ value is used for efficiency reasons. However, it might happen that some nesting motif pairs appearing in the true mcECGMs are pruned. Moreover, although occasional, a nesting stem is matched to a non-nesting stem, but this is not considered in any previous phase. Given the mcECGMs $M_{\text{all}}$ from phase 4, this phase generates the candidate space $T_{\text{miss}}$ (line 3) and recovers all these missing vertex pairs from it (line 4). Because nesting stems are

involved, the ECGM edit operation costs applied in this phase are those defined in Section 2.1.

**Algorithm 4:** Recover missing vertex pairs in $M_{all}$

1 $M_{psm} = \emptyset$
2 For each $m_{all} \in M_{all}$
3 Denote $V_{all}$, $V'_{all}$ such that $m_{all} : V_{all} \rightarrow V'_{all}$. $T_{miss} = (V - V_{all}) \times (V' - V'_{all})$
4 Obtain all mcECGMs by expanding $m_{all}$ with $T_{miss}$, and add them to $M_{psm}$
5 End for
6 Return $M_{psm}$

### 2.3 Alignment function, non-bijective stem matching, and overall structure alignment generation

Motif alignment (i.e. $a(h(p(v)), h'(p'(v')))$ is used for vertex mapping between $V_{non\text{-}nest}$ and $V'_{non\text{-}nest}$ (phases 2 and 4), while stem alignment (i.e. $a(p(v), p'(v'))$) is used when the vertex mapping involves those representing nesting stems (phases 3 and 5). The difference is that the former aligns single segments from the two structures while the latter aligns a pair of stem upstream and downstream regions. For a nesting stem, its upstream and downstream regions are disjoint in the structure. Our alignment function $a$ is capable of performing these two types of alignment. Interested readers may refer to the supplementary material for details.

It is possible for a stem in a structure to be matched with more than one stem in another structure. For example, if the hydrogen bonds in several contiguous base pairs in the middle of a long helix are broken, the helix becomes two helices connected by an internal loop. Meanwhile, mcECGM is a bijective mapping and has to be extended to model such non-bijective stem matching. This process and the secondary structure alignment generation from the extended mcECGM are described in the supplementary material.

## 3 Results

### 3.1 Experimental setup

We implemented our approach in a tool called PSMAlign using Perl, and its alignment function is modified from rna_align (Jiang *et al.*, 2002) in C++. Two experiments were performed. The first experiment compared the performance of PSMAlign with other structure alignment tools in identifying similar structures. The second compared PSMAlign with the Needleman-Wunsch sequence alignment (Needleman and Wunsch, 1970) for aligning pseudoknotted structures. Pairwise alignment results involving pseudoknots in at least one of the input structures are also presented. The default parameter values for PSMAlign are as follows: For phase 2, $\mu = 5$, initial $b_{start} = -1$, initial $b_{end} = 0.1$, $\Delta = 0.05$ and $f = 2$. For phase 3, $K = 1$. Various costs used by its alignment function are: $w_b = 1.5$, $w_{bm} = 0.5$, $w_r = 2$, $w_a = 1.75$, $w_d = 1$ and $w_m = 1$. Both experiments were performed on an Intel Core2 Duo 3.0 GHz machine with 4GB of RAM running 64-bit Ubuntu 12.04.

### 3.2 Experiment 1: BRASERO benchmarks using SRP and 16S ribosomal RNA families

BRASERO (Allali *et al.*, 2012) evaluates the performance of various secondary structure alignment tools in identifying similar structures. Each of its testing datasets consists of a reference structure set $R_{ref}$ from an RNA family, and a sequence set $S$. $S$ contains a positive class of various sequences in the same family as but not in $R_{ref}$, and a negative class of some noise sequences with similar length distribution as $R_{ref}$. For each sequence $s$ in $S$, its optimal and suboptimal secondary structures are obtained using tools such as mfold (Markham and Zuker, 2005). Each of the predicted structures is aligned with every structure in $R_{ref}$ using the alignment tool examined. The best similarity score (or alignment cost depending on the tool) achieved is the score of $s$. All sequences in $S$ are then sorted by their scores. Afterwards, every sequence can be classified as positive (structurally similar to those in $R_{ref}$) or negative (structurally dissimilar) according to a threshold. From the classification result, the proportion of correctly classified sequences in the positive class is the true positive rate (TPR); and the proportion of incorrectly classified sequences in the negative class is the false positive rate (FPR). A receiver operating characteristic (ROC) curve for this alignment tool is a plot of TPR versus FPR obtained by varying the threshold, and the area under the ROC curve (AUC) illustrates the average performance of the tool (Fawcett, 2006). A higher AUC means it is more likely to give higher similarity scores or lower alignment costs to similar structures than to dissimilar structures, and hence a better performance. The maximum AUC is 1.

