A New Procedure to Analyze RNA Non-Branching Structures

Giulia Fiscon[1,2*], Paola Paci[1], Teresa Colombo[3], and Giulio Iannello[4]

[1] Institute for System Analysis and Computer Science "Antonio Ruberti" (IASI), CNR, Viale Manzoni 30, 00185 Rome, Italy.
[2] Department of Computer, Control and Management Engineering (DIAG), "Sapienza" University of Rome, Viale Ariosto 25, 00185 Rome, Italy.
[3] Institute for Computing Applications "Mauro Picone" (IAC), CNR, Via dei Taurini 19, 00185, Rome Italy.
[4] Centro integrato di ricerca, Università Campus Bio-medico di Roma, Via Alvaro del Portillo 21, 00128, Rome, Italy.
* Corresponding author: Email: fiscon@dis.uniroma1.it; Tel: (+39) 06 7716435; fax (+39) 06 7716461.

Abstract

We propose a novel procedure to detect structural motifs shared between two RNAs (a reference and a target). In particular, we developed two core modules: (i) nbRSSP_extractor, to assign a unique structure to the reference RNA encoded by a set of non-branching structures; (ii) SSD_finder, to detect structural motifs that the target RNA shares with the reference, by means of a new score function that rewards the relative distance of the target non-branching structures compared to the reference ones. We integrated these algorithms with already existing software to reach a coherent pipeline able to perform the following two main tasks: prediction of RNA structures (integration of RNALfold and nbRSSP_extractor) and search for chains of matches (integration of Structator and SSD_finder).

Keywords:

1. Common structural motifs
2. Long non-coding RNA
3. Dynamic programming
4. RNA local structure prediction
5. RNA secondary structure
6. RNA structure comparison

Background

The prediction of RNA secondary structures and the search for structural motifs shared between two RNAs are really computational onerous problems, as much as cutting-edge topics in the RNA functional studies. Similarities between two nucleic acid chains are usually investigated by taking into account only for the primary structure (sequence) and thus ignoring structural elements.

The issues become more complicate with the discovery of long non-coding RNAs (lncRNAs), generally classified as antisense, intronic or intergenic transcripts longer than 200 nucleotides and lacking significant open reading frames[1, 2, 3, 4, 5, 6, 7, 8]. For a long time lncRNAs were dismissed as "transcriptional noise" [9] because of their low level of expression [10] and general absence of evolutionary sequence conservation [11]. However, it has become increasingly apparent that lncRNAs are important regulatory molecules in many physiological and pathological cellular processes [12, 13, 14]. In fact, a bulk of recent evidence shows that the expression of lncRNAs is modulated in response to specific stimuli[15, 16] and suggests their crucial involvement in transcriptional and post-transcriptional control mechanisms as well as in epigenetic processes and, particularly, in chromatin remodeling[17, 18, 19, 20]. This corroborates the assumption that the lack of conservation does not imply lack of functionality. Indeed, it has been suggested that some lncRNAs can act as scaffolds for multiple proteins or as guides to recruit effector proteins to specific genomic regions [21, 22]. Therefore, a way of functioning relied on lncRNAs structures can constitute a functional signature to search for in order to infer the putative mechanism of action of uncharacterized lncRNAs.

To functionally characterize a lncRNA, or more generally, any RNA with unknown function (*target RNA*), one could search for structural motifs potentially common to a lncRNA (or RNA) whose function has been already identified *(reference RNA*). This implies to assign a structure to the reference RNA and to look for structural similarities with a target RNA.

There exist several tools performing the RNA secondary structure prediction (Table 1 and 2), as well as detecting RNA structural similarities (Table 3 and 4). However, they are not immediately suitable to deal with lncRNAs for two main reasons. First, large part of the existing tools are unable to efficiently deal with long nucleotides sequences (e.g., all tools listed in Table 3 and mostly of Table 1 and Table 4); second, most of the listed tools requires multiple sequence alignment [23] (e.g., all tools listed in Table 2 and some of Table 3 and Table 4), which are generally not available for lncRNAs.

Pursuing the idea of functionally characterize RNA by seeking structural similarities with a reference RNA, it would be very useful to have a unique software capable of analyzing RNA sequences of any length, short as well as long RNAs.
Here we present a novel package MONSTER v1.0 (Method Of Non-branching STructures Extraction and seaRch), which integrates some existing tools with *ad-hoc* implemented algorithms in one new coherent pipeline. MONSTER v1.0 mostly consists of two core modules: one for extracting RNA non-branching structure (*nbRSSP_extractor*), and one for detecting chains of matches shared between two RNAs (*SSD_finder).* MONSTER v1.0 makes use of *RNALfold* from the Viennan RNA Package [24, 25], to obtain the folding predictions and make use of *Structator* [26] to obtain the searching of shared matches between target and reference RNA.

This decision stems from the specific features of both selected tools. In particular, *RNALfold* is a prediction tool based on thermodynamic models[27, 28, 29, 30] that performs a local folding (i.e., a restriction on the span of base-pairs of the RNA molecule is taken into account, rather than the structure of the entire RNA). It has been shown that thermodynamic models leads to very fast algorithms and reliable local structure predictions [31, 32].

On the other hand, *Structator* appears as the most computationally efficient software to deal with long sequences. It is able to perform two different tasks: the *matches* and *chains searching*. The matches searching provides the occurrences of Non-Branching-Structures (NBSs), representing the reference structure, into a target sequence, considering as the only constraint the target base-paring. The chains searching identifies groups of matched NBSs representing sub-structures shared between the reference and the target.

Taking advantage of *Structator* efficiency, MONSTER v1.0 uses it to perform the matches searching sub-task, while employs a novel dynamic programming algorithm (*SSD_finder*) to achieve the chains searching sub-task. *SSD_finder* rewards the rightness of the NBS relative position in both the reference and the target with an appropriate score function.

Following [26], we choose to characterize the folding on an RNA sequence by means of a Sequence Structure Descriptor (SSD) (i.e., a sequence of NBSs positioned on the RNA sequence). However, a problem exists of output/input incompatibility between *RNALfold*, which provides in output overlapped branching structures, and *Structator*, which requires in input only NBSs. To overcome this limit, MONSTER v1.0 employs a new algorithm (*nbRSSP_extractor*) to extract from the *RNALfold* output the more stable NBSs and to encode them into the suitable format for *Structator*.

To test the MONSTER procedure, we evaluate the performance of the two core modules (*nbRSSP_extractor* and *SSD_finder*), using dataset of RNAs with known structures (rRNAs) and class of RNA families obtained from online freely available database (e.g., Rfam and RNAstrand2.0). The results are reported in the section *Results and Discussion*.

Finally, we use MONSTER v1.0 to study two lncRNAs, HOTAIR and ANRIL, that are long intergenic non-coding RNAs (lincRNAs). In particular, they constitute exemplar lncRNAs whose function is related to their structure. It has been shown [21, 22] that HOTAIR and ANRIL interact with the same chromatin-remodeling complex (Polycomb Repressive Complex 2), and they could share some structural motifs. The entire procedure is thoroughly explained in the "Basic Usage" section of the *User_Guide* (provided as an additional file of this paper).

Materials and Methods

In order to functional characterize an RNA (target RNA) one may search for structural motifs that are shared with RNA of known function (reference RNA). We propose that this task can be accomplished using MONSTER that consists in the following procedure (Figure 1):

step 1. Selection of a functionally uncharacterized RNA (target RNA).
step 2. Selection of a functionally annotated RNA (reference RNA).
step 3. Extraction of the NBSs representing the reference RNA.
step 4. Encoding of reference NBSs into an SSD.
step 5. Searching for matches of the SSD of the reference RNA in the target RNA sequence.
step 6. Filtering out of low-probability matches.
step 7. Detecting top-candidate chains of matches that the target RNA may share with the reference RNA.

In the following sub-sections, we discuss every step of the pipeline for a given pair of target and reference RNA.

Step 3: extraction of the reference RNA non-branching structures
The core module of MONSTER that we called *nbRSSP_extractor* performs this task. First of all, we need secondary structure predictions of the reference RNA. To this aim, we use *RNALfold* that provides locally stable sub-structures according to a given parameter *L* representing the maximum allowed distance between a base-pair. *RNALfold* also computes for each sub-structure both the starting position in the sequence and its free energy. It is worth noting that two or more sub-structures may overlap (i.e., more predictions correspond to an identical piece of sequence). Thus, *RNALfold* gives as output a list of all possible local sub-structures. However, a unique prediction has to be composed by non-overlapping sub-structures. The core module *nbRSSP_extractor* extracts a set of non-branching structures that do not overlap, representing the structure of the reference RNA. We introduce the following definitions:

Definition 1
Two local predictions of non-branching structures are the same sub-structures if: (i) their structural description coincides in length, base-pairs, and unpaired bases, and (ii) they are placed at the same initial position on the RNA sequence.

Definition 2
Two local predictions of non-branching structures that are not the same sub-structures have a non-branching sub-structure in common if: (i) the length of their external loop is the same, and (ii) this loop is placed at the same positions on the RNA sequence.

Note that the common sub-structures of *Definition 2* between two local predictions of non-branching structures extend before and after the external loop as long as paired and unpaired bases of the two predictions coincide. Therefore in the following, unless differently stated, the common part is the larger possible.

The module *nbRSSP_extractor* extracts from the *i*-th sub-structure (*i=1...N* with *N* number of the sub-structures provided by *RNALfold*) the set of non-branching structures (nbs_i). Let $u^{(i)}$ be one of the non-branching structures belonging to nbs_i of the *i*-th sub-structure and $v^{(j)}$ one of the non-branching structures belonging to nbs_j of the *j*-th sub-structure. It may happen that $u^{(i)}$ and $v^{(j)}$ coincide or have a non-branching sub-structure in common (i.e., either $u^{(i)}$ is strictly contained in $v^{(j)}$, or $v^{(j)}$ strictly contains $u^{(i)}$, or the common part is strictly contained in both $u^{(i)}$ and $v^{(j)}$). Based on this observation, *nbRSSP_extractor* constructs the set of all different Non-Branching Predictions (*NBP*), including the ones that are in common between any pair of different predicted local structures computed by *RNALfold*.

