Lightweight Comparison of RNAs Based on Exact Sequence-Structure Matches

Steffen Heyne, Sebastian Will, Michael Beckstette, Rolf Backofen

{heyne,will,mbeckste, backofen}@informatik.uni-freiburg.de

Albert-Ludwigs-University Freiburg,
Institute of Computer Science, Chair of Bioinformatics,
Georges-Koehler-Allee 106,
79110 Freiburg, Germany

Abstract: Specific functions of RNA molecules are often associated with different motifs in the RNA structure. The key feature that forms such an RNA motif is the combination of sequence and structure properties. In this paper we introduce a new RNA sequence-structure comparison method which maintains exact matching substructures. Existing common substructures are treated as whole unit while variability is allowed between such structural motifs.

Based on a fast detectable set of overlapping and crossing substructure matches for two nested RNA secondary structures, our method computes the longest colinear sequence of substructures common to two RNAs in $O(n^2m^2)$ time and $O(nm)$ space. Applied to different RNAs, our method correctly identifies sequence-structure similarities between two RNAs. The results of our experiments are in good agreement with existing alignment-based methods, but can be obtained in a fraction of running time, in particular for larger RNAs. The proposed algorithm is implemented in the program `expaRNA`, which is available from our website (www.bioinf.uni-freiburg.de/Software).

1 Introduction

Ribonucleic acids (RNAs) are associated to a large range of important cellular functions in living organisms. Moreover, recent findings show that RNAs can perform regulatory functions formerly assigned to proteins only. Likewise to proteins, these functions are often associated with evolutionary conserved motifs that contain specific sequence and structure properties. Examples for such regulatory RNA elements, whose function is mediated by sequence-structure motifs are selenocysteine insertion sequence (SECIS) elements [HWB96] (see Figure 1 for an example), iron-responsive elements (IRE)[HK96], different riboswitches [SP07], or internal ribosomal entry sites (IRES)[MLBM$^+$04]. Therefore, the detection of similar structural motifs in different RNAs is an important aspect for function determination and should be considered in pairwise RNA comparison methods. Although this problem is addressed in sequence-structure alignment methods, these approaches are often very time-consuming and do not necessarily preserve functionally important common substructures in the alignment [JLMZ02, JWZ95].

In this paper we propose a new lightweight, motif-based method for the pairwise comparison of RNAs. Instead of computing a full sequence-structure alignment, our approach efficiently computes a significant arrangement of sequence-structure motifs, common to two RNAs. For the

sake of algorithmic complexity and applicability in practice, we neglect higher order interactions like pseudoknots. This allows to describe sequence-structure motifs with nested RNA secondary structures, as shown in Figure 1.

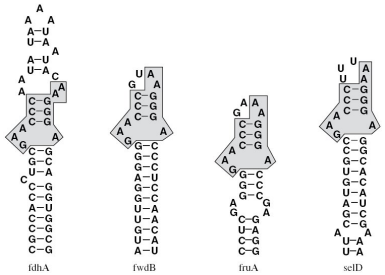

Figure 1: Putative SECIS elements in non-coding regions of *Methanococcus jannaschii* according to [WSPB97]. The indicated substructure represents a common substructure.

In [BS07] the authors presented a fast $O(nm)$ time and space algorithm for the identification of isolated common substructures for two given RNAs of lengths n and m with nested secondary structures. More precisely, their method identifies the complete, but overlapping set of exact common substructures. Our approach makes use of these common substructures and computes the longest colinear, non-overlapping sequence of substructures common to two RNAs in $O(n^2m^2)$ time and $O(nm)$ space. Herein after, we call this the LONGEST COMMON SUBSEQUENCE OF EXACT PATTERN MATCHINGS problem (LCS-EPM).

Related Work

Existing approaches addressing the sequence-structure comparison problem for RNA molecules can be distinguished by the given structural information and their representation. The standard alignment-based comparison approach employs the computation of edit distances between given RNA secondary structures [BMR95, JLMZ02]. In [Eva99] the author introduced the problem of finding the longest arc-preserving common subsequence (LAPCS). However, even for two *nested* RNA secondary structures, both problems remain NP-hard [BFRS03, LCJW02]. With some restrictions to the scoring scheme, the time complexity for determination of the edit distance can be lowered to polynomial time [JLMZ02].

If the nested secondary structure is represented as a tree, comparison methods exist for the edit distance between two ordered labeled trees [ZS89] as well as for the alignment of trees [JWZ95]. An improved version of the tree alignment method with extension to global and local forest alignments is given in [HTGK03] and implemented in the program RNAforester. The MIGAL approach extends the tree edit distance model by two new tree edit operations and is especially efficient due to its usage of different abstraction layers [AS05].

