

RESEARCH

Open Access

LocARNAscan: Incorporating thermodynamic stability in sequence and structure-based RNA homology search

Sebastian Will^{1,2†}, Michael F Siebauer^{3†}, Steffen Heyne², Jan Engelhardt¹, Peter F Stadler^{1,5,6,7,8*}, Kristin Reiche^{1,4,5*} and Rolf Backofen^{2,7*}

Abstract

Background: The search for distant homologs has become an import issue in genome annotation. A particular difficulty is posed by divergent homologs that have lost recognizable sequence similarity. This same problem also arises in the recognition of novel members of large classes of RNAs such as snoRNAs or microRNAs that consist of families unrelated by common descent. Current homology search tools for structured RNAs are either based entirely on sequence similarity (such as blast or hmmer) or combine sequence and secondary structure. The most prominent example of the latter class of tools is Infernal. Alternatives are descriptor-based methods. In most practical applications published to-date, however, the information contained in covariance models or manually prescribed search patterns is dominated by sequence information. Here we ask two related questions: (1) Is secondary structure alone informative for homology search and the detection of novel members of RNA classes? (2) To what extent is the thermodynamic propensity of the target sequence to fold into the correct secondary structure helpful for this task?

Results: Sequence-structure alignment can be used as an alternative search strategy. In this scenario, the query consists of a base pairing probability matrix, which can be derived either from a single sequence or from a multiple alignment representing a set of known representatives. Sequence information can be optionally added to the query. The target sequence is pre-processed to obtain local base pairing probabilities. As a search engine we devised a semi-global scanning variant of LocARNA's algorithm for sequence-structure alignment. The LocARNAscan tool is optimized for speed and low memory consumption. In benchmarking experiments on artificial data we observe that the inclusion of thermodynamic stability is helpful, albeit only in a regime of extremely low sequence information in the query. We observe, furthermore, that the sensitivity is bounded in particular by the limited accuracy of the predicted local structures of the target sequence.

Conclusions: Although we demonstrate that a purely structure-based homology search is feasible in principle, it is unlikely to outperform tools such as Infernal in most application scenarios, where a substantial amount of sequence information is typically available. The LocARNAscan approach will profit, however, from high throughput methods to determine RNA secondary structure. In transcriptome-wide applications, such methods will provide accurate structure annotations on the target side.

Availability: Source code of the free software LocARNAscan 1.0 and supplementary data are available at <http://www.bioinf.uni-leipzig.de/Software/LocARNAscan>.

*Correspondence: studla@bioinf.uni-leipzig.de;
kristin.reiche@izi.fraunhofer.de; backofen@informatik.uni-freiburg.de

[†]Equal contributors

¹Bioinformatics Group, Department of Computer Science, and Interdisciplinary Center for Bioinformatics, University of Leipzig, Härtelstraße 16-18, Leipzig D-04107, Germany

⁵RNomics Group, Fraunhofer Institute for Cell Therapy and Immunology, Perlickstraße 1, Leipzig D-04103, Germany

Full list of author information is available at the end of the article

Background

Over the last decade, a series of large-scale transcriptome projects has profoundly changed our perception of the transcriptome. Reviewed e.g. in [1], pervasive transcription is widespread and plays a crucial role in controlling gene expression and genomic plasticity. Gene prediction and gene annotation of non-protein coding entities have remained non-trivial problems, nevertheless. In part, this is due to our incomplete understanding of the diversity of ncRNAs, of which novel types and subtypes keep being discovered at a rapid pace. An important confounding factor, however, is the rapid evolution of many ncRNA sequences [2-4], which intrinsically limits the applicability of homology search methods [5,6] and hence hide distant homologs.

The three-dimensional structure is important for the functionality and/or the proper processing of a large and important subgroup of ncRNAs. The most prominent representatives are ribosomal RNAs (rRNAs), transfer RNAs (tRNAs), spliceosomal RNAs (snRNAs), small nucleolar RNAs (snoRNAs), and microRNAs (miRNAs). While rRNAs and tRNAs are among the best-conserved sequences also at sequence level, other classes such as C/D box and H/ACA box snoRNAs exhibit sometimes very large substitution rates. The conservation of spatial structure implies that secondary structure, i.e., base pairing patterns, are also under stabilizing selection. In many cases, the structure evolves much slower than the sequence, see [7] for a recent detailed analysis of this phenomenon. Thus, several computational tools have been devised to utilize secondary structure alongside with sequence information for homology search. The same effect is exploited by tools such as RNAz [8], Evofold [9], or SISSIz [10] that detect selection pressure on RNA secondary structure in multiple sequence alignments.

Structural similarity is either inherited from a common ancestor or arises by convergent evolution as the result of similar selective constraints. Operationally, we distinguish *RNA families* and *RNA classes*. The members of RNA families share a sufficiently high level of sequence similarity to establish the existence of a common ancestor, which in practice translates to the possibility of representing them as structure-annotated multiple sequence alignment. The Rfam database [11] serves as a comprehensive repository for this type of data. Representatives of RNA classes share secondary structures (e.g. as a consequence of a common processing pathway in the case of microRNA precursors) or a combination of sequence and structure features (e.g. as a consequence of being incorporated into analogous ribonucleoproteins in the case of snoRNAs).

Homology search programs are geared towards detecting novel members of known RNA families, reviewed e.g. in [12]. The most commonly used tool

Infernal [13] uses covariance models (CMs), i.e., the stochastic context free grammar analogue of profile hidden Markov models. Similar to search heuristics such as Erpin [14], the CMs are trained from sequence alignments that are annotated by a consensus secondary structure. When a lack of known examples precludes the construction of consensus models, tools such as RSEARCH [15] and BlastR [16] allow single structure-annotated or even unstructured RNAs as queries. A common feature of all these methods is that they heavily (or even exclusively) rely on the sequence information contained in the query model, and that they evaluate whether a piece of the target sequence *can* be folded to match a prescribed query structure. Consistency with the query structure, however, does not necessarily imply that a putative homolog is thermodynamically predisposed to actually fold into this structure. Whether the query structure is close to the target's groundstate or whether it is an unfavourable high energy structure, therefore can provide additional information to improve specificity.