For each $k \in NBP$, the module *nbRSSP_extractor* computes the *mean free energy per base* defined as:

$$me_{pb}(k) = \left(\frac{1}{n(k)} \cdot \sum_{i=1}^{n(k)} \frac{e_i}{l_i} \right)$$

where e_i is the free energy of the *i*-th sub-structure provided by *RNALfold*, l_i is its length, and $n(k)$ represents the occurrences of *k* in the structure predictions.

Then, *nbRSSP_extractor* sorts *NPB* according to decreasing me_{pb} and, starting from the first element, constructs a list of NBSs by selecting all predictions that do not overlap. This list is then reordered according to increasing position in the sequence.

Step 4: encoding of reference NBSs into an SSD

This task is performed by the core module *nbRSSP_extractor* of MONSTER. An RNA secondary structure (Figure 2a) can be broken down into separated NBSs (Figure 2b) that are conveniently represented by dot-bracket notation (Figure 2c). The list of NBSs that describes the RNA secondary structure is the SSD (Figure 2d). Following [26], we finally represent each NBS of the SSD as an RNA Sequence-Structure Pattern (RSSP). More specifically, an RSSP is a pair formed by a sequence (i.e., a string of bases) and a structure (i.e., a string representing the secondary structure in the dot-bracket notation). The format used includes also a list of parameters associated to the RSSP, such as its position in the sequence, the number of times it has been predicted and its me_{pb}.
Since we are interested in finding structural similarities without specific sequence constraints, we set all nucleotides of the RSSP sequences to wildcard characters N that can be equal to A/U/G/C.

Summarizing, an SSD represents the set of non-overlapping NBSs that are likely to be present in the folded RNA. Contiguous subsets of these NBSs can be considered representative of structural motifs.

Step 5: searching for matches of the SSD of the reference RNA in the target RNA sequence

The *Structator* module called *afsearch* performs this task. It searches for all matches among sub-sequences in the target that could fold into the NBSs found in the reference by *nbRSSP_extractor*. The core module *nbRSSP_extractor* encodes the NBSs to be searched in the format required by *Structator* (i.e., the SSD descriptor of the reference RNA structure).

First of all, we summarize the notation that we will use throughout the next sections:

- *R* is the *reference sequence*;
- *S* is the list of NBSs present in the predicted structure of *R*, sorted in increasing sequence positions;
- s_i is the *i*-th NBS in *S*, pos(s_i) is its position in *R*, and length(s_i) its length;
- *T* is the *target sequence*;
- *M* is the list of matches found in *T*, sorted in increasing sequence positions;
- m_i is the *i*-th match in *M*, pos(m_i) is its position in *T*, length(m_i) its length, and nbs(m_i) is the NBS in *S* which m_i corresponds to;

- ind(·) can be applied to both NBSs in *S* and matches in *M*, and it gives the index of the argument in the respective list (starting from 1).

Structator takes as input *S*, *T* and the set of allowed base-pairings, and produces as output *M*, that corresponds to pairs consisting of:

- the index *i* into *S* of the matching NBS s_i;
- the position p_i of the matching subsequence of *T* (i.e., a subsequence that can potentially fold into s_i according to base-pairing rules).

Note that *M* contains all potential matches, including overlapping matches and matches that do not respect the order of NBSs in *S*. Therefore, the latter have to be further processed to extract the ones that could correspond to interesting structural motifs.

Step 6: filtering out of low-probability matches

The module of MONSTER that we called *matches_filter* performs this task. This module filters out unlikely matches obtained from the step 5.
In fact, the *Structator* module *afsearch* looks for matches taking into account the potential base-pairing as the only constraint. This represents only a necessary condition and it could produce many false positives.
To discard unlikely matches, we apply again the equal schema used to predict the structural motifs of the reference sequence (step 3-4), with the difference that we use a less selective criterion to accept sub-structures of putative matches: only the matches whose NBSs appear in the list predicted by *RNALfold*/*nbRSSP_extractor* analysis are retained. Since the contribution of a match to find not trivial structural motifs can be associated to the corresponding NBS length, we assigned to each match a weight proportional to this length.

Step 7: detecting top-candidate chains of matches that the target RNA may share with the reference RNA

The core module of MONSTER that we called *SSD_finder* performs this task. This module finds groups of matches that may correspond to structural motifs shared between *R* and *T*.

Since the step 6 returns a list of matches corresponding to single NBSs only, we should search for groups of matches that:

- correspond to NBSs that are close on the reference sequence;
- preserve on the target sequence the order and the relative positions that the corresponding NBSs have on the reference sequence.

Let us consider a chain $C = \{m_{j_1}, m_{j_2}, ..., m_{j_n}\}$ of matches in *M* satisfying $\forall i, 1 \le i \le n-1$ the following conditions:

(i) $\text{ind}\left(\text{nbs}(m_{j_i})\right) < \text{ind}\left(\text{nbs}(m_{j_{i+1}})\right)$

(ii) $\text{pos}(m_{j_i}) + \text{length}(m_{j_i}) \le \text{pos}\left(m_{j_{i+1}}\right)$

Note that condition (i) implies that *C* is ordered according to increasing positions in *T*, hence that $j_i < j_{i+1},\ \forall i,\ 1 \le i \le n-1$. To simplify notation, hereafter we will denote the matches in a chain as $m_1, ..., m_n$.
Based on these definitions, we define the score of *C* as follows:

$$sc(C) = \sum_{i=1}^{n} P(m_i) + \sum_{i=1}^{n-1} \text{Q}(m_i, m_{i+1}) \qquad (1)$$

where:

- $P(m_i)$ is the weight of match m_i taking into account its individual relevance;

- $Q(m_i, m_{i+1})$ is a weight taking into account how much the pair (m_i, m_{i+1}) in *T* has positions consistent with the corresponding NBSs in *R*.

This score is then used to select chains that could correspond to non-trivial structural motifs present both in *R* and *T*.

As a comparison, the global chains search algorithm, implemented by *Structator*, sets *Q* to 0 for any pair of matches. In this way, *Structator* finds the chain containing the matches whose sum of weights is maximum independently of their relative positions.
On the contrary, we define $Q(m_i, m_{i+1})$ as follows:

$$Q(m_i, m_{i+1}) = \begin{cases} Q_1 + Q_2 & \text{if } (Q_1 + Q_2) \geq 0 \\ -\infty & \text{otherwise} \end{cases}$$

where

$$Q_1(m_i, m_{i+1}) = GAP_{nbs} - \left[\text{ind}\left(\text{nbs}(m_{i+1})\right) - \text{ind}\left(\text{nbs}(m_i)\right)\right]$$

and

$$Q_2(m_i, m_{i+1}) = GAP_{pos} - \frac{\left|\left[\text{pos}\left(\text{nbs}(m_{i+1})\right) - \text{pos}\left(\text{nbs}(m_i)\right)\right] - \left[\text{pos}(m_{i+1}) - \text{pos}(m_i)\right]\right|}{\text{pos}\left(\text{nbs}(m_{i+1})\right) - \text{pos}\left(\text{nbs}(m_i)\right)}$$

GAP_{nbs} and GAP_{pos} are two thresholds. The first one corresponds to distance between nbs(m_i) and nbs(m_{i+1}) beyond which Q_1 becomes negative. The second one corresponds to the discrepancy between the distances of reference and target NBSs. We choose $GAP_{nbs} = 3$ meaning the nbs(m_i) and nbs(m_{i+1}) are considered close if the distance between the corresponding NBSs in the reference is lower than 3; and $GAP_{pos} = 0.1$ meaning that the distance between m_i and m_{i+1} in the target is considerable acceptable if the difference with the corresponding distance in the reference is at most 10%. Note that the score (1) may evaluate -∞, when Q=-∞. This implies the rejection of chains containing matches whose positions on *T* are too different from the corresponding NBSs on *R*.
In Figure 3 is shown an example of the importance of the *Q* term: given a reference RNA sequence composed of four RSSPs and a target RNA sequence having six matches to them, two chains of matches can be extracted (chain 1 and chain 2 in the figure). A shorter chain made of two RSSPs and a longer one made of four RSSPs. However, while the RSSP1-RSSP2 distance in the shorter chain is preserved, the same is distorted in the longer one. Thus, rewarding only the number of matched RSSPs (term *P*), the second chain would get the best score, neglecting the potentially most representative first chain.

We compute the score for all chains of matches in *M* satisfying conditions (i) and (ii), and then select chains with the highest score. However, this is unfeasible for long sequences, since its complexity grows exponentially with the number of matches. To reduce the complexity we consider for all matches $m \in M$ only the chain ending with *m* that has the highest score. This can be done with dynamic programming using the recursion:

$$OPT(m_i) = P(m_i) + \max_{j \in C}\left\{Q(m_{j,} m_i) + OPT(m_j)\right\} \qquad (2)$$

where:

$$C = \{\text{j} \mid \text{j} < \text{i} \wedge \text{ind}\left(\text{nbs}(m_j)\right) < \text{ind}\left(\text{nbs}(m_i)\right) \wedge \text{pos}(m_j) + \text{length}(m_j) \leq \text{pos}(m_i)\},$$

and conventionally assuming that in the trivial case ($C = \emptyset$), the second term in (2) is equal to 0.
$OPT(m_i)$ gives for any $m_i \in M$ the highest score of chains ending with m_i and the corresponding optimal chain can be easily determined by backtracking.
The dynamic programming algorithm therefore returns one optimal chain for every match in *M* and we can select chains with the highest scores as candidates to represent possible common structural motifs between reference and target sequences.

Results and Discussion

In the following, we show the computational experiments carried out to test the performance of the MONSTER core modules (*nbRSSP_extractor* and *SSD_finder*).

Evaluation of nbRSSP_extractor module performance

To evaluate the performance of *nbRSSP_extractor* predictions, we use a dataset of rRNAs obtained from the RNAstrand_v2.0 database (http://www.rnasoft.ca/strand/). This database collects known RNA secondary structures for different RNA type and organisms. We select rRNAs with sequence length larger than 1000 bases: rRNA16S (723 sequences) and rRNA23S (205 sequences).

We use *RNALfold* to predict for each input sequence its secondary structure; we run *nbRSSP_extractor* module to extract the NBSs and hence to build the SSD (*predicted SSD*). Likewise, we take the known structures of rRNA16S and rRNA23S and we apply *nbRSSP_extractor* to extract the NBSs and hence to obtain the SSD of the known structures (*known SSD*).

We implemented a further algorithm (*SSD_compare*[1]) to compare *predicted* and *known* SSDs. Moreover, we added an option to the module *nbRSSP_extractor* to exclude non-overlapping predictions of *RNALfold* in an alternative way[2], that we called *RNALfold_lnrz* analysis.