PSMAlign was benchmarked with Gardenia (Blin *et al.*, 2010), MiGaL (Allali and Sagot, 2008) and RNAForester (Hochsmann *et al.*, 2003, 2004), which are standalone secondary structure alignment tools publicly available. The signal recognition particle (SRP) and the 16S ribosomal RNA (rRNA) datasets were selected for benchmarks due to their significantly long sequence lengths (>300 nucleotides). Figure 3 shows that PSMAlign exhibited the highest AUC in both SRP and 16S rRNA datasets, which are 0.764 and 0.931, respectively. Moreover, at a very low FPR (<0.01), it achieved a significantly higher TPR of 0.6 in the SRP dataset and 0.675 in the 16S rRNA dataset. This suggests that 60% of the positive class sequences in the SRP dataset, and 67.5% of those in the 16S rRNA dataset, are structurally more similar to the reference structures than over 99% of the noise sequences. An interesting observation is the performance comparison between exact approach and heuristics in different datasets. Both Gardenia and RNAForester are exact approaches, while MiGaL and PSMAlign are heuristics assuming better conservation in structure than in sequence. They construct alignment according to the similar local motifs identified. In the 16S rRNA dataset, the ROCs of the heuristic approaches are well above that of the exact methods, meaning that the proposition holds. On the other hand, in the SRP dataset, the ROC of PSMAlign falls below the ROCs of the exact methods in its middle segment. A small segment is even under the diagonal (denoting the ideal ROC of random guess), meaning that at some thresholds, TPR < FPR. MiGaL performed even worse in that its ROC is well below the ROCs of the exact methods, and below the diagonal for a large segment. We believe that some positive class instances were more conserved in their sequences than in their structures. Heuristics performed worse than exact methods when the proposition did not hold for some molecules. This is supported by the even worse performance in MiGaL which also takes this proposition. The abstract properties, such as the number of helices or base pairs, utilized at its first three levels of structure comparison limited its precision in this dataset.

### 3.3 Experimental 2: Structural alignments involving pseudoknots

On the other hand, only PSMAlign is able to align pseudoknotted RNA secondary structures. Also, since the sequence folding tools in BRASERO do not predict pseudoknots, pseudoknotted structures