The members of *RNA classes*, on the other hand, share structural features and some sequence motifs deriving from common binding partners and functions. The term *RNA clan* [17] has been proposed for RNAs that derive from a common ancestor but have diverged far enough to either be difficult to align or have distinct functions, or both. As the distinction between clans and classes requires detailed knowledge of their evolutionary histories, we will not distinguish between clans and classes in this contributions. Examples of RNA classes are animal microRNAs (featuring a characteristic precursor hairpin and processing pattern) or the two distinct classes small nucleolar RNAs (snoRNAs) defined by the C/D and H/ACA "boxes" (short common sequence motifs) and very different characteristic secondary structures. These three paradigmatic RNA classes each comprise large numbers of families. Computational surveys [18-21], furthermore, gathered convincing evidence for large numbers of conserved RNA structures; subjecting these data sets to structure-based clustering suggested the existence of additional, so far undescribed RNA classes [22].

There are mainly two approaches for the search for novel members of RNA classes. One class consists of descriptor-based methods [23-25] or with the help of class-specific tools that combine an efficient filtering step with elaborate, often machine-learning based, evaluation procedure that ensure the required specificity [26]. In either case an in-depth knowledge of the RNA class under consideration is required. The other class does not require in-depth information on the RNA-class and its characteristic patterns of conservation. Instead it exploits information on sequential and structural similarity directly

by using sequence-structure alignment. The first practical approaches for multiple structural alignment, such as RNAforester [27] and MARNA [28], depend on predicted or known secondary structures. In practice, however, these approaches are limited by the low accuracy of non-comparative structure prediction. For this reason, several derivatives of the Sankoff-based approach [29] of simultaneous alignment and folding have been introduced. In approaches such as FoldAlign [30,31], Dynalign [32], and Stemloc-AMA [33] a full energy model for RNA is implemented that is evaluated during the alignment computation. However, in its full form, these approaches suffer from a high worst case computational complexity of $\mathcal{O}(n^6)$ time and $\mathcal{O}(n^4)$ space. In contrast, PMcomp [34] and LocARNA [22] use a full-featured energy model in a precomputation step to determine a reduced representation of the structure ensemble in form of base pair probability matrices [35]. During the alignment process, base pair probabilities are used to assess the similarity of the secondary structures. Using additional computational optimizations, the complexity of LocARNA could be reduced to quartic time and quadratic memory consumption, making it currently one of the most efficient versions of the Sankoff algorithm. Several improvements and extensions of LocARNA have been discussed before: to additionally reduce LocARNA's runtime, ExpaRNA-P [36,37] utilizes a fast structural filtering method based on local structural motifs [38,39]; REAPR [40] introduces a multiple alignment-based banding method to realign eukaryotic whole genome alignments based on RNA structure; recently, [41] introduces the very efficient LocARNA descendant SPARSE; and LocARNA-P [42] extends LocARNA by computing reliabilities, thus enabling new applications of Sankoff-style alignment. None of these approaches, however, addressed efficient scanning.

In practice, homology search relies predominantly on the sequence information in the query. Even in the CMs representing the heavily structured Rfam alignments sequence information by far outweighs the additional bit score contributed by the consensus secondary structure ([43], Figure one point nine). Indeed, for most Rfam families, the structural information is well below the 20 bits that would be required to push the E -value below 1 in a small, 1M bacterial genome in a structure-based search that completely ignored sequence conservation.

CM methods ask how well the target sequence matches the query's sequence and structure model that is derived from the input alignment and its provided consensus structure. However, they do not take into account the thermodynamic stability of the target sequence folding into the consensus structure. This could provide an additional source of information on the structural concordance of query and target.

In principle, this information is accessible in two quite different ways. Following the philosophy of Thermodynamic Matchers [25] and of the structure conservation index [44], one can interpret the consensus structure as constraint and evaluate the energy difference between constrained and unconstrained folding. The alternative is to predict structures also for the target.

A pilot study [45] provided first indications that structural alignments could be used for genome-wide homology search in a regime where sequence information is scarce. In an approach to find class-members, a model specified as base pairing probability matrix, possibly augmented with some additional sequence information, is searched against a target for which local base pairing probabilities are provided. To this end, base pairing probability matrices were computed with McCaskill's partition function folding algorithm [35]. These were then aligned locally with LocARNA [22]. Here we describe and evaluate an optimized semi-global scanning variant of the LocARNA algorithm that can be employed in genome-wide applications.

The LocARNAscan algorithm

LocARNA is a computationally light-weight and very efficient variant of the Sankoff algorithm [29]. It improves the CPU and memory requirements each by a quadratic factor over the original algorithm. For this purpose, LocARNA allows for matches of base pairs that occur with a given minimum probability in the structure ensembles of the single input sequences.

Here we devise a scanning variant of LocARNA, called LocARNAscan, that computes alignments of a *query* RNA with a much longer *target* sequence based on sequence and structure similarity. In our discussion, we require such alignments to be *semi-global*; in such an alignment the entire query is compared to a subsequence of the target. We achieve this by allowing free end gaps, i.e. arbitrary long deletions at both ends of the target. This semi-global scenario is designed for the common case of known motif boundaries in the query. Nevertheless, our method can be adapted to align locally with respect to both target and query as long as the locality is of biological origin. Technical locality, which was introduced in the trCYK algorithm [46] as a means of dealing with partial sequences from RNA-seq data, cannot easily be integrated, however.

We provide a generic description of the algorithm, where the target is a single sequence and, in general, the query is a multiple alignment. Furthermore, both, query and target are annotated with a base pair probability matrix, which works similarly to a weighted contact map. In general, such matrices can be computed by McCaskill's algorithm [35]. In the common case of known query structure, we generate an appropriate base pair probability matrix by setting the probabilities of the query structure

base pairs to 1 (and all others to 0). For the target, we suggest to compute local base pair probabilities by RNAplfold [45,47], which limits the span $(j - i)$ of base pairs (i, j) . Thus, we assume a maximum span $L \ll n$ of target base pairs that we consider for the comparison to the query. This restriction of the base pair span serves two purposes. First, the restriction allows predicting the base pair probabilities much more efficiently. Second, RNA structure prediction methods generally tend to mispredict large base pairs. As a consequence, the accuracy is usually even increased by limiting the size of base pairs (cf. [48].)