Finally, we compare *RNALfold*/*nbRSSP_extractor* analysis with the state-of-the-art prediction tool *Rfold* (with the usual base-pair span of 150), *RNAfold* (with default parameters) and with *RNALfold_lnrz* analysis.

Table 5 shows the results obtained for the four analyzed procedures in term of True Positive (TP) and False Positive (FP) values. In particular, a TP value represents a base-pair of the predicted structure having a corresponding base-pair in the known one, whereas a FP value represents a predicted base-pair for which there is not a corresponding base-pair in the known structure.
Our algorithm produces a number of TP higher than other considered tools, although it yields a higher number of FP too. However, it requires drastically lower computational costs, as discussed below.

We then build a ROC curve for *nbRSSP_extractor* by computing the True Positive Rate (*TPR*, or sensitivity) and the False Positive Rate (*FPR*) as functions of the parameter me_{pb}, assigned to each predicted NBS (Figure 4a for rRNA 16S and Figure 4b for rRNA 23S).
TPR is defined as usually as:

$$TPR(me_{pb}) = \frac{TP(me_{pb})}{P}$$

where $TP(me_{pb})$ is the TP value when all predicted NBSs with score lesser tha me_{pb} are discarded; while P is the set of all positive values, given by the total number of base-pairs in the known structure.
FPR is defined as:

$$FPR(me_{pb}) = \frac{FP(me_{pb})}{FP}$$

where $FP(me_{pb})$ is the FP value when all predicted NBSs with score lesser than me_{pb} are discarded; while the denominator represents the total number of FP values (i.e., me_{pb} set to zero).
As previously observed, the explained algorithms show different computational complexity. Therefore, we measured the time required to compute the structure predictions and extract the NBSs for increasing sequence lengths (n). The results are shown in Figure 5. As expected the computational time of *RNALfold* followed by our *nbRSSP_extractor* to optimize the NBSs selection (red curve of Figure 5a) is linear with respect to n. In addition, Figure 5b depicts the comparison among all the tested algorithms for increasing n values, using a logarithmic scale for both axes. *RNAfold*

[1] *SSD_compare* computed the matches between the known and the predicted structures. It returned the number of base-pairs that were correctly predicted by *nbRSSP_extractor*.
[2] The predictions of *RNALfold* are selected basing their decreasing free energies, and then the non-overlapping ones are chosen.

(violet curve in Figure 5b) has a polynomial computational time of $O(n^{2.4})$. *RNALfold* with the optimized NBSs selection (red curve in Figure 5b) and *RNALfold* without any optimization (blue curve in Figure 5b) show the same performances (i.e., the curves overlap). In particular, their trend is equal to the *RNAfold* one up to an input size of 150 as expected, while it becomes linear for longer sequences. Lastly, also the local folding algorithm *Rfold* (green curve in Figure 5b) shows a double trend: it exhibits the time performances slightly higher than *RNAfold* up to the input size of 150, while it reveals a trend slightly more than linear ($O(n^{1.15})$) for larger sequence lengths, but with much higher multiplicative constants with respect to the *RNALfold* case.

Evaluation of SSD_finder module performance

To evaluate *SSD_finder*, we measure its performance in the identification of members of four families obtained from the RFAM 11.0 database (http://rfam.sanger.ac.uk/). RFAM is a curated database of ncRNA families aimed at providing an automated and common system for the analysis and annotation of ncRNA sequences. Each RNA family in the database is represented by a multiple sequence alignment that includes both a subset of manually-curated known members of the family and automatically inferred members based on sequence homologies.

The selected RFAM families are the following: (i) the Citrus tristeza virus replication signal (RFAM Acc.: RF00193), a regulatory element which plays a crucial role in the virus replication through its structures [33]; (ii) the small ncRNAs OxyS family (RFAM Acc.: RF00035), induced in response to oxidative stress in Escherichia Coli [34]; (iii) the lncRNAs family HAR1A (RFAM Acc.: RF00635), overlapping the Human Accelerated Region 1 (HAR1); and (iv) the lncRNAs family HOTAIRM1 (RFAM Acc.: RF01975), acting in myeloid transcriptional regulation[35].

We apply MONSTER to each of the aforementioned families, the experiment workflow is the following (Figure 6): (i) the multiple sequences alignment of the family is used as the reference; (ii) a database of RNA sequences, including the four selected families and a subset of families randomly extracted from the RFAM and RNAstrand databases (more than 700 sequences in total), is used as the target; (iii) through *RNAalifold* [36], we obtain the consensus structure prediction of the family, that we use as input to *nbRSSP_extractor* to obtain the SSD of the reference (Figure 7); (iv) through *Structator*, we search for this reference SSD in the target; (v) the returned matches are used as an input to *SSD_finder* which computes the chains of matches with the highest score; note that, since the chains with length one can be conceivably considered not significant, we filter out them in this step.

The chains returned by the MONSTER module *SSD_finder* for each family have been sorted in decreasing order with respect to the score in order to evaluate if this score can be able to discriminate the reference RFAM family among a width of false elements.

For comparison, the chaining analysis for the same RFAM families is performed using *Structator* both in global and local modes (according to equations (2) and (3) of [26], respectively). The results for the four families are reported in Table 6.
All three methods failed to recover all members the given family under exam. The reason could be that *Structator* matching algorithm only found matches to subsequences that may exactly fold into the reference NBSs. Consequently, it could happen that no matches are found for the family members that have significant gaps in the alignment between their structure and the consensus one.
Focusing on the family members for which high score chains can be found, we note that *SSD_finder* and *Structator* global have the equal good performance for two families (RF00193 and RF00635), which are characterized by a quite specific SSD, consisting of 11 RSSPs and 4 complex RSSPs, respectively.

Concerning the other two families (RF00035 and RF01975) both *SSD_finder* and *Structator* global are able to detect more than 80% of the members of families, however *SSD_finder* achieves higher specificity, since it attained this result with a significantly lower number of FPs.

The Figure 8 shows the trend of the score computed by *SSD_finder*, *Structator* global and local with respect to the target sequences to be covered. In every case, the score computed by *SSD_finder* drastically decreases approaching to the number of sequences that corresponds to the number of the family members. By contrast, the score computed by

Structator shows a gradual decrease, avoiding a clear identification of the exact number of detected members. Thus without a priori-knowledge about the TP values, the rapid decrease observed in the score of *SSD_finder* can be used as selection criterion of the sequences belonging to a given family. In fact, once the list of target sequences has been sorted based on the score, the number of elements belonging to a given reference family can be chosen as the value at which the jump occurs.

Conclusions

We built up a coherent pipeline for detecting structural motifs shared between two RNAs, by integrating some existing tools (i.e., *RNALfold* and *Structator*) with new implemented *ad hoc* tools. We called this procedure MONSTER. The rationale behind our work was to produce a tool able to infer the function of an RNA (target RNA), looking for structural motifs shared with an RNA whose function has been already identified (reference RNA). MONSTER assumes greater importance in the context of the new discovered long non-coding RNA whose function is more likely to be related to structure [37, 38].

The two core modules of MONSTER are: (i) *nbRSSP_extractor*, to assign a unique structure to reference RNA; (ii) *SSD_finder*, to detect the sub-structures, which a target RNA shares with the reference one.
In terms of performances, the module *nbRSSP_extractor* has comparable reliability to existing tools (i.e., *Rfold*, *RNAFold*) and the advantage of significantly lower computational costs.
On the other hand, the module *SSD_finder* offers several key advantages:(i) it identifies groups of matches with high specificity and sensitivity; (ii) it is flexible and hence suitable to interact with others methods which could perform the matches searching; iii) it relies on a specific score function which not only weights the single matches and the chains length, but also rewards the closer relative distance of the target NBSs compared to the reference ones; iv) it is computationally efficient.

Availability and requirements
The developed software package is available as supplementary file (zipped file named: “archive.zip”)

Supplementary material
Additional file 1: User_Guide.pdf. It contains details for setting up of MONSTER_v1.0 package, a tutorial of the MONSTER application, and additional advanced information about the MONSTER algorithms.
Additional file 2: archive.zip. It contains the MONSTER_v1.0 software package, the “data” and “example_data” folders which store all needed files to run the tutorial of the User_Guide file.

List of abbreviations
- lncRNA = long non-coding RNA;
- mfe = minimum free energy;
- nt = nucleotides;
- SCFGs = stochastic context-free grammars;
- bp = base-pair;
- bpp = base-pairing probability;
- mea = maximum expected accuracy;
- cpd = conditional probability distributions;
- NBS = Non-Branching Structure;
- RSSP = RNA Sequence Structure Pattern;
- SSD = Secondary Structure Descriptor;
- me_{pb} = mean free energy per base;
- e = free energy;
- e_{pb} = free energy per base;
- TP = True Positive;
- FP = False Positive;
- TPR = True Positive Rate;
- FPR = False Positive Rate.

Parameters setting

- L=150;
- GAP_{pos}=1/10;
- GAP_{nbs}=3.

Competing interest

The authors declare that they have no competing interests.

Authors' contributions

GI, PP and TC conceived and designed the research. GF and GI developed algorithms. TC and GF collected data. GF analyzed the data. GF and GI made interpretation of data. All authors wrote the paper. All authors read and approved the final manuscript.

Acknowledgments

G.F. acknowledges financial support from The Epigenomics Flagship Project (Progetto Bandiera Epigenomica) EPIGEN funded by Italian Ministry of Education, University and Research (MIUR) and the National Research Council of Italy (CNR). The authors would like to thank Prof. Manuela Helmer Citterich for fresh biological insights and Dr. Paola Bertolazzi for inspiring discussions on the dynamic programming algorithms. A final special thank to Emanuel Weitschek and Vincenzo Bonifaci for their precious advices.

References

[1] J.S.Mattick, «The central role of RNA in human development and cognition,» *Febs Letters*, n. 585, pp. 1600-1616, 2011.

[2] J.S.Mattick, R.J.Taft, K.Pang, T.R.Mercer e M.Dinger, «Non-coding RNAs: regulators of disease,» *Journal of Pathology*, pp. 126-139, 2009.

[3] P.Sumazin, X.Yang, H.Chiu, W.Chung, P.Guarnieri, J.Silva e A.Califano, «An Extensive MicroRNA-Mediated Network of RNA-RNA Interactions Regulates Established Oncogenic Pathways in Glioblastoma,» *Cell*, n. 147, p. 370–381, 2011.