2 Exact Pattern Matchings and Longest Common Subsequences of Two RNA Secondary Structures

RNA is a macro molecule described formally by a pair $\mathcal{R} = (S, B)$ of a primary structure S and a secondary structure B. A *primary structure* S is a sequence of nucleotides $S = s_1 s_2 \dots s_n$

over the alphabet $\{A, C, G, U\}$. With $|S|$ we denote the length of sequence S. $S[i]$ indicates the nucleotide at position i in sequence S. With $S[i...j]$ we define the substring of S starting at position i until j for $1 \leq i < j \leq |S|$. A *secondary structure* B is a set of base pairs $B = \{(i, i') \mid 1 \leq i < i' \leq |S|\}$ over S, where each base takes part in at most one base pair. A secondary structure B is called *crossing* if there are two pairs $(i, i'), (j, j') \in B$ with $i < j < i' < j'$. Otherwise it is called *non-crossing* or *nested*.

For the definition of local RNA motifs, we represent an RNA $\mathcal{R} = (S, B)$ as undirected labeled graph $G = (V, E)$, called the *structure graph of* $\mathcal{R}$. Its set of vertices V is the set of positions in S, i.e. $V = \{1, \ldots, |S|\}$. Its set of edges E comprises all backbone bonds and all base pairs, i.e. $E = \{(i, i+1) \mid 1 \leq i < |S|\} \cup B$. An *RNA pattern in* $\mathcal{R}$ is a set of positions $\mathcal{P} \subseteq \{1, \ldots, |S|\}$, such that the *pattern graph for* $\mathcal{P}$ *in* G, defined as the subgraph $G' = (V', E')$ of G, where $V' = \mathcal{P}$ and $E' = \{(i, i') \in E \mid i \in \mathcal{P} \text{ and } i' \in \mathcal{P}\}$, is connected. By this definition, an RNA pattern corresponds to a local motif, i.e. a substructure that preserves the local neighborhood induced by backbone bonds and base pairs within a fixed secondary structure.

2.1 Exact Pattern Matchings of Two RNAs

In the following we consider two fixed, non-crossing RNAs $\mathcal{R}_1 = (S_1, B_1)$ and $\mathcal{R}_2 = (S_2, B_2)$. Their corresponding structure graphs are $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, respectively. We will define an exact pattern matching as an *ordered matching of* V_1 *and* V_2, i.e. as a set $\mathcal{M} \subseteq V_1 \times V_2$, where for all $(p, q), (p', q') \in \mathcal{M}$ it holds that $p < p'$ implies $q < q'$ and $p = p'$ iff $q = q'$.

According to an ordered matching $\mathcal{M}$ of V_1 and V_2, we merge the graphs G_1 and G_2 into a *matching graph* $G_{\mathcal{M}} = (\mathcal{M}, E_{\mathcal{M}})$, where $E_{\mathcal{M}} = \{((p, q), (p', q')) \in \mathcal{M} \times \mathcal{M} \mid (p, p') \in E_1 \text{ and } (q, q') \in E_2\}$. A pair $(p, q) \in \mathcal{M}$ is called *admissible* if it satisfies the following conditions: (a) $S_1[p] = S_2[q]$ and (b) $\mathsf{STRUCT}_1(p) = \mathsf{STRUCT}_2(q)$. Here, function $\mathsf{STRUCT}_i(j)$ yields one of the three possible structural types for a nucleotide at position j in structure i: *single stranded*, *left paired*, or *right paired*. Further we want to preserve base pairs, i.e. $\forall (p, q), (p'q') \in \mathcal{PM} : (p, p') \in B_1 \Leftrightarrow (q, q') \in B_2$. Then, an *exact pattern matching* $\mathcal{PM}$ is an ordered matching where $G_{\mathcal{PM}}$ is connected, all $(p, q) \in \mathcal{PM}$ are admissible and all base pairs are preserved.

Hence, an exact pattern matching $\mathcal{PM}$ describes the matching between sets of positions in the two RNAs $\mathcal{R}_1$ and $\mathcal{R}_2$, namely the projections $\pi_1\mathcal{PM} = \{p | (p, q) \in \mathcal{PM}\}$ and $\pi_2\mathcal{PM} = \{q | (p, q) \in \mathcal{PM}\}$. Note that $\pi_1\mathcal{PM}$ and $\pi_2\mathcal{PM}$ are patterns in $\mathcal{R}_1$ and $\mathcal{R}_2$ respectively, i.e. in particular they correspond to the connected pattern graphs G_1^p and G_2^p. Note, although we claim an isomorphism on base pairs, $\mathcal{PM}$ does not necessarily describe an isomorphism on backbone edges in the pattern graphs G_1^p and G_2^p, since for $(p, q), (p', q') \in \mathcal{PM}$ where p and p' form an edge in G_1^p, q and q' do not necessarily form an edge in G_2^p. For details and proofs we refer to [BS07].