Notation and scoring model

Both, target T and query Q are sequences of nucleotides or, in the case of multiple alignment, alignment columns with lengths n and m , respectively. The sequences are annotated with respective substochastic matrices $P^T \in [0, 1]^{n \times n}$ and $P^Q \in [0, 1]^{m \times m}$. We write $(i, j) \in P^X$ as shorthand for $P^X(i, j) \geq p_{\min}(X \in \{T, Q\})$, where p_{\min} is a fixed cutoff probability. Note that for each fixed i , the number of j satisfying $(i, j) \in P^X$ is constantly bounded, since $\sum_j P^X(i, j) \leq 1$ (cf. LocARNA [22]).

An alignment \mathcal{A} of T and Q is a set of pairs (i, j) of indices i of T and j of Q , where all pairs $(i, j), (i', j') \in \mathcal{A}$ satisfy $(i < i' \text{ iff } j < j')$ and $(i = i' \text{ iff } j = j')$.

A *secondary structure of length n* is a set R of base pairs (i, j) with $1 \leq i < j \leq n$, where base pairs do not cross, i.e. no two base pairs $(i, j) \in R$ and $(i', j') \in R$ satisfy $i < i' < j \leq j'$ or $i \leq i' < j < j'$.

Together with an alignment \mathcal{A} of target and query, we are going to predict a consensus structure \mathcal{S} . A consensus structure is a set of pairs of base pairs $((i, j), (k, l))$, where the set of the respective first and second components are secondary structures of T and Q ; it is *consistent with an alignment \mathcal{A}* iff for all $((i, j), (k, l)), (i, k) \in \mathcal{A}$ and $(j, l) \in \mathcal{A}$.

The score of a consistent pair $(\mathcal{A}, \mathcal{S})$, which represents a *sequence-structure alignment*, is of the form

$$\sum_{((i,j),(k,l)) \in \mathcal{S}} \left[\Psi_{(i,j)}^T + \Psi_{(k,l)}^Q + \tau(T_i, T_j; Q_k, Q_l) \right] + \sum_{(i,k) \in \mathcal{A}^s} \sigma(T_i, Q_k) + \gamma N_{\text{gap}}. \quad (1)$$

Here, \mathcal{A}^s denotes the set of single-stranded alignment edges $\mathcal{A} \setminus \{(i, k), (j, l) \mid ((i, j), (k, l)) \in \mathcal{S}\}$, i.e. the alignment \mathcal{A} without all edges that match base pairs in \mathcal{S} . Furthermore, $\gamma < 0$ is the gap penalty and N_{gap} counts the scored gaps in \mathcal{A} . Recall that for semi-global alignment, we allow free, non-scored, deletions at both ends of the target; thus, for the semi-global score

$$N_{\text{gap}} = \max\{i \mid (i, k) \in \mathcal{A}\} - \min\{i \mid (i, k) \in \mathcal{A}\} + 1 + m - 2|\mathcal{A}|; \quad (2)$$

the global score is of the same form of Equation (1), where $N_{\text{gap}} = n + m - 2|\mathcal{A}|$. The functions $\sigma(T_i, Q_k)$ and $\tau(T_i, T_j; Q_k, Q_l)$ yield sequence-based similarities between elements of target and query in the unpaired and paired part of the consensus structure, respectively. We are going to discuss their instantiations later. The explicit dependence of τ on the aligned base pairs can be used to include contributions based on covariation or substitution. Finally, $\Psi_a^T + \Psi_b^Q$ scores the structural contribution of the consensus base pair $(a, b) \in \mathcal{S}$. As in LocARNA, Ψ_a^X ($X \in \{Q, T\}$) is derived as log odd from the base pair probability $P^X(a)$, where we set $\Psi^X(a) := -\infty$ if $P^X(a) < p_{\min}$ to rule out base pairs with very low probability in (finitely scoring) consensus structures.

For simplicity, we present this score for linear (non-affine) and position independent gap cost, since the extensions of the presented method are straightforward.

Finally, we define the *subscore for $i \dots j$ and $k \dots l$* (denoting respective subsequences of T and Q) to be of the form of Equation (1), but – unlike the total score – valid only for alignments $\mathcal{A} \subset \{i, \dots, j\} \times \{k, \dots, l\}$, where furthermore N_{gap} is defined as

$$j - i + 1 + l - k + 1 - 2|\mathcal{A}|,$$

such that the subscore penalizes deletions at both ends of the target subsequence $i \dots j$.

Finally, we define the analogous subscore *with free left end deletions* by defining N_{gap} as

$$j - \min\{i \mid (i, k) \in \mathcal{A}\} + 1 + l - k + 1 - 2|\mathcal{A}|.$$

Dynamic programming recursions

For maximizing the semi-global score of $(\mathcal{A}, \mathcal{S})$, we start by introducing dynamic programming matrices S and D in analogy to the dynamic programming matrices of LocARNA [22] and PMcomp [34]. Thus, we define the entry $S_{i,j;k,l}$ as the best subscore for $i \dots j$ and $k \dots l$. Furthermore, let $D_{i,j;k,l}$ be the best subscore for $i \dots j$ and $k \dots l$ of an alignment and consensus structure \mathcal{S} subject to the constraint that \mathcal{S} contains $((i, j), (k, l))$.

Equivalently, we redefine S and D recursively by

$$S_{i,j;k,l} = \max \begin{cases} S_{i+1,j;k,l} + \gamma, \\ S_{i,j;k+1,l} + \gamma, \\ S_{i+1,j;k+1,l} + \sigma(T_i, Q_k), \\ \max_{\substack{(i,j') \in P^T \\ (k,l') \in P^Q}} (D_{i,j';k,l'} + S_{j'+1,j;l'+1,l}) \end{cases} \quad (3)$$

$$D_{i,j;k,l} = S_{i+1,j-1,k+1,l-1} + \Psi_{(i,j)}^T + \Psi_{(k,l)}^Q + \tau(T_i, T_j; Q_k, Q_l) \quad (4)$$

with appropriate initializations

$$S_{j+1,j;k,l} = \gamma(l - k) \text{ and } S_{i,j;l+1,l} = \gamma(j - i) \quad (5)$$

for $1 \leq i < j \leq n$ and $1 \leq k < l \leq m$.