[4] the ENCODE pilot project, «Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project,» *Nature*, vol. 447, pp. 799-816, 2007.

[5] Akimitsu, Keiko, Tano e Nobuyoshi, «Long non-coding RNAs in cancer progression,» *frontiers in genetics*, vol. 3, n. 219, 2012.

[6] P.Amaral e S.Mattick, «Noncoding RNA in development,» *Mamm Genome*, pp. 454-492, 2008.

[7] J. R. Prensner e A. M. Chinnaiyan, «The emergence of lncRNAs in cancer biology,» *Cancer Discov.*, vol. 1, n. 5, pp. 391-407, 2011.

[8] J. S.Mattick, «The Genetic Signatures of Noncoding RNAs,» *PLoS Genet.*, vol. 5, n. 4, p. e1000459, 2009.

[9] J. Ponjavic, C. P. Ponting e G. Lunter, «Functionality or transcriptional noise? Evidence for selection within long noncoding RNAs,» *Genome research*, n. 17, p. 556–565, 2007.

[10] J.S.Mattick e V.Makunin, «Non-coding RNA,» *Human Molecular Genetics,* vol. 15, pp. 17-29, 2006.

[11] K. Pang, M. Frith e J. Mattick, «Rapid evolution of noncoding RNAs: lack of conservation does not mean lack of function,» *TRENDS in Genetics,* vol. 22, n. 1, 2006.

[12] M.Esteller, «Non-coding RNAs in human disease,» *Genetics,* vol. 12, pp. 861-874, 2011.

[13] M.B.Clark e J.S.Mattick, «Long noncoding RNAs in cell biology,» *Seminars in Cell & Developmental Biology,* n. 22, pp. 366-376, 2011.

[14] S. Knowling e K. V. Morris, «Non-coding RNA and antisense RNA. Nature's trash or treasure?,» *Biochime,* vol. 93, n. 11, pp. 1922-1927, 2011.

[15] V. A. Moran, R. J. Perera e A. M. Khalil, «Emerging functional and mechanistic paradigms of mammalian long non-coding RNAs,» *Nucleid Acids Research,* vol. 40, n. 14, p. 6391–6400, 2012.

[16] C.P.Ponting, P.L.Oliver e W.Reik, «Evolution and Functions of Long Noncoding RNAs,» *Cell,* n. 136, p. 629–641, 2009.

[17] M.Cesana, D.Cacchiarelli, I.Legnini, T.Santini, O.Sthandier, M.Chinappi, A.Tramontano e I.Bozzoni, «A Long Noncoding RNA Controls Muscle Differentiation by Functioning as a Competing Endogenous RNA,» *Cell,* n. 147, pp. 358-369, 2011.

[18] I. A. Qureshi, J. S. Mattick e M. F. Mehler, «Long non-coding RNAs in nervous system function and disease,» *elsevier,* pp. 20-35, 2010.

[19] A. Saxena e P. Carninci, «Long non-coding RNA modifies chromatin: epigenetic silencing by long non-coding RNAs,» *Bioessays,* vol. 33, n. 11, pp. 830-839, 2011.

[20] T. Mercer, M. Dinger e J. Mattick, « Long non-coding RNAs: insights into functions,» *Nature Reviews Genetics,* vol. 10, pp. 155-159, 2009.

[21] T.Nagano e P.Fraser, «No-Nonsense Functions for Long Noncoding RNAs,» *Cell,* n. 145, pp. 178-181, 2011.

[22] E. A. Gibb e C. J. B. a. W. L. Lam, «The functional role of long non-coding RNA in human carcinomas,» *Mol Cancer,* vol. 10, n. 38, 2011.

[23] I. Hofacker, M. Fekete e P. Stadler, «Secondary structure prediction for aligned RNA sequences,» *J Mol Biol,* vol. 319, pp. 1059-1066, 2002.

[24] I. Hofacker, B. Priwitzer e P. Stadler, «Prediction of Locally Stable RNA Secondary Structures for Genome-Wide Surveys,» *Bioinformatics,* vol. 20, n. 2, pp. 186-190, 2004.

[25] I.Hofacker, «The Vienna RNA Secondary Structure Server,» 2003.

[26] F.Meyer, S.Kurtz, R.Backofen, S.Will e M.Beckstette, «Structator: fast index-based search for RNA sequence-structure patterns,» *BMC Bioinformatics,* 2011.

[27] T. Xia, J. J. SantaLucia, M. Burkard, R. Kierzek, S. Schroeder, X. Jiao, C. Cox e D. Turner,

«Thermodynamic parameters for an expanded nearestneighbor model for formation of RNA duplexes with Watson-Crick pairs,» *Biochemistry,* vol. 37, pp. 14719-14735, 1998.

[28] D. Mathews, J. Sabina, M. Zuker e D. Turner, « Expanded sequence dependence of thermodynamic parameters provides improved prediction of RNA secondary structure,» *Mol Biol,* vol. 288, pp. 911-940, 1999.

[29] D. H. Mathews e D. H. Turner, «Prediction of RNA secondary structure by free energy minimization,» *Current Opin in Structl Biol.,* vol. 16, n. 3, p. 270–278, 2006.

[30] M. Zucker e P. Stiegler, «Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information,» *Nucleic Acid Res,* vol. 9, pp. 133-148, 1981.

[31] C. Flamm, W. Fontana, I. L. Hofacker e e. al., «RNA folding at elementary step resolution,» *RNA,* vol. 6, pp. 325-338, 2000.

[32] S. J. Lange, D. Maticzka, M. Möhl, J. N. Gagnon, C. M. Brown e R. Backofen, «Global or local? Predicting secondary structure and accessibility in mRNAs,» *Nucleic Acids Research,* vol. 40, n. 12, p. 5215–5226, 2012.

[33] T.Satyanarayana, S.Gowda, M. Ayllon, M.R.Albiach-Marti e W.O.Dawson, «Mutational analysis of the replication signals in the 3'-nontranslated region of citrus tristeza virus,» *Virology,* vol. 300, n. 1, pp. 140-152, 2002.

[34] S. Altuvia, A. Zhang, L. Argaman, A. Tiwari e G. Storz, «The Escherichia coli OxyS regulatory RNA represses fhlA translation by blocking ribosome binding,» *EMBO J.,* vol. 17, n. 20, pp. 6069-75, 1998.

[35] X. Zhang, Z. Lian, C. Padden, M. B. Gerstein, J. Rozowsky, M. Snyder, T. R. Gingeras, P. Kapranov, S. M. Weissman e P. E. Newburger, «A myelopoiesis-associated regulatory intergenic noncoding RNA transcript within the human HOXA cluster,» *Blood,* vol. 113, n. 11, pp. 2526-2534, 2009.

[36] S. H. Bernhart, I. L. Hofacker, S. Will, A. R. Gruber e P. F. Stadler, «RNAalifold: improved consensus structure prediction for RNA alignments,» *BMC Bioinformatics,* vol. 9, n. 474, 2008.

[37] A. M. Khalil, M. Guttman, M. Huarte, M. Garber, A. Raj, D. R. Morales, K. Thomas, A. Presser, B. E. Bernstein, A. v. Oudenaarden, A. Regev, E. S. Lander e J. L. Rinn, «Many human large intergenic noncoding RNAs associate with chromatin-modifying complexes and affect gene expression,» *Proc Natl Acad Sci,* vol. 106, n. 28, p. 11667–11672, 2009.

[38] M.Tsai, O.Manor, Y.Wan, N.Mosammaparast, J.K.Wang, F.Lan, Y.Shi, E.Segal e H.Y.Chang, «Long Noncoding RNA as Modular Scaffold of Histone Modification Complexes,» *Science,* vol. 329, pp. 689-693, 2010.

[39] N. R. Markham e M. Zuker, «UNAFold: software for nucleic acid folding and hybridization,» *Methods in Molecular Biology,* vol. 453, pp. 3-31, 2008.

[40] M. Zuker, «On finding all suboptimal foldings of an RNA molecule,» *Science,* vol. 244, pp. 48-52, 1989.

[41] M. Zuker, «The use of dynamic programming algorithms in RNA secondary structure prediction.,» in *Mathematical Methods for DNA Sequences,* boca raton, M. S. Waterman, 1989, pp. 159-184.

[42] M. Zucker, «Mfold web server for nucleic acid folding and hybridization prediction,» *Nucleic Acids Research,,* vol. 31, n. 13, pp. 3406-3415, 2003.

[43] I.Hofacker, W.Fontana, P.F.Stadler, L. S. Bonhoeffer, M.Tacker e P.Schuster, «Fast Folding and Comparison of RNA Secondary Structures,» *Monatshefte fu¨r Chemie,* vol. 125, pp. 167-188, 1994.

[44] S. Wuchty, W. Fontana, I. L. Hofacker e P. Schuster, «Complete Suboptimal Folding of RNA and the Stability of Secondary Structures,» *Biopolymers,* vol. 49, pp. 145-165, 1999.

[45] Y. Ding e C. E. Lawrence, «A statistical sampling algorithm for RNA secondary structure prediction,» *Nucleic Acids Research,* vol. 31, n. 24, pp. 7280-7301, 2003.

[46] P. Steffen, B. Voß, M. Rehmsmeier, J. Reeder e R. Giegerich, «RNAshapes: an integrated RNA analysis package based on abstract shapes,» *Bioinformatics,* vol. 22, n. 4, pp. 500-503, 2006.

[47] B. H. Stephan, I. L. Hofacker e P. F. Stadler, «Local RNA base pairing probabilities in large sequences,» *Bioinformatics,* vol. 22, n. 5, pp. 614-615, 2006.

[48] H. Kiryu, K. Taishin e A. Kiyoshi, «Rfold: an exact algorithm for computing local base pairing probabilities,» *bioinformatics,* vol. 24, n. 3, pp. 367-373, 2008.

[49] J. S. Reuter e D. H. Mathews, «RNAstructure: software for RNA secondary structure prediction and analysis,» *BMC Bioinformatics,* vol. 11, n. 129, pp. 1471-2105, 2010.

[50] C. Do, D. Woods e S. Batzoglou, «CONTRAfold: RNA Secondary Structure Prediction without Energy-Based Models,» *Bioinformatics,* vol. 22, n. 14, pp. 90-98, 2006.