For our algorithm, we utilize only *maximal* exact pattern matchings, i.e. $\forall \mathcal{PM}' : \mathcal{PM} \subseteq \mathcal{PM}' \Rightarrow \mathcal{PM}' = \mathcal{PM}$. We abbreviate the term exact matching pattern by EPM. In the following, EPMs are always maximal. Similar to the minimal word size as e.g. used in BLAST [AMS+97], it is reasonable to consider a minimal size γ for EPMs. Hence, the set of all maximal exact pattern matchings $\mathcal{E}$ over two RNAs $\mathcal{R}_1$ and $\mathcal{R}_2$ is defined as

$$\mathbf{E}_{\gamma}^{1,2} = \{\, \mathcal{E} \mid \mathcal{E} \text{ is EPM } \wedge\ |\mathcal{E}| \geq \gamma \,\}.$$

Note that each EPM is an arc-preserving common (but not longest common) subsequence as defined in [Eva99] for the LAPCS problem. However, the set of all EPMs is not a solution for the LAPCS problem since the combination of several EPMs is not necessarily arc-preserving. Since EPMs have in addition the above described properties, the detection of all EPMs is a computationally easy problem, compared to LAPCS, which is NP-complete even for nested sequences [BFRS03]. Using the dynamic programming approach described in [BS07], the set of all EPMs can be found in $O(nm)$ time and $O(nm)$ space, making this approach applicable for fast sequence-structure comparisons.

Now recall that each EPM is maximal. This implies that any two exact pattern matchings are disjoint and therefore a pair $(p, q) \in \mathcal{E} \in \mathbf{E}_\gamma^{1,2}$ is unique in $\mathbf{E}_\gamma^{1,2}$ and part of at most one EPM. Of course, two EPMs can overlap in one RNA and even in both RNAs. But this overlapping case implies that one exact pattern matching has to match to another region in the other RNA. The number of EPMs contained in $\mathbf{E}_\gamma^{1,2}$ is bounded by $n \cdot m$, with $n = |S_1|$ and $m = |S_2|$.

$\mathbf{E}_\gamma^{1,2}$ can be seen as a "library" of all common motifs between two RNAs, that can be utilized for a pairwise comparison method. In the following we describe the main aspects of our method based on common substructures. The EPMs in $\mathbf{E}_\gamma^{1,2}$ differ in their size and shape as well as in their structural positions in both RNAs. Taking two or several of these substructures into account they probably overlap or cross each other (see Figure 2). Clearly, a meaningful subset of common substructures excludes overlapping and crossing patterns. This guarantees that the backbone order of matched nucleotides as well as base pairs of the given RNAs are preserved. Compatible EPMs are called *non-crossing*.

Figure 2 shows an example of a possible set $\mathbf{E}_\gamma^{1,2}$. A "good" subset to describe the similarity between the two RNAs would probably exclude the EPMs indicated in red.

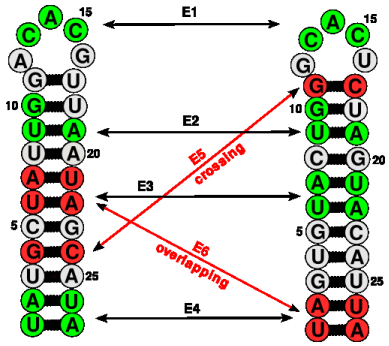

Figure 2: A possible set $\mathbf{E}_\gamma^{1,2}$ for two RNAs $\mathcal{R}_1, \mathcal{R}_2$. The set $\{\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4\}$ can be used for a comparison, whereas $\{\mathcal{E}_5, \mathcal{E}_6\}$ should be excluded. $\mathcal{E}_5$ is crossing $\mathcal{E}_2$ and $\mathcal{E}_3$ whereas $\mathcal{E}_6$ is overlapping with $\mathcal{E}_3$ in $\mathcal{R}_1$ and with $\mathcal{E}_4$ in $\mathcal{R}_2$. Note, that not all possible EPMs are indicated.

2.2 A Global Comparison Approach: The Longest Common Subsequence of Exact Pattern Matchings (LCS-EPM)

The formulation of LCS-EPM is motivated by the fact that different RNA secondary structures share similar structural elements. Examples are shown in our result section for the comparison of thermodynamically folded as well as experimentally verified secondary structures. The knowl-

edge of such a "common core" of identical substructures in two RNAs is interesting for different tasks.

For our global approach we are interested in a *maximal* possible arrangement of substructures shared by two RNAs. If the motives are given in the form of exact pattern matchings, we call this the LCS-EPM problem. Basically, we search for a maximal combination of EPMs that form a common subsequence. Note that albeit the problem shares some similarity with LAPCS, it is restricted in such a way that an efficient solution is possible.

Formally, LCS-EPM is defined as follows. Given two nested RNAs $\mathcal{R}_1$, $\mathcal{R}_2$ and a set of exact pattern matchings $\mathbf{E}_\gamma^{1,2}$over these two RNAs, LCS-EPM is the problem of finding the longest common subsequence of S_1 and S_2 which preserves the exact pattern matchings in $\mathbf{E}_\gamma^{1,2}$; i.e. finding a mapping $\mathcal{M}_{\mathsf{EPM}} \subseteq V_1 \times V_2$ of maximal length such that:

1. for each pair $(p,q) \in \mathcal{M}_{\mathsf{EPM}}$ there exists one EPM in $\mathbf{E}_\gamma^{1,2}$:
   $\forall (p,q) \in \mathcal{M}_{\mathsf{EPM}} : \exists \mathcal{E} \in \mathbf{E}_\gamma^{1,2}$with $(p,q) \in \mathcal{E}$ and $\mathcal{E} \subseteq \mathcal{M}_{\mathsf{EPM}}$
2. $\mathcal{M}_{\mathsf{EPM}}$ is a bijective mapping and preserves the order of the nucleotides:
   $\forall (p,q),(p',q') \in \mathcal{M}_{\mathsf{EPM}} : p = p' \iff q = q', p < p' \iff q < q'$

Condition one claims that for any matched nucleotide, there exists one EPM in $\mathbf{E}_\gamma^{1,2}$. In addition, condition one includes that the complete EPM is part of $\mathcal{M}_{\mathsf{EPM}}$. The second condition ensures that the found subsequence is a common subsequence, i.e. a sequence which preserves the backbone order. Arcs or base pairs are induced by the EPMs itself.