Since these equations are left reducing variants of the otherwise well known recursions of PMcomp [34], one computes all entries in D in $\mathcal{O}(n^2m^2)$ time and $\mathcal{O}(nm)$ space by applying the evaluation strategy of LocARNA, which takes advantage of filtering base pairs by probability p_{\min} . However, note that the dependency on the target size n is still unacceptable for our purpose (of scanning genome-sized targets); therefore, we are going to improve on these complexities later.

The matrices S and D are defined to compute global sub-scores (i.e., without free end gaps). Since we finally need alignments with free end gaps, we have to introduce an additional DP matrix S^* . Furthermore, since we need the semi-global alignment score only for the entire target and query, it suffices to define the matrix only for scores of target and query prefixes. Thus, define $S_{j,l}^*$ non-recursively as the best sub-score for $1 \dots j$ and $1 \dots l$ with free left end deletions. This is equivalent to the recursion

$$S_{j,l}^* = \max \begin{cases} S_{j-1,l}^* + \gamma, \\ S_{j,l-1}^* + \gamma, \\ S_{j-1,l-1}^* + \sigma(T_j, Q_l), \\ \max_{\substack{(i',j) \in P^T \\ (k',l) \in P^Q}} (S_{i'-1,k'-1}^* + D_{i',j;k',l}), \end{cases} \quad (6)$$

where the free end gaps are achieved purely by the specific initialization

$$S_{j,0}^* = 0 \quad S_{0,l}^* = \gamma l \quad (7)$$

for $0 \leq j \leq n$ and $0 \leq l \leq m$.

Efficient evaluation of the recursion equations

As mentioned before, the computational demands of the straightforward evaluation of these recursions are unacceptable for scanning. This holds even when applying the evaluation strategy of LocARNA, which never stores more than $\mathcal{O}(nm)$ S entries at each time. Although, as suggested before, we generally limit the maximum base pair span to L , this leaves us with prohibitive linear space dependency on the target size, since $\mathcal{O}(nm)$ entries of D have to be kept in memory before the S^* recursion can be evaluated. Furthermore, storing the entire matrix S^* requires $\mathcal{O}(nm)$ space.

Therefore, we rearrange the evaluation interleaving the computation of entries in D and S^* . This allows us to compute all S^* entries in one pass over the target sequence, while successively materializing required matrix entries and in turn removing “old” entries that are not accessed anymore (Algorithm 1, cf. Figure 1).

Computing and storing the D entries in lines 3–5 requires to compute and store the subset of S entries that allows to derive $D(i, j, k, l)$ efficiently from $S(i + 1, j - 1, k + 1, l - 1)$. Formulating recursion (3) recursing only to entries with the same fixed right ends j and l enables

Algorithm 1: LocARNAscan algorithm

```

1 for j := 1 to n do
2   for l := 1 to m do
3     foreach (i, j) ∈ PT, (k, l) ∈ PQ do
4       compute and store D(i, j, k, l);
5     end
6     compute and store S*(j, l);
7     if j > L then
8       forget D(j - L, j', l, l') for all j', l';
9       forget S*(j - L, l);
10    end
11  end
12 end
    
```

an important optimization: for each j and l , we compute a matrix slice of entries in S that have right ends $j - 1$ and $l - 1$ and then derive all $D(i, j, k, l)$ from this matrix slice. Note that we need to compute and store at most $\min(j - 1, L) \times (l - 1)$ such entries, but potentially less depending on the actual base pairs $(i, j) \in P^T$ and $(k, l) \in P^Q$. After the D entries are derived, these S entries are not accessed anymore and their space can be reused.

In lines 8 and 9, we free the space of all D and S^* entries that cannot be accessed by subsequent algorithm steps anymore. This is guaranteed for all entries with target left end $i \leq j - L$, since no base pair spans more than L positions. Consequently, the space requirements of this algorithm are bounded by the requirement to store Lm entries in slices of S , Lm entries of S^* , and $\mathcal{O}(Lm)$ entries of D , i.e., $\mathcal{O}(Lm)$ space in total.

Allocating and freeing entries can be implemented in constant time by using rotating matrices, commonly implemented by addressing based on target indices modulo L . Since D is a sparse matrix, it is conveniently implemented based on a hash. Thus, the time complexity of this algorithm is bounded by computing all matrix entries, i.e. nm entries of S^* , $\mathcal{O}(nmL^2)$ entries of S , and $\mathcal{O}(nm)$ entries of D . Note that each entry is computed in constant

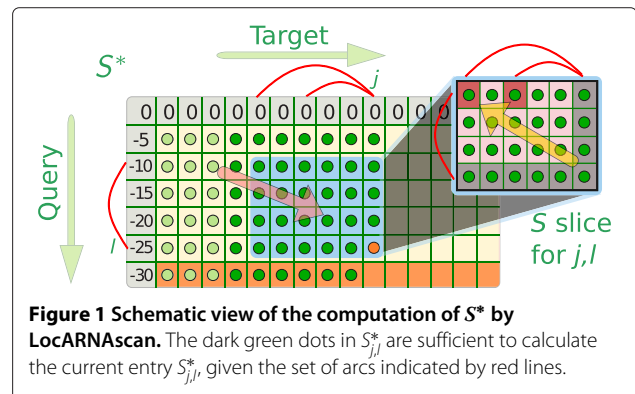


Figure 1 Schematic view of the computation of S^* by LocARNAscan. The dark green dots in $S_{j,l}^*$ are sufficient to calculate the current entry $S_{j,l}^*$, given the set of arcs indicated by red lines.

time; for S and S^* this holds due to considering only a sparse subset of base pairs. Thus, we derive the total time complexity $\mathcal{O}(nmL^2)$.

Sequence score contributions

The sequence contribution of our score is defined via the two similarity functions σ and τ . Note that LocARNAscan was designed to work even without sequence information. In this special case, we set both functions to constantly return 0.

In general, LocARNAscan accepts information about the query sequence in the form of a multiple alignment. Replacing the Ribosum-like [15] definition of σ and τ in LocARNA, we suggest to utilize log-odd scores based on nucleotide (and nucleotide pair) frequencies in this multiple alignment. Whereas the original LocARNA score is tailored for comparing single sequences or constructing multiple alignments (there used in a sum-of-pairs score), log-odd based scores, which are similarly utilized by Infernal [13], are more appropriate for scanning applications.