[51] C. Flamm, W. Fontana, I. L. Hofacker e P. Schuster, «RNA folding at elementary step resolution,» *RNA,* vol. 6, pp. 325-338, 2000.

[52] A. Xayaphoummine, T. Bucher, F. Thalmann e H. Isambert, «Prediction and statistics of pseudoknots in RNA structures using exactly clustered stochastic simulations,» *PNAS,* vol. 100, n. 26, p. 15310–15315, 2003.

[53] A. Xayaphoummine, T. Bucher e H. Isambert, «Kinefold web server for RNA/DNA folding path and structure prediction including pseudoknots and knots,» *Nucleic Acids Res.,* vol. 33, pp. 605-610, 2005.

[54] M. Geis, C. Flamm, M. T. Wolfinger, I. L. Hofacker, M. Middendorf, C. Mandel, P. F. Stadler e C. Thurner, «Folding Kinetics of Large RNAs,» *JMB,* vol. 379, pp. 160-173, 2008.

[55] J. Proctor e M. Meyer, «CoFOLD:an RNA secondary structure prediction method that takes co-transcriptional folding into account,» *Nucl. Acids Res,* 2013.

[56] S. Seemann, J. Gorodkin e J. Backofen, «Unifying evolutionary and thermodynamic information for RNA folding of multiple alignments,» *Nucleic Acids Res,* vol. 36, n. 20, pp. 6355-6362, 2008.

[57] R. Elena e S. R. Eddy, «Noncoding RNA gene detection using comparative sequence analysis,» *BMC Bioinformatics,* vol. 2, n. 8, pp. 8-26, 2001.

[58] I. L. Hofacker, M. Fekete e P. F. Stadler, «Secondary structure prediction for aligned RNA sequences,» *Journal of Molecular Biology,* vol. 319, n. 5, pp. 1059-1066, 2002.

[59] S. Washietl e I. L. Hofacker, «Consensus folding of aligned sequences as a new measure for the detection of functional RNAs by comparative genomics,» *Journal of Molecular Biology,* vol. 342, n. 1, pp. 19-30, 2004.

[60] S. Washietl, I. L. Hofacker e P. F. Stadler, «Fast and reliable prediction of noncoding RNAs,» *Proc Natl Acad Sci U S A,* vol. 102, n. 7, pp. 2454-2459, 2005.

[61] J. S. Pedersen, G. Bejerano, A. Siepel, K. Rosenbloom, K. Lindblad-Toh, E. S. Lander, J. Kent e W. M. a. D. Haussler, «Identification and Classification of conserved RNA secondary structures in the human genome,» *PLoS Comput Biol.,* vol. 2, n. 4, pp. 0251-0262, 2006.

[62] D. d. Bernardo, T. Down e T. Hubbard, «ddbRNA: detection of conserved secondary structures in multiple alignments,» *Bioinformatics,* vol. 19, n. 13, pp. 1606-1611, 2003.

[63] A. Coventry, D. J. Kleitman e B. Berger, «MSARI: multiple sequence alignments for statistical detection of RNA secondary structure,» *Proc Natl Acad Sci U S A,* vol. 101, n. 33, p. 12102–12107, 2004.

[64] N. J. P. Wiebe e I. M. Meyer, «Transat—A Method for Detecting the Conserved Helices of Functional RNA Structures, Including Transient, Pseudo-Knotted and Alternative Structures,» *plos computation biology,* vol. 6, n. 6, 2010.

[65] S. Siebert e R. Backofen, «MARNA: A Server for Multiple Alignment of RNAs,» *GCB,* pp. 135-140, 2003.

[66] S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler e R. Backofen, «Inferring Non-Coding RNA Families and Classes by Means of Genome-Scale Structure-Based Clustering,» *PLoS Comput Biol,* vol. 3, n. 4, p. e65, 2007.

[67] S. Heyne, S. Will, M. Beckstette e R. Backofen, «Lightweight Comparison of RNAs Based on Exact Sequence-Structure Matches,» *bioinformatics,* vol. 25, n. 16, p. 2095–2102, 2009.

[68] D. Wei, L. V. Alpert e C. E. Lawrence, «RNAG: a new Gibbs sampler for predicting RNA secondary structure for unaligned sequences,» *bioinformatics,* vol. 27, n. 18, p. 2486–2493, 2011.

[69] D. A. Sorescu, M. Möhl, M. Mann, R. Backofen e S. Will, «CARNA - Alignment of RNA Structure Ensembles,» *Nucleic Acids Res,* vol. 40, n. W1, pp. W49-W53, 2012.

[70] S. Heyne, F. Costa, D. Rose e R. Backofen, «GraphClust: alignment-free structural clustering of local RNA secondary structures,» *Bioinformatics,* vol. 28, n. 12, pp. i224-i232, 2012.

[71] J. Allali e M.-F. Sagot, «A multiple layer model to compare RNA secondary structures,» *Software—Practice & Experience,* vol. 38, n. 8, pp. 775-792 , 2008.

[72] M.-F.Sagot e J.Allali, «A new distance for high level RNA secondary structure comparison,» *IEEE/ACM Transactions on computational biology and informatics,* vol. 2, n. 1, pp. 3-14, 2005.

[73] G. Blin, A. Denise, S. Dulucq, C. Herrbach e H. Touzet, «Alignment of RNA structures,» *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* 2008.

[74] T. J. Macke, D. J. Ecker, R. R. Gutell, D. Gautheret, D. A. Case e R. Sampatha, «RNAMotif, an RNA secondary structure definition and search algorithm,» *Nucleic Acids Res,* vol. 29, n. 22, pp. 4724-4735, 2001.

[75] J. Reeder, J. Reeder e R. Giegerich, «Locomotif: from graphical motif description to RNA motif search,» *Bioinformatics,* vol. 23, p. i392–i400, 2007.

[76] I. Veksler-Lublinsky, M. Ziv-Ukelson, D. Barash e K. Kedem, «A Structure-Based Flexible Search Method for motifs in RNA,» *JCB,* vol. 14, n. 7, p. 908–926, 2007.

[77] J. Liu, B. Ma e K. Zhang, «An algorithm for searching RNA motifs in genomic sequences,» *Biomolecular Engineering,* vol. 24, p. 343–350, 2007.

[78] G. Pavesi, G. Mauri, M. Stefani e G. Pesole, «RNAProfile: an algorithm for finding conserved secondary structure motifs in unaligned RNA sequences,» *Nucleic Acids Res.,* vol. 32, n. 10, p. 3258–3269, 2004.

[79] E. P. Nawrocki, D. L. Kolbe e S. R. Eddy, «Infernal 1.0: inference of RNA alignments,» *bioinformatics,* vol. 25, n. 10, p. 1335–1337, 2009.

Figures

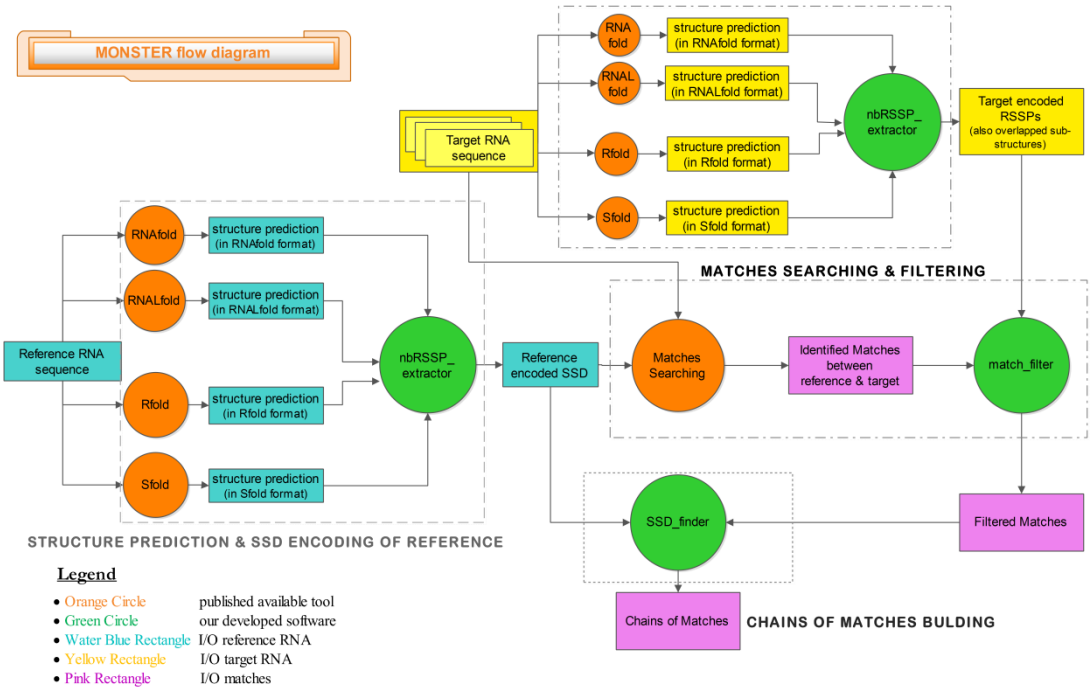

Figure 1 Overview of the MONSTER procedure. The pipeline is composed of three parts: (1) Structure prediction and SSD encoding of the reference (step 1-4 in the text) (2) Matches searching and filtering (step 5-6 in the text); (3) Chains of matches building (step 7 in the text). More details are given in the text and in the user guide.
Legend: orange circles represent published available tools; green circles represent software developed by us; rectangles represent software input and output (I/O), colored with water blue and yellow for what concerns reference and target, respectively.