2.2.1 Boundaries and Holes

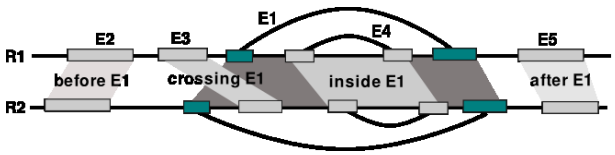

Figure 3: Ordering of exact pattern matchings relative to EPM $\mathcal{E}_1$ (indicated in green and dark gray). The cases *before, inside* and *after* do not violate the non-crossing condition. Only EPM $\mathcal{E}_3$ crosses $\mathcal{E}_1$. Note that an arc denotes a base pair within an EPM.

Our algorithm works by combining compatible EPMs. Given a single EPM of a library of EPMs, the relative order of the other EPMs can be distinguished as given in Figure 3. Formally, this is defined via the bounds and holes of a single EPM.

Bounds of EPMs The nucleotide positions of a pattern $\mathcal{P}$ of size k can be written as an increasing sequence. Similarly, an EPM $\mathcal{E}$ of size k over two RNAs is given with its corresponding patterns $\mathcal{P}_1$ in $\mathcal{R}_1$ and $\mathcal{P}_2$ in $\mathcal{R}_2$ and their increasing sequences $\mathcal{P}_1 = \langle p_1, p_2, ..., p_k \rangle$ and $\mathcal{P}_2 = \langle q_1, q_2, ..., q_k \rangle$.

In the view of the secondary structure, the elements (p_1, p_k) and (q_1, q_k) determine the outside borders of the EPM. Therefore we call them *outside-bounds* and write them as $\mathsf{OUT}_\mathcal{E} = \big\langle (p_1, p_k), (q_1, q_k) \big\rangle$. In the view of an arc-annotated sequence, we call (p_1, q_1) *left-outside-bounds* and (p_k, q_k) *right-outside-bounds* and denote them as $\mathsf{LEFT}_\mathcal{E}$ and $\mathsf{RIGHT}_\mathcal{E}$.

If an EPM contains base pairs, the structural shape is more complex and the outside-bounds are not sufficient to describe all structural borders. If not all enclosed nucleotides of a base pair are part of the EPM, then there exist two positions in each RNA that form an additional structural border *inside* the range of the outside-bounds. In addition, if a pattern contains several independent base pairs (e.g. in a multi-loop), there can be several such inside borders. The set of all such borders is called *inside-bounds* and is defined as $\mathsf{IN}_{\mathcal{E}} = \{\langle (p_i, p_{i+1}), (q_j, q_{j+1}) \rangle \mid p_{i+1} > p_i + 1 \Leftrightarrow q_{j+1} > q_j + 1\}$. Note, that *outside-bounds* always exists, whereas the set *inside-bounds* can be empty. For example, suppose an EPM that comprises only unbound nucleotides or a complete hairpin inclusive the closing bond. If an EPM consists of only one base pair in each sequence, then inside and outside bounds are identical. With the superscript index for the RNA we retrieve the bounds for a single RNA. For example $\mathsf{LEFT}^1_{\mathcal{E}} = p_1$.

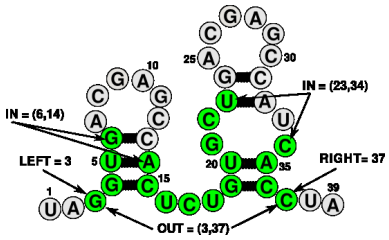

Figure 4: A pattern of an EPM in one RNA (green nucleotides). The different bounds are indicated.

Holes Holes are directly related to inside-bounds and describe the subsequences which are not part of the subsequence $S_i[\mathsf{LEFT}^i, \mathsf{RIGHT}^i]$ of an EPM. For a given EPM $\mathcal{E}$ with its set of inside-bounds $\mathsf{IN}_{\mathcal{E}}$, the set of holes with minimal size γ is defined as $\mathsf{HOLES}_{\mathcal{E}} = \{\ \langle (l^1, r^1), (l^2, r^2) \rangle \mid r^1 \geq l^1 + \gamma \ \wedge \ r^2 \geq l^2 + \gamma\ \}$. For each $h \in \mathsf{HOLES}_{\mathcal{E}}$ there exists a pair of inside-bounds with $\langle (l^1 - 1, r^1 + 1), (l^2 - 1, r^2 + 1) \rangle \in \mathsf{IN}_{\mathcal{E}}$. Clearly, a hole defines a substring $S_1[l^1, r^1]$ in the first RNA and a substring $S_2[l^2, r^2]$ in the second RNA. With γ we refer to the same size as indicated by $\mathbf{E}^{1,2}_{\gamma}$.