Given two column vectors q and q' of nucleotides and gaps from the query multiple alignment and given two nucleotides t and t' from the target, $f_q(t)$ denotes the frequency of t in q ; $f_{q,q'}(t, t')$, the frequency of pairs t and t' in the corresponding rows of q and q' . Let b_t be the background frequency of nucleotide t ; $b_{tt'}$, of the nucleotide pair t, t' in canonical base pairs. The similarity in the single stranded case is then defined by

$$\sigma(t, q) = \log \frac{f_q(t)}{b_t}; \quad (8)$$

the similarity in the base paired case, by

$$\tau(t, t', q, q') = \log \frac{f_{q,q'}(t, t')}{b_{tt'}}. \quad (9)$$

For simplicity, we assume uniform distribution of single nucleotides and nucleotides in base pairs. That is we use background frequencies $b_t = 1/4$ and $b_{tt'} = 1/6$; the latter reflects that we consider all six canonical base pairs.

In our implementation, for fast evaluation of the similarity functions, we compute the profiles, consisting of f_{Q_k} and f_{Q_k, Q_l} for all query position $k < l$, prior to the actual scanning. Furthermore, to handle small query alignments and smoothen the scoring functions, we add pseudocounts depending on the query's mononucleotide background frequencies.

Reporting optimal and suboptimal occurrences of the query

An occurrence of Q in T is a subsequence $T[i] \dots T[j]$ of T ; its score is the best subscore for $i \dots j$ and $1 \dots m$. The optimal occurrence of Q in T is determined during the run of Algorithm 1 by recording the j with the best score $S^*(j, m)$. By definition of S^* , j is the right end of the

target subsequence that optimally aligns with the query, in the case that an arbitrarily long left end of the target can be deleted for free. Albeit finding the left end requires extra work (see next subsection), the right end suffices to specify the occurrence.

For reporting suboptimal occurrences, one can record all scores $s_j := S^*(j, m)$ during the run of the algorithm. However, reporting all occurrences down to a certain score threshold in this vector is unsatisfactory, since good scores are usually flanked by only slightly worse scores that refer to the same occurrence with slightly altered alignment. For this reason, we report only local maxima of s_j , i.e. j , where $s_j \geq \max(s_{j-1}, s_{j+1})$. Furthermore, we do not report a local maximum j' if it is too close to a reported local maximum j with a better or equal score. Setting the distance threshold to one query length (m), we avoid reporting occurrences with substantial overlap. More formally, for j and j' where $|j - j'| \leq m$, we say that j *dominates* j' , iff $s_j > s_{j'}$ or ($s_j = s_{j'}$ and $j < j'$). We prune all dominated local maxima and report only the non-dominated ones.

Limiting the memory consumption independent of the target length is crucial for scanning. In addition, random access to data growing linearly with the target length has to be avoided. Consequently, we devise an online pruning algorithm with space requirements linearly bounded by the query length. While scanning the target by algorithm 1, we maintain a list of local maxima j' and remember their corresponding scores $s_{j'}$. We maintain three invariants: 1) list entries are increasing, 2) the distance of successive list entries is smaller than or equal to m , and 3) scores of list entries are strictly increasing.

When identifying a new local maximum j (after computing $s_{j+1} = S^*(j + 1, m)$), we try to resolve as many domination relations as possible. If $j - m$ is larger than the last list entry, entries of the list are independent of the new local maximum and all further local maxima. Thus, the list is resolved by iteratively reporting the last list entry and removing all dominated entries. All non-dominated entries are reported and the list is deleted. Finally, unless j is dominated by the last j' in the list, we push j to the list. Note that here we push to the empty list or the score s_j is higher than the score of the last list entry. After scanning the entire target, the list is resolved again.

The correctness follows from preserving the invariants in all cases. It remains to show that the list length is linearly bounded by the query length. This is due to the discretization of our score (we round all score contributions to limited precision); being an additive score, the score is linearly bounded by the query length; consequently, lists satisfying the third invariant have linearly bounded length. Note that in the average case, lists are even much shorter than this theoretic worst case bound suggests.

To handle large targets without excessive use of main memory, we implement two reporting strategies. Either we output all reported occurrences immediately or we limit the number of reported occurrences a priori to some K . To output only K best occurrences, we store them in a priority queue, sorted such that the occurrence with minimum score is on top and thus can be removed, whenever the limit is exceeded. Finally, the contents of the queue are output. While outputting the occurrences, we keep a histogram of the locally maximal scores for later use in determining the significance of an occurrence from the empirical score distribution. The entire strategy is generally similar to GotohScan [49] and extended here by the online pruning procedure.

To determine the significance of an occurrence, we compared several theoretical distributions to the empirical score distribution, but none approximated the alignment scores well (Figure 2). Hence, we followed the approach presented in [49]: for the identification of extreme alignment scores, it suffices to approximate the right tail of the distribution; the tail can be fitted well by an affine function. This strategy enables to return only *significant* occurrences with e-value up to a given threshold.

Traceback and Bound on Alignment Length

Obtaining the actual alignments corresponding to occurrences with right end j and recovering the left end of the occurrence requires a traceback procedure through the dynamic programming matrices. This poses the problem that space limitations forbid storing the entire matrices,

such that these matrices are not available after running Algorithm 1.

Therefore, given j , we determine and recompute the relevant part of matrix S^* and the corresponding D entries. Efficiency demands to keep the recomputed relevant part of S^* as small as possible. Given the score s_j , we derive a lower bound i for relevant left ends i' of the occurrence. Consequently, since the occurrence score cannot depend on entries with lower target left ends $i' < i$, one initializes all $S^*(i-1, k')$ with $-\infty$ ($1 \leq k' \leq m$), all $S^*(j', 0)$ with 0 ($i-1 \leq j' \leq j$), and recomputes all entries $S^*(j', k')$, where $i \leq j' \leq j$ and $1 \leq k' \leq m$, according to Eq. (6).