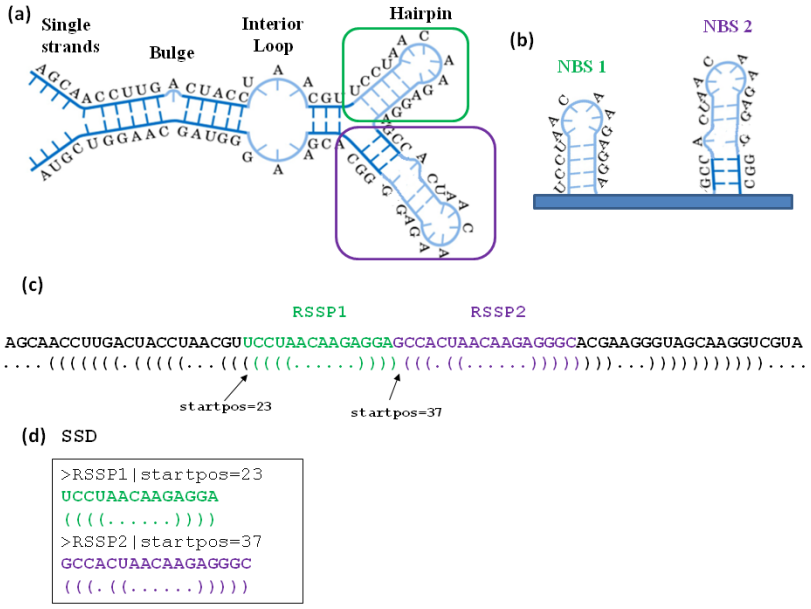

Figure 2 Encoding of RNA secondary structures. a) Example of RNA secondary structure where structural non branching elements are highlighted: *interior loops* (i.e., sequences of unpaired bases linking two different helices); *bulges* (i.e., internal loops caused by unpaired bases only on one side); *hairpins* (i.e, sequences of unpaired bases closing a helix). b) The reference branching structure is broken down into a set of non-branching structures (e.g., NBS 1 and NBS 2). c) Representation of the RNA secondary structure in dot-bracket notation. The RSSPs are highlighted. d) The SSD offers an complete description of the RNA secondary structure.

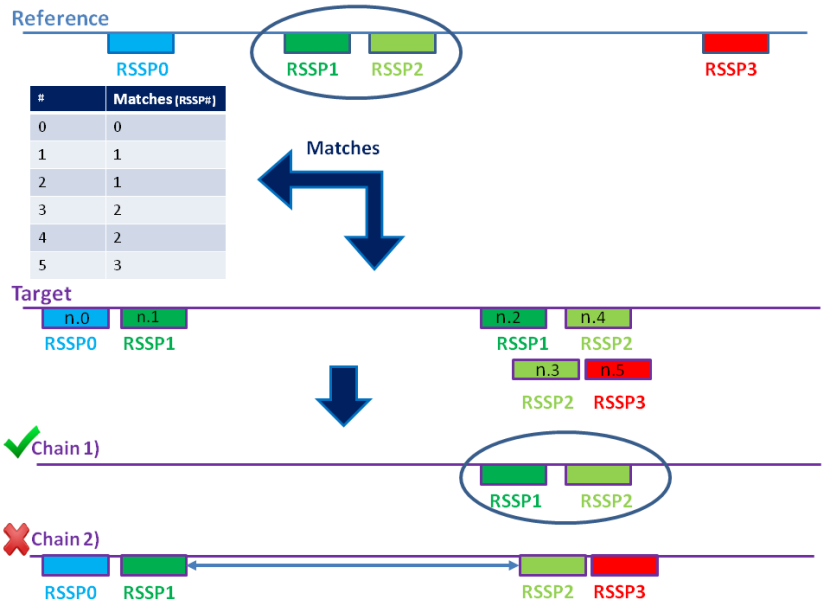

Figure 3 Relevance of the *Q* term. An example of the chaining steps showing the relevance of including the evaluation of the distance between RSSPs (*Q* term) along with the number of RSSPs in the selection of the best chain of matches.

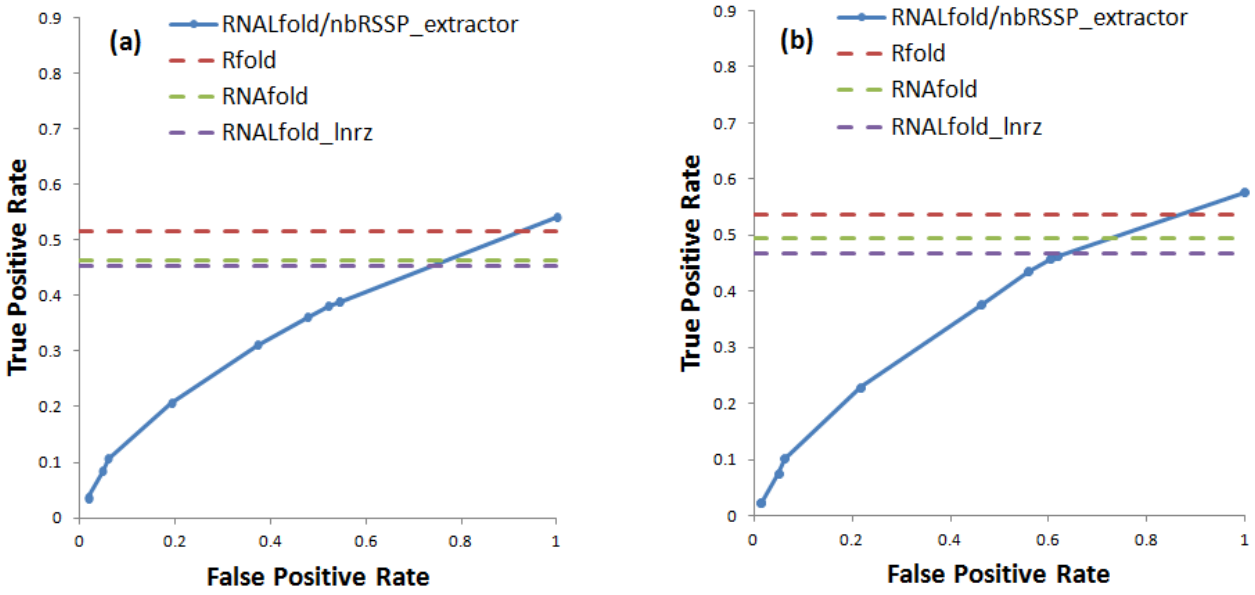

Figure 4 ROC Curves. Plots show the performance of our method (*RNALfold/nbRSSP_extractor*) in terms of True Positive Rate (TPR) *versus* False Positive Rate (FPR), for rRNA16S (a) and rRNA23S (b). TPR and FPR are function of the me_{pb} parameter. Reference performance for other tools (i.e., *RNAfold*, *Rfold*, and *RNALfold_lnrz*) are also indicated for comparison. The *RNAfold*, *Rfold* and *RNALfold_lnrz* performances do not depend on the parameter.
Legend: blue solid line refers to our method; red dashed line refers to *Rfold*; green dashed line refers to *RNAfold*; violet dashed line refers to *RNALfold_lnrz.*

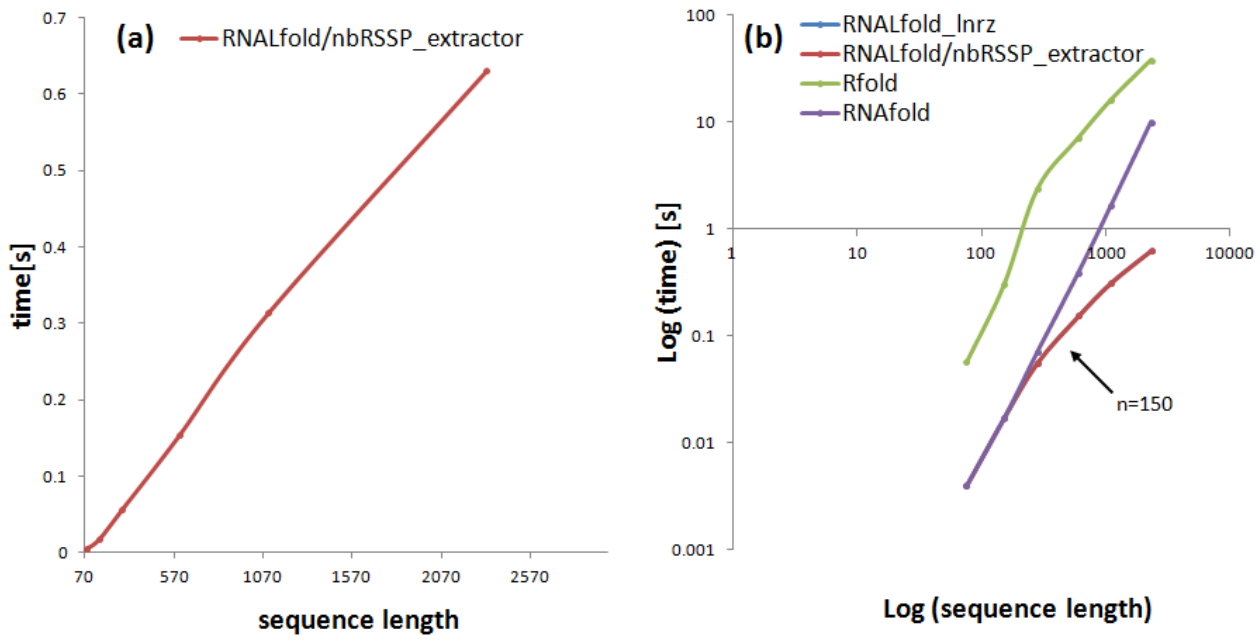

Figure 5 Computational time according to the input size. (a) Computational time of *RNALfold/nbRSSP_extractor* is plotted with respect to the input sequence length. (b) The time performance comparison of all tested algorithms to predict and extract the NBSs is depicted: computational time is plotted with respect to the increasing sequence length, both scales are logarithmic.
Legend: blue and red curves (overlapped) represent the time performance of *RNALfold/nbRSSP_extractor* with both optimized and trivial NBSs selection; green curve refers to *Rfold* time performance; violet curve refers to *RNAfold* time performance.