According to the length of the induced subsequences $S_i[l^i, r^i]$, we can sort all holes in one RNA. Let $h_i \in \mathsf{HOLES}_{\mathcal{E}_i}$ and $h_j \in \mathsf{HOLES}_{\mathcal{E}_j}$ two holes for any two $\mathcal{E}_i, \mathcal{E}_j \in \mathbf{E}^{1,2}_{\gamma}$. We define a partial ordering $h_i \preceq_{\mathsf{HOLES}} h_j$ in $\mathcal{R}_1$ if and only if h_i is of smaller size than h_j or of equal size in $\mathcal{R}_1$, i.e. $h_i \preceq_{\mathsf{HOLES}} h_j \Longleftrightarrow (r^1_i - l^1_i) \leq (r^1_j - l^1_j)$

2.2.2 Algorithm to Solve LCS-EPM

The crucial point and the main difference to alignment-based approaches as well as the LAPCS problem is that we have to treat a common substructure as whole unit. Therefore the final mapping has to include all pairs (p, q) of an EPM. Moreover, we want to compute the longest colinear sequence of EPMs which does not contain any crossing and overlapping EPMs.

The overall solution for LCS-EPM is constructed with a bottom-up approach from the comparison of substructures. This in principle requires a four-dimensional matrix, denoted as $D(i, j, k, l)$. Here the indices i, j refer to a substring $S_1[i, j]$ and the indices k, l to a substring $S_2[k, l]$, respectively. However, we can restrict ourselves to two-dimensional matrices using our notions of bounds and holes for an exact pattern matching $\mathcal{E}$ (see below). Finding non-crossing regions

relative to an EPM is achieved as follows: all nucleotides before $\mathsf{LEFT}_{\mathcal{E}}$, i.e $S_i[1, \mathsf{LEFT}^i_{\mathcal{E}} - 1]$, as well as all nucleotides after the $\mathsf{RIGHT}_{\mathcal{E}}$, i.e. $S_i[\mathsf{RIGHT}^i_{\mathcal{E}} + 1, |S_i|]$ fulfill the non-crossing condition. This means that any EPM with its outside-bounds $\mathsf{OUT}_{\mathcal{E}}$ in these regions is non-crossing relative to the considered EPM. Similar we handle EPMs that contain base pairs with the introduced notion of $\mathsf{HOLES}_{\mathcal{E}}$. All nucleotides inside these bounds are non-crossing, i.e. all EPMs which have outside-bounds within these regions satisfy the inside condition for non-crossing.

The recursion scheme for a dynamic programming algorithm is as follows. Any $\mathcal{E}$ is handled only once at its right-outside-bound $\mathsf{RIGHT}_{\mathcal{E}}$. The score of $\mathcal{E}$ is composed of the score *before* $\mathcal{E}$, given at the position $\mathsf{LEFT}_{\mathcal{E}} - 1$, plus the size of $\mathcal{E}$ itself, denoted by the function ω, plus possible scores of inside-bounds, given recursively by the computation of $\mathsf{HOLES}_{\mathcal{E}}$. This last recursion describes possible substructures and would lead to filling up a four-dimensional matrix. An improvement is achieved by ordering all holes according to the above introduced partial ordering $\preceq_{\mathsf{HOLES}}$. The recursion starts with one of the smallest holes and the remaining holes are computed in the order induced by $\preceq_{\mathsf{HOLES}}$. Hence, all necessary matrix entries exist, if an EPM with a hole is considered. Thus, only a two-dimensional matrix is necessary which leads directly to a quadratic space complexity. If two holes are of the same size, they can be treated in any order.

Suppose a given hole $h = \langle (l^1, r^1), (l^2, r^2) \rangle$, the following recursion scheme works for any $l^1 \leq j \leq r^1$ and $l^2 \leq l \leq r^2$. The best score is computed from treating the whole sequence as hole. With a standard traceback technique the set of EPMs that form the LCS-EPM are found.

$$\mathbf{D}(j,l) = \max \begin{cases} \mathbf{D}(j-1,l) \\ \mathbf{D}(j,l-1) \\ \mathbf{D}(i-1,k-1) + \mathbf{S}_{\mathcal{E}}, \\ \quad \text{if } \exists \mathcal{E} \in \mathbf{E}^{1,2}_{\gamma} \text{ with } \mathsf{RIGHT}_{\mathcal{E}} = (j,l) \text{ and} \\ \qquad \mathsf{LEFT}_{\mathcal{E}} = (i,k), i \geq l^1, k \geq l^2 \end{cases}$$

$$\mathbf{S}_{\mathcal{E}} = \omega(\mathcal{E}) \; + \sum_{h \in \mathsf{HOLES}_{\mathcal{E}}} \mathbf{D}(r^1, r^2) \quad \text{with } h = \langle (l^1, r^1), (l^2, r^2) \rangle$$

Complexity: The lengths of the sequences are $|S_1| = n$, $|S_2| = m$. The time complexity depends primarily on the number of holes. The set $\mathbf{E}^{1,2}_{\gamma}$ contains maximal $n \cdot m$ different holes which is estimated with $O(nm)$. The proof is omitted. For each hole, we fill a two-dimensional matrix with a size of at most $|S_1[l^1, r^1]| \leq |S_1| = n$ and $|S_2[l^2, r^2]| \leq |S_2| = m$. Consequently, for all holes we need $O(n^2 m^2)$ time as worst case complexity. For real RNAs, a more appropriate time complexity can be given as $O(H \cdot nm)$ with H as the number of holes, since $H \ll n \cdot m$. This also explains the fast running times of our examples. The space complexity is estimated with $O(nm)$ because the score of each hole is added to its EPM and the filled matrix is then discarded.