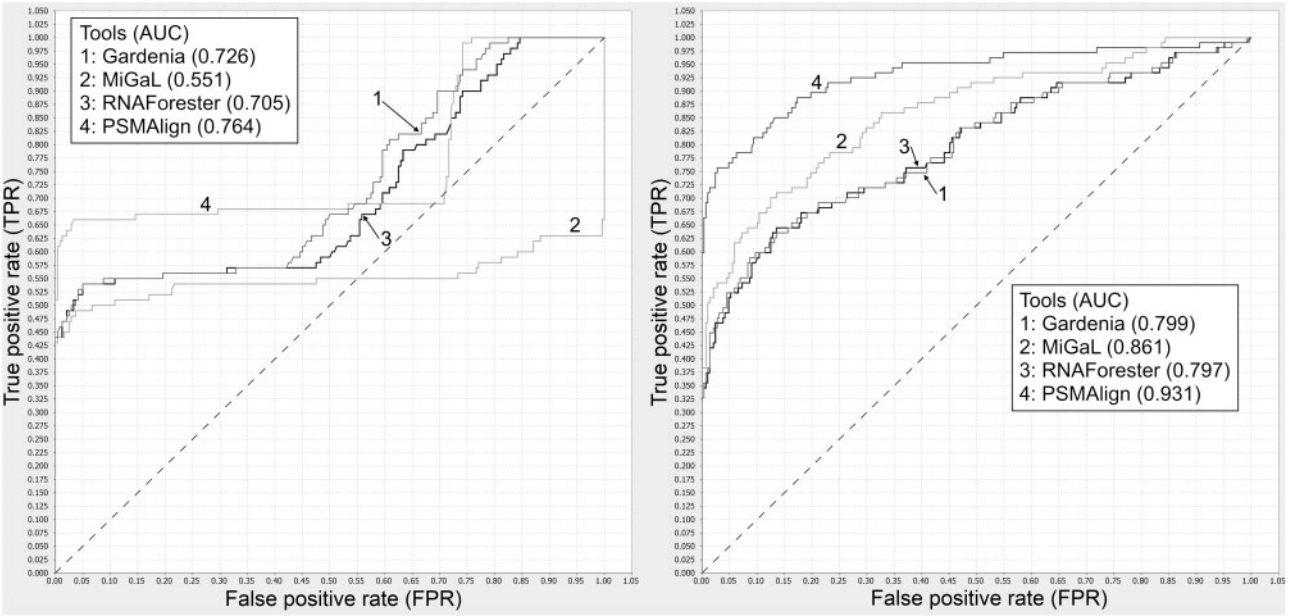


**Fig. 3.** ROC curves of BRASERO benchmark of Gardenia, MiGaL, RNAForester and PSMAlign on **Left**: SRP family; **Right**: 16S rRNA family

**Table 1.** Running time comparisons between PSMAlign and Needleman-Wunsch sequence alignment (in brackets) on selected 23S rRNA structures

| **Molecule Id** (length, unit: nucleotides, nts) | **PDB_00187** (2738) | **CRW_00520** (2902) | **CRW_00546** (3514) | **PDB_00993** (2802) | **CRW_00525** (2850) | **CRW_00521** (2877) | **CRW_00504** (3122) |
|---|---|---|---|---|---|---|---|
| **CRW_00471** (2903) | 30.12s (**15.94s**) | 20.56s (**16.64s**) | **12.54s** (20.15s) | 21.5s (**16.7s**) | 30.3s (**16.46s**) | 20.36s (**16.13s**) | 26.1s (**17.73s**) |
| **PDB_00187** (2738) | | 33.68s (**16.13s**) | **9.79s** (19.41s) | 27.65s (**15.34s**) | 29.92s (**15.48s**) | 34.67s (**16.35s**) | 35.16s (**17.73s**) |
| **CRW_00520** (2902) | | | 21.4s (**20.39s**) | 23.71s (**15.62s**) | 23.3s (**16.04s**) | 17.23s (**15.97s**) | 26.13s (**17.85s**) |
| **CRW_00546** (3514) | | | | **17.49s** (19.35s) | **11.85s** (20.12s) | **15.51s** (20.8s) | 171.8s (**21.52s**) |
| **PDB_00993** (2802) | | | | | 28.48s (**15.56s**) | 21.93s (**15.96s**) | 25.31s (**17.47s**) |
| **CRW_00525** (2850) | | | | | | 33.93s (**15.87s**) | 29.91s (**17.49s**) |
| **CRW_00521** (2877) | | | | | | | 26.65s (**17.84s**) |

The lower running time in each alignment is in bold. The organisms for the molecules are: *Acinetobacter calcoaceticus* (CRW_00471), *D.radiodurans* (PDB_00187), *C.reinhardtii* (CRW_00520), *Zea mays* (CRW_00546), *D.radiodurans* (PDB_00993), *Giardia intestinalis* (CRW_00525), *Epicrates gracilis* (CRW_00521), *Mycobacterium leprae* (CRW_00504).

were not examined in the first experiment. Consequently, we aligned several pairs of structures from other RNA families, and at least one structure of any pair contains pseudoknotted motifs. The structures were all obtained from the RNA STRAND database (Andronescu *et al.*, 2008), with non-canonical base pairs removed.