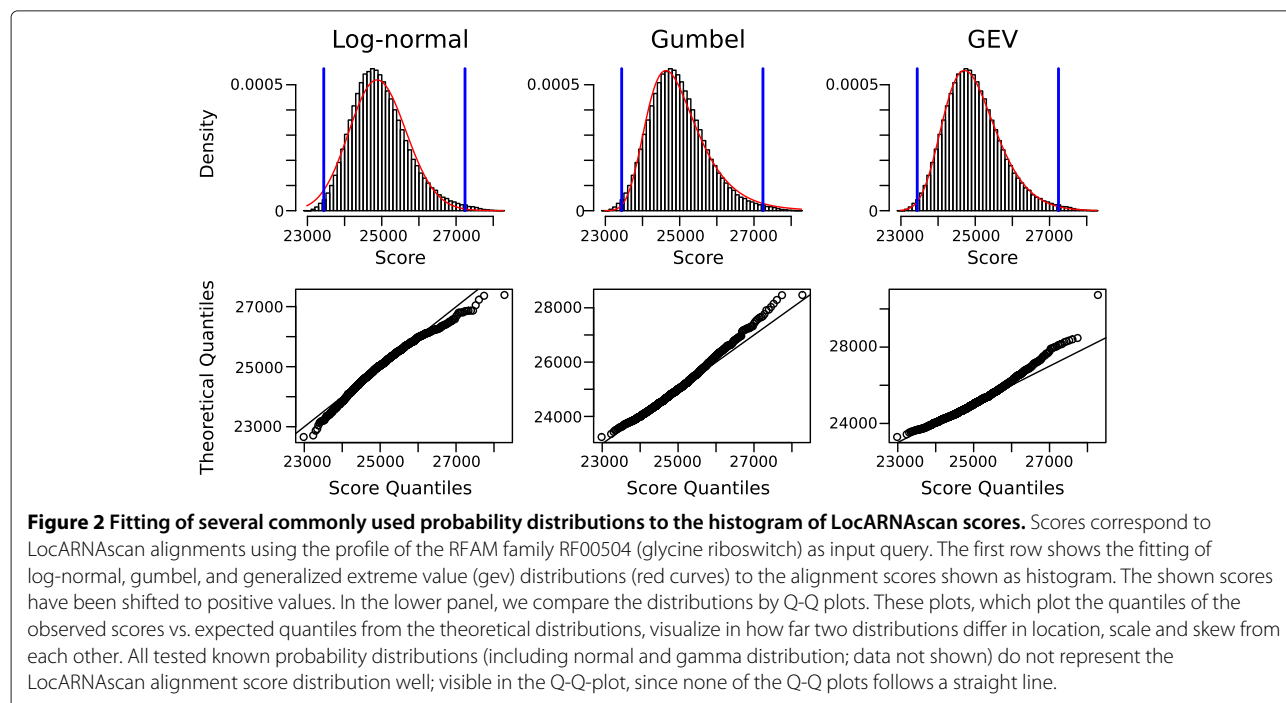
Finally, transferring a strategy known from LocARNA, we free S entries during this first recomputation, such that we recompute the S entries “on the trace” a second time, when performing the actual traceback. In this way, the traceback does not require more space than the score computation.

Analogously to [45], we bound the difference between the length of query and occurrence by Δ , where we choose $\Delta := (s_j - m \max(\sigma_{\max}, \tau_{\max}/2))/\gamma$ such that

$$s_j \leq m \max(\sigma_{\max}, \tau_{\max}/2) + \Delta\gamma,$$

and σ_{\max} and τ_{\max} denote the respective maximum single base match and maximum base pair match contributions between query and target; notably, the latter similarity includes sequence contributions due to τ and structure contributions due to Ψ^T and Ψ^Q .

Whereas [45] suggests to limit the “history” during the search phase by $m + \Delta$, we limit this more strongly to L at the price of recomputation. Making the common case



fast, this strategy is generally advantageous; it reduces load in the scanning phase, while only very few entries close to high scoring occurrences have to be recomputed, causing largely negligible extra cost. Similarly, one bounds the number of required entries of D , since the benefit of matching two base pairs has to justify a potential length difference of the enclosed subsequences.

LocARNAscan recognizes thermodynamic stability of occurrences

To study the specific behavior of LocARNAscan, we compared the performance of LocARNAscan and Infernal on a designed target containing a series of thermodynamically stable and (presumably) unstable occurrences. This allowed us to measure the difference in sensitivity to both classes given different training information.

First, we designed two sets of 1000 RNA decoys each. In the first set, which we call *stabilized class*, the RNA decoys were designed, applying inverse folding, to fold into tRNA structures with high thermodynamic stability. In the second set, called *non-stabilized class*, the decoys were generated with the potential to fold into tRNA structures forming canonical base pairs, but their stability is purely by chance. By design, the decoys are not related on the sequence level to each other or to known RNAs but share approximately the same mononucleotide frequencies. However, each decoy has the length and structure of a different randomly selected sequence from the Rfam tRNA family. From these decoys, we generated one *pseudogenome* consisting of an RNA decoy every 200 bases padded with random nucleotides from the same mononucleotide distribution.

For each class, we selected 100 samples at random to use them as training sets. Based on the Rfam seed alignment, we obtained a multiple alignment of the 100 samples annotated with a consensus structure derived from the Rfam consensus structure of the entire family. From both multiple alignments, we generated covariance models for Infernal. From the model for the non-stabilized class, we stripped off all sequence information, replacing it by a background model based on the mononucleotide frequencies of the pseudogenome. This procedure resulted in a *stripped model* and a *stabilized model*. For LocARNAscan, we generated two different queries. The first query consists of a tRNA of median length, where the sequence was replaced by a string of Ns and P^Q was generated from the Rfam structure of this tRNA; the second, of the "stabilized" multiple alignment and a matrix P^Q that was generated from its Rfam-derived consensus structure. By design, the first query contained at most the information of the stripped model, whereas the second query and the stabilized model contain exactly the same information.

We emphasize that both, the stripped CM and the all-N query sequence represent intentionally extreme cases.