We summarize the complexity of solving the LCS-EPM problem as follows. Given two nested RNAs $\mathcal{R}_1 = (S_1, B_1)$ and $\mathcal{R}_2 = (S_2, B_2)$. The problem to determine the longest common subsequence of exact pattern matchings (LCS-EPM), including computation of $\mathbf{E}^{1,2}_{\gamma}$, is solvable in total $O(n^2\, m^2)$ time and $O(nm)$ space.

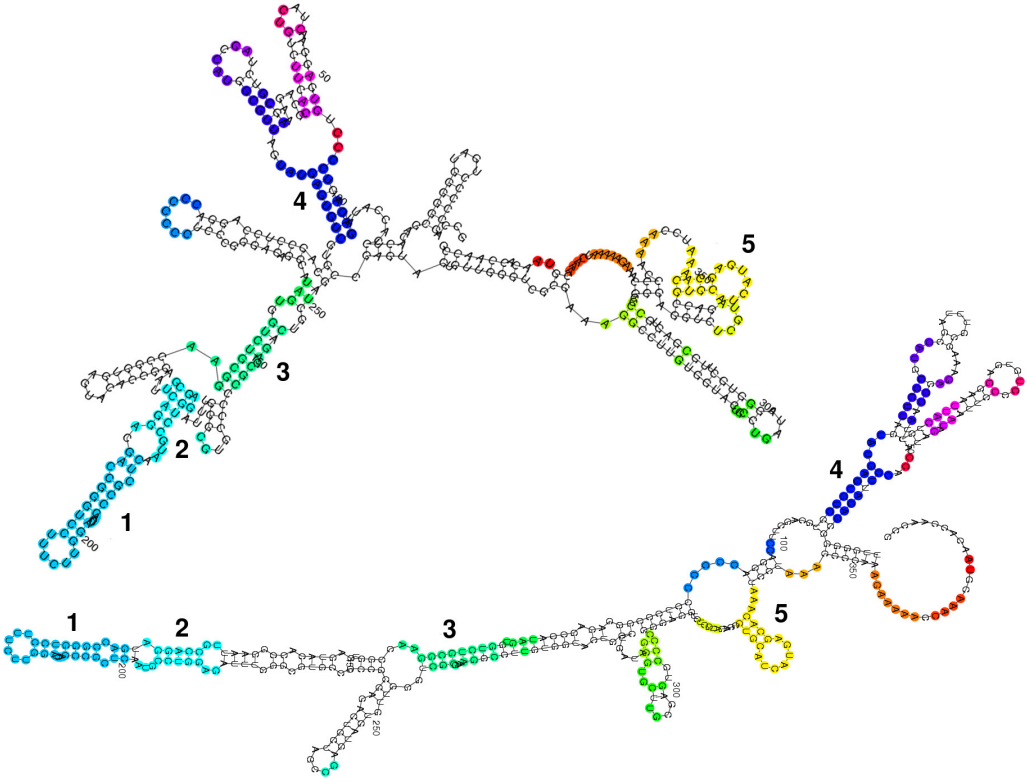

Figure 5: LCS-EPM approach applied to two Hepatitis C virus IRES RNAs. The colored nucleotides represent the found LCS-EPM with a length of 175 bases. Each EPM is shown in a different color. The numbers indicate the five largest EPMs from $\mathbf{E}_{\gamma}^{1,2}$. GenBank: D45172 (upper RNA), AF165050 (lower RNA)

3 Results

We implemented the algorithm for finding the longest common subsequence of exact RNA patterns in the tool **expaRNA** (exact pattern alignment of RNA). The algorithm to determine all EPMs was obtained from [BS07]. In order to analyze the performance of our approach, we have chosen two pairs of RNAs: **a)** Two IRES RNAs from Hepatitis C virus, which belong both to the Rfam family HCV_IRES for internal ribosomal entry sites (IRES) [GJMM$^+$05]. GenBank: AF165050 (bases 1-379) and D45172 (bases 1-391). The secondary structures were found via `RNAfold` [HFS$^+$94]. **b)** Two 16S rRNAs. The first RNA is from *Escherichia coli* and is 1541 bases long. The second RNA is from *Dictyostelium discoideum* and is 1551 bases long (GenBank codes: J01859 and D16466). The secondary structures were taken from the Comparative RNA Web (CRW) site [CSS$^+$02].