PSMAlign was also compared with the Needleman-Wunsch (NW) algorithm (Needleman and Wunsch, 1970) for aligning a set of structures randomly selected from the 23S rRNA family. This family has been known to exhibit a diverse range of pseudoknot topologies categorized previously (Chiu and Chen, 2012). A molecule was randomly selected from each topology to form the testing dataset. A knot-free 23S rRNA molecule was also included. Table 1 shows the running times of both approaches for pairwise alignments among the secondary structures in the testing dataset. The NW algorithm showed relatively steady running times as the algorithm takes $O(n^2)$ time for alignment and $O(n)$ time in backtracking a single alignment, where $n$ is the sequence length. In contrast, the time performance of PSMAlign mainly relied on the stem graph order and the mcECGM candidate pruning. In most cases, the running time of PSMAlign is about double that for the NW algorithm in this dataset, and it varies from 9.79s–171.8s. It performed faster than the NW algorithm when the pairwise alignment involved 23S rRNA from *Zea mays* (molecule Id: CRW_00546), because it has a very long unpaired region, resulting in a smaller graph order than other molecules. Because the selected secondary structures are quite conserved in their sequences, many matched stem pairs identified by PSMAlign could also be found by the NW algorithm. However, when a certain portion of paired bases in a stem have been substituted, they become hard to be matched by pure sequence alignment. Section E of the supplementary material discusses this issue with an example from the testing dataset.

Supplementary Figure S6 shows the approximate true conserved stem pattern in the ribonuclease P RNAs of *Mycoplasma genitalium* (molecule Id: ASE_00194) and *Streptococcus equi* (molecule Id: ASE_00320). Their structure alignment is depicted in Supplementary Figure S7A. Supplementary Figure S7B shows their alignment constructed from the best common stem pattern. An interesting observation is that the first alignment is likely to be a more reasonable alignment despite its higher alignment cost (281.25 versus 262). Even without the penalty for the unmapped stems, its alignment cost is still higher (201.75 versus 194.5). The alignment

of the nesting stems of the largest multi-loop in both structures are indicated by the black boxes in the figure. The highly conserved base sequences and base pairs strongly suggest that they are matched. This also shows that the best conserved stem pattern might not be the true conversed stem pattern. More importantly, exact structure alignment cannot reveal this conservation due to the non-minimum alignment cost. Supplementary Figure S8A–D in the supplementary material highlight the approximate true conserved stem pattern between the 23S ribosomal RNAs in *Chlamydomonas reinhardtii* (molecule Id: CRW_00520) and in *Deinococcus radiodurans* (molecule Id: PDB_00993). This pattern is very close to the true pattern since only a few stems are left unmatched. The overall alignment is shown in Supplementary Figure S9.

## 4 Discussion

Our progressive stem matching heuristics is similar to RNAStrAT (Guignon *et al.*, 2005), since both decompose the input structures into stems, and construct a conserved stem pattern based on the stem alignment results. The major difference between them is that our heuristics applies mcECGM for pattern finding, while RNAStrAT utilizes dynamic programming and therefore does not allow pseudoknots in the input structures. Moreover, it only compares the similarity between the nesting stems without considering the similarities among other stems nested in them. Progressive stem matching employs a structure profile to also compare the nested components as well as aligning the nesting stem themselves.

Although the experiment results suggest that true conserved stem pattern approximation works well for pairwise RNA secondary structure alignment, PSMAlign still retains the exact mcECGM algorithm finding the best conserved stem pattern. The reason is that there might not be enough highly similar non-nesting motif pairs ($<\mu$) at phase 2. Reducing the value of $\mu$ appears to be a possible solution, however the results become trivial when $\mu$ is set to just 1 or 2. When no result is found for the current $\mu$ value, it turns to discover the mcECGMs for the best conserved pattern, and users can also specify which stem pattern to use for alignment.

An interesting observation from the first experiment is that the average performance of PSMAlign in the SRP dataset is better than Gardenia and RNAForester, both of which are exact secondary structure alignment approaches. In the 16S rRNA dataset, both PSMAlign and MiGaL performed substantially better than the exact methods. The conclusion is that heuristics aligning RNA secondary structures by detecting similar structural motifs outperform exact approaches computing lowest cost global alignments. This is supported by our finding in experiment 2, in which the approximate true conserved stem pattern could result in a biologically more significant alignment despite its non-minimum alignment cost. This verifies that ncRNAs are, in general, more conserved in structures than in sequences. Because an alignment with the lowest edit distance or highest similarity score does not guarantee all important biological features to be aligned, it is therefore necessary to discuss the meaning of "optimal" secondary structure alignment, such as proposing new measures to quantify the pairwise similarity between the structures, or what types of base pair stacks or stems should be matched with a higher priority.

mcECGM is effective in approximating a true conserved stem pattern with arbitrary crossing stems. The pattern shown in Supplementary Figure S8A–D suggests that all the conserved crossing stems are matched correctly, no matter how complicated the underlying pseudoknots are. As the stem graph models the stem crossing relations with its edges, conserved crossing stems are able to be revealed by mcECGM. Meanwhile, it relies on a carefully designed candidate pruning scheme to reduce the computation time.