Since stable RNAs favor CG base pairs over AU and GU base pairs and conversely, A and U in single stranded regions over C and G, one could usually utilize such knowledge, even in the absence of sequence information. Without doubts, Infernal would profit from this knowledge; building corresponding models is even supported via `cmbuild's` option `eset 0` in Infernal 1.1. As well, LocARNAscan could be similarly extended to mirror Infernal's behavior. Nevertheless, we intentionally study the extreme scenario to isolate the effect of incorporating thermodynamic stability.

Finally, we performed four scans of the pseudogenome. We run LocARNAscan with both designed queries and Infernal with the stripped model and the stabilized model. While the former model allowed Infernal to make full use of its training machinery, the latter tested Infernal's performance without sequence information. Per query, Infernal scanned the genome in 9 minutes (on Intel Q9400 2.66 Ghz), whereas LocARNAscan took only 3:10 minutes. Using its HMM filter, Infernal improves to roughly 6 minutes. LocARNAscan additionally requires to precompute base pair probabilities by folding the genome once (for potentially many queries); RNAplfold performed this in 5 minutes.

In Figure 3, we compare the separation of the two decoy classes by the different runs by their classification behavior. For each run, we plot the number of occurrences that coincide with stabilized decoys vs. the number with non-stabilized decoys at the same score threshold. The curves show that Infernal without sequence information completely fails to distinguish the two classes; in contrast, samples of stabilized decoys contain enough

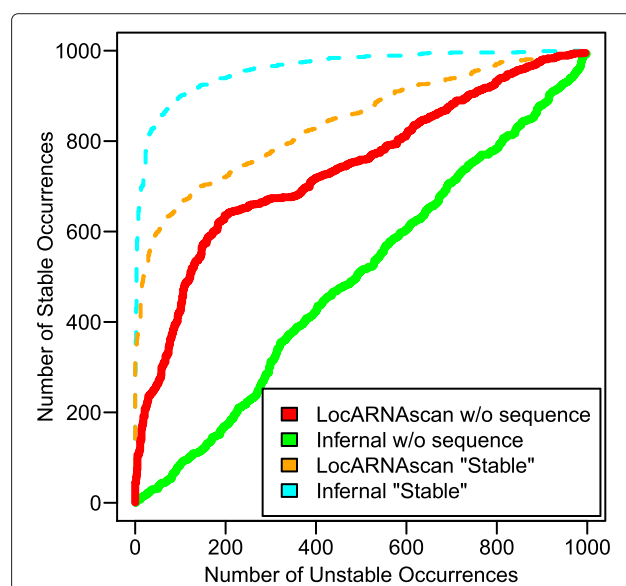


Figure 3 Classification of thermodynamically stabilized vs. non-stabilized occurrences.

sequence information to allow Infernal to classify almost perfectly. Trained without sequence information, LocARNAScan is superior to Infernal in sensing the stability of the decoys. Given the multiple alignment, LocARNAScan gains classification strength, but does not match the excellent performance of Infernal with sequence information.

Discussion and Conclusions

Genome-wide search for homologs of structured RNAs can benefit from the additional information encoded in their base pairing patterns. Nevertheless, currently available tools predominantly utilize the sequence information contained in query and target. The structural contributions are incorporated at the level of consistency between target sequence and query structure. Here we asked whether it can be worthwhile to include direct structural information, and thus implicitly evidence for the stability of secondary structure, also on the side of the target.

This is feasible in practice based on “scanning versions” of RNA secondary structure prediction tools that compute e.g. probabilities for local base pairs with a limited span in the target. Homology search on such a structure-annotated target is naturally performed as a local or semi-global sequence-structure alignment. With LocARNAScan we devised an efficient implementation of a semi-global variant of the Sankoff algorithm. For applications of genome-wide searches, we developed several algorithmic improvements relative to the global LocARNA approach. Rethinking the trade-offs between storing and recomputing intermediate data, LocARNAScan requires only $\mathcal{O}(Lm)$ memory, dependent only on the query size m and the span L of the precomputed base pairs in the target, but independent of the size n of the target itself, which is linearly read from disc and does not need to be stored in its entirety.

LocARNAScan’s CPU requirements of $\mathcal{O}(nmL^2)$ make genome-wide scanning feasible. In our experiment, LocARNAScan including the precomputation of base pair probabilities by RNAplfold, is about as fast as Infernal; this does not even change much when Infernal is allowed to use HMM filtering, since the small contribution of sequence information in our setting limits the effect of filtering. Calibration of the Infernal model, which is usually required, would have increased its total time requirements dramatically. The actual scan by LocARNAScan and the RNAplfold precomputation took almost the same time. We point out that the theoretical worst case complexities $\mathcal{O}(nmL^2)$ of LocARNAScan, $\mathcal{O}(nL^2)$ of RNAfold, and $\mathcal{O}(nm^2)$ of Infernal would suggest a different ordering of run times. The observed run times are a consequence of very different constant overheads. While LocARNAScan’s L^2 factor is strongly reduced by the filtering of base pairs by their probability, e.g. RNAplfold’s overhead is much larger due to the complex energy model and Infernal’s run

time actually depends on the number of CM states, which is a multiple of the query length.

Not surprisingly, little can be gained by the extra expense of the sequence-structure alignment as long as sufficient sequence information is contained in the query. We therefore concentrated here on the regime in which our current homology search tools are effectively blind, i.e., cases in which sequence conservation is completely absent. We find that in such an extreme setting LocARNAScan retains the ability to distinguish between thermodynamically stable RNA elements and decoys that admit the same base pairing patterns albeit far away from their groundstate. In contrast, Infernal is blind to this difference.

Of course, this is an extreme regime that is of interest in rare applications since most RNA families also exhibit an appreciable level of sequence conservation. A further practical limitation is the accuracy of the predicted base pairs of the target structure. Several approaches for the genome-wide measurements of secondary structure (see [50] and the references therein), however, promise to at least alleviate this issue in the near future.

Methods

Design of decoy RNAs

Each decoy (in both the stabilized and the non-stabilized class) is generated from a randomly selected entry in the Rfam 11.0 seed alignment of the tRNAs. First, we remove all gap columns and the corresponding symbols in the dot-bracket consensus structure string. If this deletes only one parenthesis of a parentheses pair, we replace the other one by a dot. In this way, we generate the ungapped tRNA sequence S and the corresponding projection of Rfam’s tRNA family consensus structure. Then, we fold the sequence constrained with the consensus structure projection (using RNAfold). This results in a specific structure R for the selected tRNA. For the stabilized class, we apply inverse folding into structure R optimizing the probability of R in the ensemble of the designed sequence S ; for the inverse folding we apply RNAinverse in partition function mode. In addition, we configure RNAinverse to stop at a cutoff corresponding to probability $p = 0.75$. This generally results in sequences, where R has only slightly higher probability.