Table 1 shows the results for both pairs of RNAs. The structures with the indicated LCS-EPM can be seen in Figure 5 for the IRES RNAs and in Figure 6 for the 16S rRNAs. These figures are produced from `expaRNA` by interacting with the Vienna RNA Package [HFS$^+$94]. For the IRES RNAs, the numbers mark the five largest EPMs from the set $\mathbf{E}_{\gamma}^{1,2}$ and refer to the

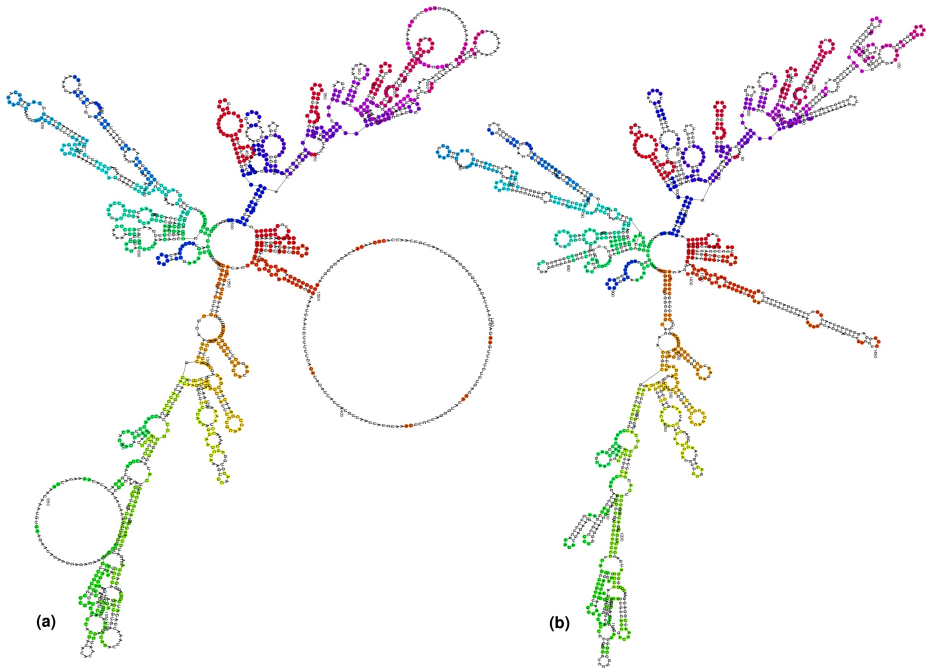

Figure 6: LCS-EPM approach applied two 16S ribosomal RNAs. The colored nucleotides represent the found LCS-EPM with a length of 875 bases. Each EPM is shown in a different color. (a) *D. discoideum* 16S rRNA (D16466), (b) *E. coli* 16S rRNA (J01859)

manually marked EPMs from the paper [BS07]. Our solution for LCS-EPM includes all of them automatically. An interesting detail is, for example, the included small blue hairpin in the top structure between number three and four. In the bottom RNA, this hairpin is opposite to the small yellow stem with number five, whereas in the top structure this stem is situated in another region. The 16S rRNAs comparison shows significant similarities in nearly all stem and loop regions. Note, for both examples the set $\mathbf{E}_{\gamma}^{1,2}$ was computed with $\gamma = 2$.

For the comparison of the results we have chosen `RNA_align` and `RNAforester`. The first method computes sequence structure alignments according to the general edit distance algorithm [JLMZ02]. The `RNAforester` program from [HTGK03] is build upon the tree alignment algorithm for ordered trees from [JWZ95] and extends it to calculate forest alignments. A comparison of these methods with our approach is possible on the number of common realized alignment edges. Therefore, we have first computed the alignments for both RNA pairs. Next, we have extracted from these alignments all positions with exact sequence structure matchings and determined the intersections with LCS-EPM. Note, the time for `expaRNA` in Table 1 includes the time to determine all EPMs for the two IRES RNAs $(0.44s)$ and for the two 16S rRNAs $(1.2s)$. The sequence coverage rate is averaged over both RNAs.

	IRES RNAs			16S rRNAs		
	#matches	coverage	time	#matches	coverage	time
expaRNA	175	45%	$0.97s$	875	57%	$16.9s$
RNA_align	192	50%	$62.1s$	861	56%	$1h\,35m$
RNAforester	128	33%	$5.41s$	847	55%	$7m\,25s$

comparison	IRES RNAs #common matches	16S rRNAs #common matches
expaRNA & RNA_align	159 (82.8%)	688(79.9%)
expaRNA & RNAforester	103 (80.5%)	700(82.6%)

Table 1: Comparison of the number of found exact matchings by LCS-EPM and two alignment methods. In the lower part, *#common matches* defines the number of identical matched nucleotides of expaRNA and the other methods.

4 Conclusion

We have developed a new algorithm for the pairwise sequence-structure comparison of RNAs and implemented it in the program expaRNA. Our approach utilizes common substructures for the detection of global similarities between two RNAs. We have applied the presented dynamic programming algorithm to two Hepatitis C virus IRES RNAs and two 16S ribosomal RNAs. In comparison to existing alignment methods, our approach found about 80% of their found exact matching edges. This also supports our assumption that "good" alignments realize a large number of common substructures. In addition, a complete gapped global alignment can be easily calculated, if the found LCS-EPM are used as anchor constraints. The impressive performance of expaRNA, in particular for large RNA molecules may allow its application as a fast prefiltering method for time-consuming RNA sequence-structure comparison approaches. This would allow genome-wide application of these methods.