Local alignment is also very important because many RNA secondary structures are only conserved in some of their structural motifs. Progressive stem matching adopts a local then global style to first search for highly similar non-nesting motifs such as hairpins. These similar motif pairs are sorted according to their similarity measures ($r_h$ in phase 2), and those with better values are more probable to be retained in the reference information ($m_h$ in phase 2) for the next phase matching nesting stems. A pair of nesting stems is only shortlisted for alignment when both stems nest certain non-nesting motifs that are regarded as similar, according to the reference information. As a result, the more similar the local nesting and non-nesting motifs, the more likely they can be identified. The true conserved stem pattern is then approximated from these identified motif pairs. Supplementary Figure S10A–C shows the approximate true conserved stem pattern between the 23S ribosomal RNAs in *Caenorhabditis elegans* (molecule Id: CRW_00533) and *Suillus sinuspaulianus* (molecule Id: CRW_00544). Although both molecules differ significantly in length, three possible conserved local structural motifs (indicated by the red boxes) were identified from the pattern.

The major limitation of PSMAlign is the imperfect overall alignment due to separate alignments of the stem regions and the sequence regions. Precautionary measures were devised to detect some cases of cross-region alignments, but there are still some complicated situations, such as the one shown in the green box of Supplementary Figure S9. The vertical line illustrates the boundary of the aligned sequence region (left) and aligned the stem region (right). The 'GCG' segments of both structures should be aligned instead of aligning the single base pair, but the detection approach proposed is unable to detect this, because the gap is not an edge gap. This is caused by the base pair shifting from the right G nucleotide in the first structure to the left G nucleotide in the second structure, and we call it 'base pair shift'. We are currently working towards new measures to mitigate these limitations.

## 5 Conclusion

We have demonstrated the ability of the conserved stem pattern in performing pairwise alignment between RNA secondary structures. It serves as a heuristics to align structures with relatively higher accuracy. It enables biologists to perform computational structure alignment that could not be done previously, such as aligning those consisting of arbitrary pseudoknots. Sample alignments revealed some highly conserved structural features, which can be further investigated for a better understanding of their biological significance.

*Conflict of Interest*: none declared.

## References

Allali, J. and Sagot, M.-F. (2008) A multiple layer model to compare RNA secondary structures, *Softw. Pract. Exp.*, **38**, 775–792.

Allali, J. *et al* (2012) BRASERO: A Resource for Benchmarking RNA Secondary Structure Comparison Algorithms. *Advances in Bioinformatics*, Article ID 893048, doi:10.1155/2012/893048.

Andronescu,M. *et al.* (2008) RNA STRAND: The RNA secondary structure and statistical analysis database. *BMC Bioinformatics*, **9**, 340.

Blin,G. *et al.* (2010) Alignments of RNA structures. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, **7**, 309–322.

Bunke,H. (1997) On a relation between graph edit distance and maximum common subgraph. *Pattern Recogn. Lett.*, **18**, 689–694.

Cesana,M. *et al.* (2011) A long noncoding rna controls muscle differentiation by functioning as a competing endogenous RNA. *Cell*, **147**, 358–369.

Chen,Q. and Chen,Y.P.P. (2009) Discovery of structural and functional features in rna pseudoknots. *IEEE Trans. Knowl. Data Eng.*, **21**, 974–984.

Chiu,J.K.H. and Chen,Y.P.P. (2012) Conformational features of topologically classified RNA secondary structures. *PLoS One*, **7**, e39907.

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

Darty,K. *et al.* (2009) VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics*, **25**, 1974–1975.

Denise,A. and Rinaudo,P. (2014) Optimisation problems for pairwise RNA sequence and structure comparison: a brief survey. *Trans. Comput. Intell. XIII*, **8342**, 70–82.