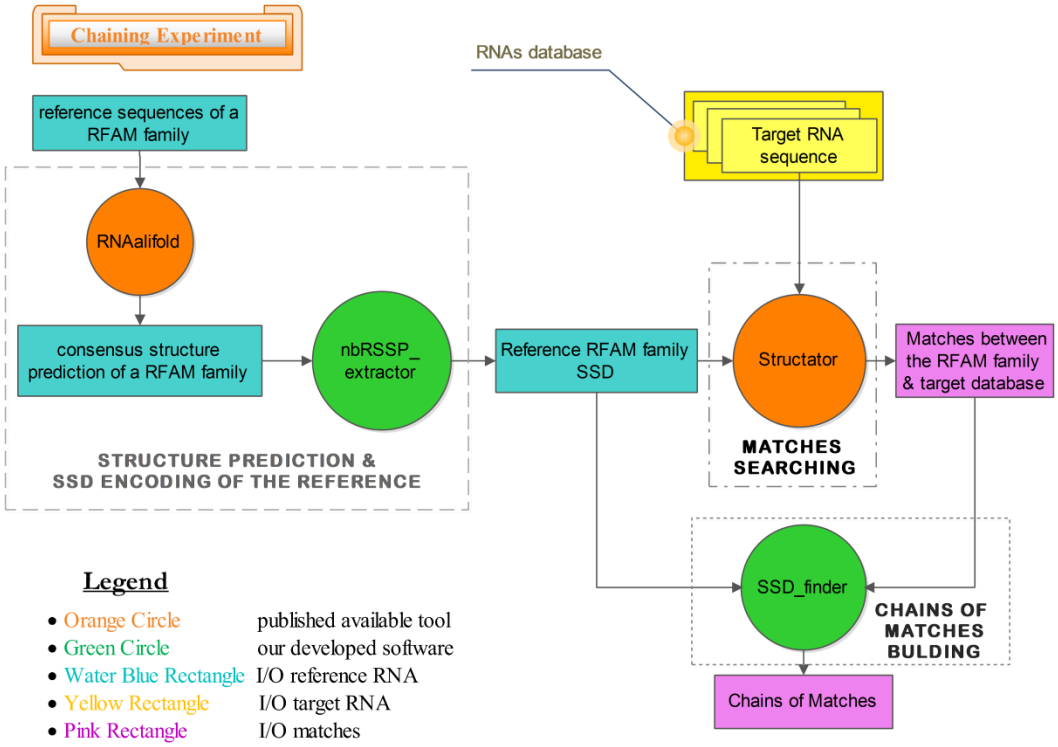

Figure 6 Workflow of the chaining experiment. The diagram depicts the pipeline applied to four selected RFAM families to test the performance of our chaining algorithm (*SSD_finder*). A full explanation of that is given in the subsection *SSD_finder validation* of the section *Results and Discussion*.

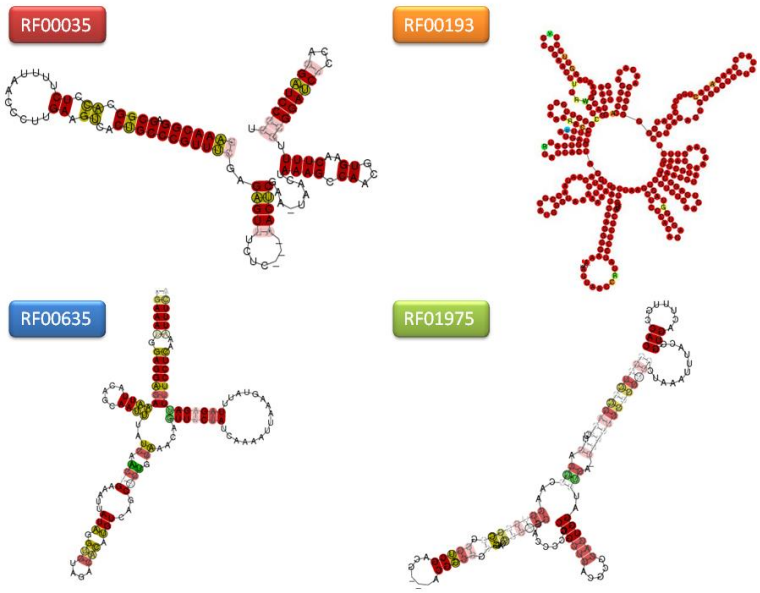

Figure 7 Consensus structures of the four selected RFAM families. The consensus secondary structures predicted by *RNAalifold* for each family (RF00035, RF00193, RF0635, and RF1975) are shown. These families are used as the starting point of the chaining experiment described in Figure 6.

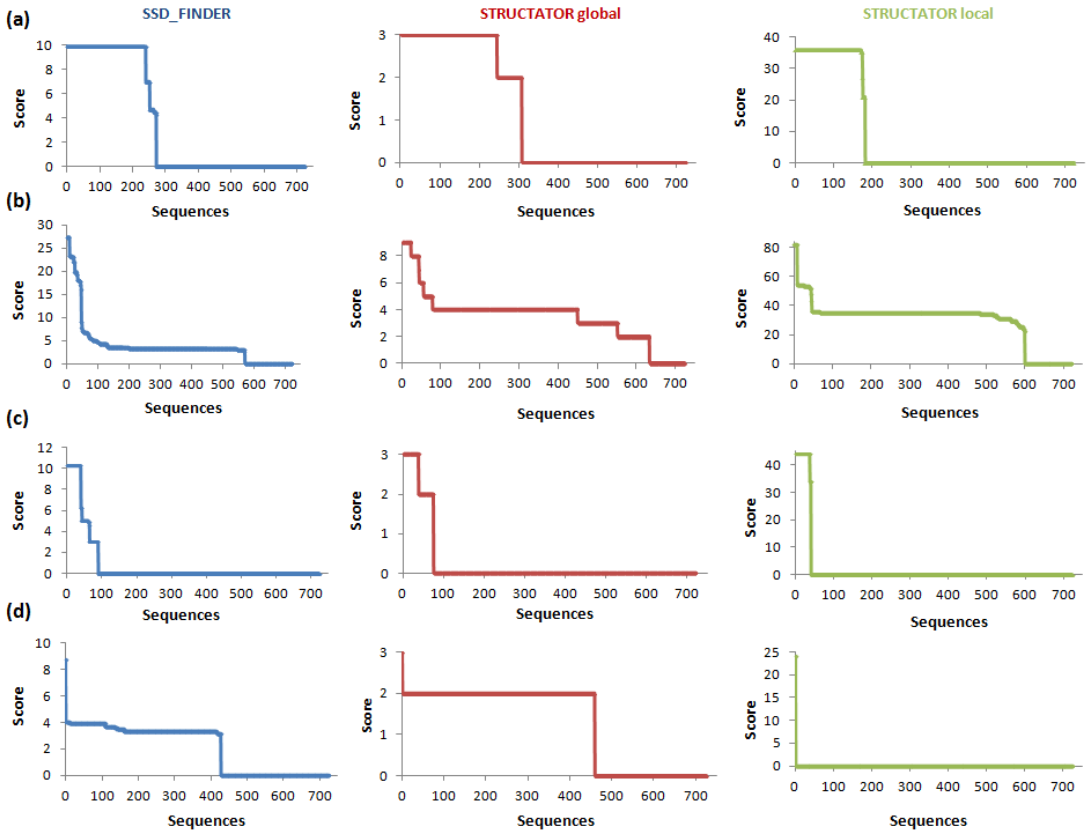

Figure 8 Chain scores evaluated from different tools. Each panel represents the efficiency of different tools (from left to right: *SSD_finder*, *Structator global*, *Structator local*) in the classification of the members of the four selected families depicted in Figure 7. Blue lines represent the score of our algorithm, *SSD_finder*, evaluated as in equation (1) in the text; red lines represent the global chain score of Structator (*gcsc*) evaluated as in equation (2) of the original paper **[26]**; green lines represent the local chain score of Structator (*lcsc*) evaluated as in equation (3) of the original paper **[26]**. The x axis represents the number of RNA sequences that constitute the database used as target in the chaining experiment (Figure 6). This database includes the four selected families (Figure 7) and a subset of families randomly extracted from the RFAM and RNAstrand databases (more than 700 sequences in total).

Table 1 RNAs folding prediction

Category	Tool	Input	Input format	Output	Description	Publication (year)
Thermodynamic Models	mfold	Single sequence ≤ 800 nt and $\leq$ 9000 nt for batch	FASTA	mfe RNA secondary structure prediction	• Dynamic programming method • Updated and renamed UNAfold [39] • Web server/standalone • Open source	1989 [40, 41, 42]
	RNA fold	Single sequence $\leq 10^4$ nt	FASTA	mfe RNA secondary structure (dot-bracket)	• Dynamic programming method • Web-server/standalone • Open source	1994 [43, 25]
	RNA subopt	• Small sequence • Minimum energy threshold (met)	FASTA	sub-optimal structures (dot-bracket) with energy $\leq$ met	• Output grows exponentially with both sequence length and energy range • Standalone • Open source	1999 [44]
	RNAL fold	• Long sequence • Base-pair span of L	FASTA	Locally stable structures with its energy and the starting position of the local structure	• It handles huge databases • Standalone • Open source	2004 [24]
	Sfold	Single sequence ≤ 200 nt and $\leq$ 5000 nt for batch	FASTA Plain text GenBank	Ensemble of possible structures sampled from the Boltzmann probability-weighted structures • Comparison between mfe structure prediction and ensemble centroid • Cluster members and distances inter/intra clusters	• K-means (using nt distances) and Calinski Harabasz- index • Evaluation of RNA/RNA interactions • 4 modules: sRNA (general folding features and output), siRNA (short-interfering RNAs), Soligo (antisense oligonucleotides), Sribo (ribozymes) • Web-server/ standalone • Proprietary	2003 [45]
	RNA shapes	Sequences<400 nt	FASTA	• mfe structure representative of each *abstract shape* (disjoint classes with common structures) and their probability • Consensus structures	• Folding space partitioned in different *abstract shapes* • Running time grows exp with seq length • Web server /standalone • Open source	2006 [46]
	Rfold	• Single sequence • Base-pair span of L	FASTA	Local secondary structures based on the local bpp	• Dynamic programming method • 40 times slower than RNALfold [47] • Standalone • Open source	2008 [48]
	RNA structure	Single sequence (<2500nt for web server)	FASTA Seq	• mfe structures and bpp • mea	• Includes pseudoknots • Web server /standalone • Open source	2010 [49]
Machine Learning algorithm	CONTRA fold	Single sequence < 1000 nt with optional structural annotations	FASTA BPSEQ plain text	Secondary structure prediction according to the conditional log-linear models	• Use statistical learning algorithms to derive model parameters • Not as accurate as the biophysics models • Web-server/standalone • Open source	2006 [50]
Kinetics Folding	Kinfold	Single sequence	FASTA	• Secondary structure considering the kinetic transcriptional parameters • Best fitting kinetic model among the all possible ones	• Monte Carlo stochastic simulation of folding model as Markov process • Running time grows exp. with seq length • Standalone • Open source	2000 [51]
	Kinefold	Small sequence <400 nt, helix < 60 base-pairs	String of bases	• Animated folding path • Programmable trajectory plot focusing on a few helices of interest	• Stochastic folding simulations • It includes pseudo-knots • Computationally onerous • Web-server/standalone • Open source	2003 [52, 53]
	Kinwalker	Single sequence (<1500 nt)	Not specified	Mfe secondary structures prediction at each step of the transcription	• Not including pseudoknots • Standalone • Open source	2008 [54]
	CoFold	Single sequence	FASTA plain text	Secondary structures considering the co transcriptional folding and thermodynamic parameters	• the accuracy increases with seq length • Web-server/Standalone • Open source	2013 [55]
Phylo-genetic	PETfold	Multiple sequence alignments	FASTA	• Secondary structures (score and reliability) • Consesus structures	• Finds bp more likely to be evolutionary conserved and energetically favored • Web server/standalone • Open source	2008 [56]