5 Acknowledgment

This work has been supported by the Federal Ministry of Education and Research (BMBF grant 0313921 FORSYS/FRISYS) and the German Research Foundation (DFG grant BA 2168/2-1 SPP 1258).

References

[AMS+97] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–402, 1997.

[AS05] Julien Allali and Marie-France Sagot. A new distance for high level RNA secondary structure comparison. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(1):3–14, 2005.

[BFRS03] Guillaume Blin, Guillaume Fertin, Irena Rusu, and Christine Sinoquet. RNA sequences and the EDIT(NESTED,NESTED) problem. Technical Report RR-IRIN-03.07, IRIN, Université de Nantes, 2003.

[BMR95] V. Bafna, S. Muthukrishnan, and R. Ravi. Computing similarity between RNA strings. In *Proc. 6th Symp. Combinatorial Pattern Matching*, pages –16, 1995.

[BS07] Rolf Backofen and Sven Siebert. Fast Detection of Common Sequence Structure Patterns in RNAs. *Journal of Discrete Algorithms*, 5(2):212–228, 2007.

[CSS+02] J. J. Cannone, S. Subramanian, M. N. Schnare, J. R. Collett, L. M. D'Souza, Y. Du, B. Feng, N. Lin, L. V. Madabusi, K. M. Muller, N. Pande, Z. Shang, N. Yu, and R. R. Gutell. The Comparative RNA Web (CRW) Site: an online database of comparative sequence and structure information for ribosomal, intron, and other RNAs: Correction. *BMC Bioinformatics*, 3(1):15, 2002.

[Eva99] Patricia Anne Evans. *Algorithms and Complexity for Annotated Sequence Analysis*. PhD thesis, University of Alberta, 1999.

[GJMM+05] Sam Griffiths-Jones, Simon Moxon, Mhairi Marshall, Ajay Khanna, Sean R. Eddy, and Alex Bateman. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Research*, 33 Database Issue:D121–4, 2005.

[HFS+94] Ivo L. Hofacker, Walter Fontana, Peter F. Stadler, Sebastian Bonhoeffer, Manfred Tacker, and Peter Schuster. Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte Chemie*, 125:167–188, 1994.

[HK96] M. W. Hentze and L. C. Kuhn. Molecular control of vertebrate iron metabolism: mRNA-based regulatory circuits operated by iron, nitric oxide, and oxidative stress. *Proc. Natl. Acad. Sci. USA*, 93(16):8175–82, 1996.

[HTGK03] Matthias Höchsmann, Thomas Töller, Robert Giegerich, and Stefan Kurtz. Local Similarity in RNA Secondary Structures. In *Proceedings of Computational Systems Bioinformatics (CSB 2003)*, page 159. IEEE Computer Society, 2003.

[HWB96] A. Huttenhofer, E. Westhof, and A. Bock. Solution structure of mRNA hairpins promoting selenocysteine incorporation in Escherichia coli and their base-specific interaction with special elongation factor SELB. *RNA*, 2(4):354–66, 1996.

[JLMZ02] Tao Jiang, Guohui Lin, Bin Ma, and Kaizhong Zhang. A General Edit Distance between RNA Structures. *Journal of Computational Biology*, 9(2):371–88, 2002.

[JWZ95] T. Jiang, J. Wang, and K. Zhang. Alignment of trees - an alternative to tree edit. *Theoretical Computer Science*, 143(1):137–148, 1995.

[LCJW02] Guohui Lin, Zhi-Zhong Chen, Tao Jiang, and Jianjun Wen. The longest common subsequence problem for sequences with nested arc annotations. *J. Comput. Syst. Sci.*, 65(3):465–480, 2002.

[MLBM+04] Yvan Martineau, Christine Le Bec, Laurent Monbrun, Valerie Allo, Ing-Ming Chiu, Olivier Danos, Herve Moine, Herve Prats, and Anne-Catherine Prats. Internal Ribosome Entry Site Structural Motifs Conserved among Mammalian Fibroblast Growth Factor 1 Alternatively Spliced mRNAs. *Mol Cell Biol*, 24(17):7622–35, 2004.

[SP07] Alexander Serganov and Dinshaw J. Patel. Ribozymes, riboswitches and beyond: regulation of gene expression without proteins. *Nat Rev Genet*, 8(10):776–90, 2007.

[WSPB97] R. Wilting, S. Schorling, B. C. Persson, and A. Böck. Selenoprotein Synthesis in Archaea: Identification of an mRNA Element of Methanococcus jannaschii Probably Directing Selenocysteine Insertion. *Journal of Molecular Biology*, 266(4):637–41, 1997.

[ZS89] Kaizhong Zhang and Dennis Shasha. Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems. *SIAM J. Comput.*, 18(6):1245–1262, 1989.