Evans,P.A. (2006) Finding common RNA pseudoknot structures in polynomial time. *Comb. Pattern Match. Lect. Notes Comput. Sci.*, **4009**, 223–232.

Fawcett,T. (2006) An introduction to ROC analysis. *Pattern Recogn. Lett.*, **27**, 861–874.

Guignon,V. *et al.* (2005) An edit distance between RNA stem-loops. String Processing and Information Retrieval. *Lect. Notes Comput. Sci.*, **3772**, 335–347.

Hamada,M. *et al.* (2006) Mining frequent stem patterns from unaligned RNA sequences. *Bioinformatics*, **22**, 2480–2487.

Herrbach,C. *et al.* (2010) Average complexity of the Jiang–Wang–Zhang pairwise tree alignment algorithm and of a RNA secondary structure alignment algorithm. *Theor. Comput. Sci.*, **411**, 2423–2432.

Hochsmann,M. *et al.* (2003) Local similarity in RNA secondary structures. In: *Proceedings of the 2003 IEEE Computer Society Bioinformatics Conference*, pp. 159–168.

Hochsmann,M. *et al.* (2004) Pure multiple RNA secondary structure alignments: a progressive profile approach. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, **1**, 53–62.

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. *Theor. Comput. Sci.*, **143**, 137–148.

Lee,K. *et al.* (1997) In vivo determination of RNA structure-function relationships: analysis of the 790 loop in ribosomal RNA. *J. Mol. Biol.*, **269**, 732–743.

Lin,G.-H. *et al.* (2001) Edit distance between two RNA structures. *Proceedings of the Fifth Annual International Conference on Computational Biology*. ACM, Montreal, Quebec, Canada, pp. 211–220.

Maden,B.E. and Hughes,J.X. (1997) Eukaryotic ribosomal RNA: the recent excitement in the nucleotide modification problem. *Chromosoma*, **105**, 391–400.

Markham,N.R. and Zuker,M. (2005) DINAMelt web server for nucleic acid melting prediction. *Nucleic Acids Res.*, **33**, W577–W581.

Möhl,M. *et al.* (2008) Fixed parameter tractable alignment of RNA structures including arbitrary pseudoknots. *Comb. Pattern Match. Lect. Notes Comput. Sci.*, **5029**, 69–81.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, **48**, 443–453.

Reiter,N.J. *et al.* (2011) Emerging structural themes in large RNA molecules. *Curr. Opin. Struct. Biol.*, **21**, 319–326.

Rivas,E. and Eddy,S.R. (1999) A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J. Mol. Biol.*, **285**, 2053–2068.

Rødland,E.A. (2006) Pseudoknots in RNA secondary structures: representation, enumeration, and prevalence. *J. Comput. Biol.*, **13**, 1197–1213.

Scott,L. and Hennig,M. (2008) RNA structure determination by NMR. In: Keith,J. (ed.) *Bioinformatics*. Humana Press, New York, pp. 29–61.

Serganov,A. and Patel,D.J. (2007) Ribozymes, riboswitches and beyond: regulation of gene expression without proteins. *Nat. Rev. Genet.*, **8**, 776–790.

Staple,D.W. and Butcher,S.E. (2005) Pseudoknots: RNA structures with diverse functions. *PLoS Biol.*, **3**, e213.

Storz,G. (2002) An expanding universe of noncoding RNAs. *Science*, **296**, 1260–1263.

Theimer,C.A. *et al.* (2005) Structure of the human telomerase RNA pseudoknot reveals conserved tertiary interactions essential for function. *Mol. Cell*, **17**, 671–682.

Turner,D.H. and Mathews,D.H. (2010) NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res.*, **38**, D280–D282.

Wadkins,T.S. *et al.* (1999) A nested double pseudoknot is required for self-cleavage activity of both the genomic and antigenomic hepatitis delta virus ribozymes. *RNA*, **5**, 720–727.

Zhang,K. and Shasha,D. (1989) Simple fast algorithms for the editing distance between trees and related problems. *SIAM J. Comput.*, **18**, 1245–1262.

Zuker,M. and Stiegler,P. (1981) Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.*, **9**, 133–148.