Table 2. RNAs folding and structure conservation

Category		Tool	Input	Input format	Output	Description	Publication (year)
Folding and Evaluation of Structure Conservation	Conservation and Thermodynamic stability	QRNA	Pair-wise sequences alignment	MFASTA (FASTA with gaps)	• RNAs label as coding or non-coding • Conserved structures information	• Uses comparative sequence analysis • Uses SCFGs to estimate a structure probability distribution • Uses pair hidden Markov model to predict evolutionarily conserved structure • Standalone • Open source	2001 [57]
		RNAalifold	Sequences alignments	ClustalW FASTA	Common mfe structures to the most of folded sequences	• Web-server/standalone • Open source	2002 [58, 36]
		Alifoldz	Multiple sequence alignments based on the thermodynamic model provided by RNAalifold	ClustalW FASTA	• Conserved structures • Z-score (thermodynamic stability index)	• Compares mfe consensus structure given by RNAalifold with one obtained by a randomized alignment • Standalone • Open source	2004 [59]
		RNAz	Sequences alignment	ClustalWF ASTA Phylip Nexus Maf Xmfa	Thermodynamically stable and conserved structures using SCI (structure conservation index) and z-score	• Uses the prediction of the consensus structure given by RNAalifold • Web-server/standalone • Open source	2005 [60]
		Evofold	Multiple Sequences alignment	Newick Ama	Structure of a multiple alignment regarding the probabilistic evolutionary model (phylo-SCFG)	• Overlap in true positives with all of the thermodynamic-only tools • Standalone • Proprietary	2006 [61]
	Conservation and Covariance	ddbRNA	Multiple or pair-wise sequences alignments ($\leq$3-way alignments)	MFASTA	Secondary structure prediction through the covariance model	• Counts the compensatory mutations of the alignment as a measure of the structure conservation with respect to a randomized alignment • High sensitive to the alignment quality • Running time $\propto$ square sequence length • Performance worse than tools using SCI • Standalone • Proprietary	2003 [62]
		MSARi	Multiple or pair-wise sequences alignments (10-15-way alignment)	ClustalW	Secondary structure prediction through the covariance model (stack of compensatory mutations needed to keep the secondary structure functionality)	• Evaluates the statistical significance of the short and contiguous regions of potential pairing, regarding different distribution models • Uses RNAfold to predict the bpp and analyzes the base-pairs in a window of 7nt looking for the compensatory mutations • Analyzes each base-pairs with probability >5% • Standalone • Proprietary	2004 [63]
	Other	TRANSAT	• Multiple sequence alignments • Related sequences tree	Not specified	• Prediction of structural features including transient, pseudo-knotted and alternative structures • Reliability estimation of all the predictions	• Sensitive to the alignment quality • Considers evolutionarily related RNA sequences from different organisms • Standalone • Proprietary	2010 [64]

Table 3 RNAs comparison: Sequence-structure alignment

Category	Tool	Input	Input format	Output	Description	Publication (year)
Sequence-structure alignment	MARNA	RNA sequences and their secondary structures	FASTA Dot/bracket	Multiple sequence-structure alignment	• Not maintained since 2005 (replaced by locARNA) • Examines only partially conserved structures • Web-server/standalone • Open source	2003 [65]
	locARNA	RNA sequences (<2000 nt for an interactive job)	FASTA	Global or local pair-wise alignment regarding the structure information	• Variant of Sankoff's algorithm for simultaneous folding and alignment • Folding RNAfold • mlocARNA computes a multiple alignment to give as input to RNAalifold • Computationally expensive • Web-server/standalone • Open source	2007 [66]
	expaRNA	Two long/small ncRNAs to compare with/without pre-defined structures, using mfe structures	FASTA	• Pair-wise sequence-structure alignment • Common sub-structures to two RNAs	• Uses the predicted sequence structure motifs as anchor points for the whole alignment • Algorithm accuracy related to considered sequence-structure motifs • Speed up state-of-the-art alignment methods • Web-server/standalone • Open source	2009 [67]
	RNAG	Set of RNA sequences not aligned	FASTA	• Alignment • Prediction of the consensus structure	• Blocked Gibbs sampling algorithm • Iteratively samples from the cpd P(Structure|Alignment) and P(Alignment | Structure) improving the alignment and structure models • Uses the Markov chains Monte Carlo method • Web-server/standalone • Open source	2011 [68]
	Carna	• Sequences and structures based on the bpp • Structure constraints	FASTA Dot/bracket	Multiple alignments of RNA with different conserved structures or of whole set of structures	• Able to align also pseudo-knots • Optimizes all the structural similarity of input RNA • Performance as good as the current alignment tools • Web server • Proprietary	2012 [69]
	Graph-clust	Set of not aligned lncRNA sequences	FASTA	Clustering: divides the RNAs into classes, each one characterized by RNA of similar structure and function	• Linear-time prediction of local structural elements • Folding with RNAshapes to obtain sub-optimal structure representing each "shape" • Sequences-structures alignment with locARNA and Infernal used as a feedback control • Web-server/standalone • Open source	2012 [70]
	Migal	Two RNA structure-sequences	Dot/ bracket Bpseq Migal Xml	• Two sequence alignment • Number of mismatches, insertions and deletions	• 4 layers representation of the secondary structure coded by a rooted orderer labelled tree (Level 0 multiloop network, Level 1 stems network, Level 2, helices network, Level 3 base paired and unpaired) • Edition algorithm: insertion, deletion, substitution plus node fusion/split edges fusion/split • Web-server/standalone • Open source	2008 [71, 72]
	Gardenia	Set of RNA sequences with their secondary structures	FASTA Dot/ Bracket	Multiple sequence alignment regarding sequence and structure	• Edit operations concerning free bases and concerning arcs between bases • Not including pseudoknots • Web-server/standalone • Open source	2008 [73]

Table 4. RNAs motifs searching tools

Category	Tool	Input	Input format	Output	Description	Publication (year)
Structure-sequence descriptors - based	RNAmotif	• Descriptor file:specifies structure to look for • Target sequences	Text file FASTA	• Matched sub-sequences • Match location • Score based on structural constraints	• Computes the thermodynamic stability score of the candidates structure and classifies the free energy • Motif descriptor and scoring system not enough suitable • Standalone • Open source	2001 [74]
	Locomotif	Small ncRNAs sequences	Graphical motif description languange	RNA motifs matchings	• Dynamic programming method who includes the structure thermodynamic model • Standalone • Proprietary	2007 [75]
	STRMS (structural RNA motif search)	• Query sequence or structures (including structural constraints) • Target sequence database	FASTA	All occurrences of the query in the target	• Tree representation and dynamic programming • Based on subtree homeomorphism for ordered, rooted tree • Pre-folding, partitioning the target sequence into consecutive overlapping windows, folding them and converting each structure to a tree representation • Includes pseudoknots • Standalone • Open source	2007 [76]
	Motifs-search	• SmallRNA sequence with known structure • Target sequence	FASTA	Structural homologies	• Tree representation of the secondary structures • Searches all potential stem-loops similar to ones of the given RNA secondary structure • Based on located stem-loops detects potential homologous structural RNAs in genomic sequences • Standalone • Proprietary	2007 [77]
	Structator	• Target sequence • SSD (set of RSSP to describe the query global or local structure)	FASTA Dot/bracket	• Matches • Match chains	• Nested and non-braching structure • Supports wide variety of pattern characterized by the wildcards nt and with stem-loop of variable length • Employs an innovative index-based bidirectional matching algorithm • Running time scales sub-linearly with the length of the searched sequences • Two programs: *Afconstruct* for the construction of the affix-array; *Afsearch* allows users to find all the possible matches with the pattern of RNA sequence-structure • Standalone • Open source	2011 [26]
Search for common motifs without descriptors	RNA profile	• Number of hairpins (h) in the motifs • Set of not aligned smallRNA sequences	Dot/bracket FASTA	Most conserved regions with respect to sequence and structure according to base-pairing and thermodynamic rules	• Greedy algorithm :generate a set of candidate regions whose mfe structure contains exactly h hairpins and compares the regions selected with each other to find the groups of most similar ones • Free-alignment method • Feasible computational complexity • Standalone • Open source	2004 [78]
	Infernal	• Multiple RNA alignment • Target sequence to look for	Stockholm	Statistical scoring system: quantitative ranking of the homologies in a sequence database	• Use covariance model to searching RNA sequence databases for RNA structure and sequence similarity • Standalone • Open source	2009 [79]

Table 5 Results of structure predictions performances for different tools.

RNAs	**Base-pairs in the known structure (P)**	*RNALfold/ nbRSSP_extractor*		*Rfold*		*RNAfold*		*RNALfold_lnrz*	
		True Positive (TP)	**False Positive (FP)**	**True Positive (TP)**	**False Positive (FP)**	**True Positive (TP)**	**False Positive (FP)**	**True Positive (TP)**	**False Positive (FP)**
rRNA 16 S	218195	117950	153933	103811	104835	100955	90620	98922	111742
rRNA 23 S	104340	60092	74191	49130	44922	51680	39778	48797	53531

Table 6 Results of RFAM families detection for different tools

RFAM reference family (# of RSSPs)	*# of detected members/ total family members*			*# of sequences taken to cover the all family members*		
	SSD_finder	**Structator global**	**Structator local**	**SSD_finder**	**Structator global**	**Structator local**
RF00035 (4 RSSPs)	266/300	268/300	171/300	271	307	175
RF00193 (11 RSSPs)	44/44	44/44	44/44	44	44	47
RF00635 (3 RSSPs)	60/66	60/66	41/66	60	60	41
RF01975 (4 RSSPs)	54/65	54/65	1/65	136	459	1