The tool RNAinverse does not directly allow to stop the optimization at a probability p . However, RNAinverse accepts a minimal energy difference between the energy $E(R|S)$ of R for S and the ensemble free energy $F(S) = -\mathfrak{R}T / \ln(\sum_{\hat{R}} E(\hat{R}|S))$ of S . We utilize this feature after inferring the p -equivalent difference by

$$\Delta E = -\mathfrak{R}T / \ln(p), \quad (10)$$

where \mathfrak{R} denotes the gas constant and T the temperature.

We generate the non-stabilized decoys based on the mononucleotide frequencies of the stabilized ones. For generating a decoy in the non-stabilized class, given a structure R and mononucleotide frequencies $p_X (X \in \{A, C, G, U\})$, we draw the unpaired bases and bases at left base pair ends randomly from this distribution. The right ends of base pairs are set complementary to the left end bases (disallowing GU pairs).

Stripping off sequence information of a covariance Model

Similar to profile hidden Markov models (HMMs), Infernal's covariance models (CMs) are generative models that describe a probability distribution over sequences. Designed for RNAs, the CMs contain information about the RNA secondary structure, such that they can distinguish between unpaired bases and base pairs. Thus, like profile HMMs, the CMs contain nucleotide emission probabilities at different match states, but additionally contain nucleotide pair emission probabilities at special base pair match states. Technically, CMs contain log odd bit scores of the emissions calculated from the emission probabilities and, in uncalibrated CMs, a simple uniform null model. The key to entirely remove sequence information from a CM is thus to replace all emission scores of nucleotides and nucleotide pairs according to a background model.

Given mononucleotide frequencies $p_X (X \in \{A, C, G, U\})$, we replace all emission scores for nucleotide X by $\log_2(\frac{p_X}{1/4})$. For base pairs, we take into account a low probability p_{nc} of emitting a non-canonical nucleotide pair ($p_{nc} = 0.001$). The scores for nucleotide pairs $(X, Y) \in \{A, C, G, U\}^2$ are then replaced by $\log_2(\frac{(1-p_{nc})p_X p_Y / 6}{1/16})$ for canonical pairs $XY \in \{AU, UA, CG, GC, GU, UG\}$ or $\log_2(\frac{p_{nc} p_X p_Y / 6}{1/16})$ for non-canonical pairs.

Running LocARNAscan and Infernal

First, we performed two scans of the pseudogenome by LocARNAscan 1.0 ; the tool is available as free software at <http://www.bioinf.uni-leipzig.de/Software/LocARNAscan>. In extension of the presented score of Equation (1), LocARNAscan 1.0 supports affine gap cost and weighs the structure against sequence similarity. We set the weighting factor to 2.0 (option `struct-weight=200`) and the weighting of the sequence contribution $\tau(T_i, T_j; Q_k, Q_l)$ in the structure score component to 4.0 (option `tau=400`). Furthermore, we set the insertion/deletion score to -1.0 and gap opening cost to -5.0 (options `indel=-100` and `indel-opening=-500`). Finally, we activate the introduced log odd scoring of LocARNAscan (option `logoddscores` on). Base pairs in the pseudogenome are predicted with RNAplfold [47] forbidding lonely base pairs and setting a folding window size of 200nt and a maximal base pair span of 100nt. From the RNAplfold

prediction, we removed all base pairs with probability less than 0.5.

Second, we performed two corresponding scans by Infernal 1.0.2. We built Infernal models by `cmbuild` with default parameters. Then, we scan the pseudogenome with `cmsearch`. To avoid missing decoys we turned off HMM filtering (option `fil-no-hmm`). Furthermore, to produce comparable results, we scan only the forward strand (`cmsearch` option `top-only`) and run Infernal in *glocal* mode (option `-g`); the latter turns on LocARNAscan-like semi-global behavior.

For both, LocARNAscan and Infernal, we determined findings of stabilized and non-stabilized decoys and the corresponding occurrence score. There, we considered an occurrence, if it overlaps a decoy by at least 10% nucleotides. In the case of multiple occurrences overlapping the same decoy, we selected the best prediction as score for this decoy.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

RB, KR, PFS, and SW designed the algorithm and study; MS and SW implemented LocARNAscan; JE, SH, MS, and SW evaluated its performance; finally, all authors contributed to writing the manuscript. All authors read and approved the final manuscript.

Authors' information

Sebastian Will and Michael F Siebauer are joint first author.

Acknowledgements

We kindly thank Jana Hertel for discussions on occurrence pruning and e-value estimation similar to GotohScan.

Author details

¹Bioinformatics Group, Department of Computer Science, and Interdisciplinary Center for Bioinformatics, University of Leipzig, Härtelstraße 16-18, Leipzig D-04107, Germany. ²Bioinformatics Group, Department of Computer Science, Albert-Ludwigs-Universität Freiburg, Georges-Köhler-Allee 106, Freiburg D-79110, Germany. ³Genetics Group, Max Planck Institute for Evolutionary Anthropology, Deutscher Platz 6, Leipzig D-04104, Germany. ⁴Young Investigators Group Bioinformatics and Transcriptomics, Department Proteomics Helmholtz Centre for Environmental Research – UFZ, Permoserstraße 15, Leipzig D-04318, Germany. ⁵RNomics Group, Fraunhofer Institute for Cell Therapy and Immunology, Perlickstraße 1, Leipzig D-04103, Germany. ⁶Max Planck Institute for Mathematics in the Sciences, Inselstraße 22, Leipzig D-04103, Germany. ⁷Center for non-coding RNA in Technology and Health, University of Copenhagen Grønnegårdsvej 3, Frederiksberg C DK-1870, Denmark. ⁸Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, USA.

Received: 21 March 2013 Accepted: 28 March 2013

Published: 20 April 2013

References

1. Berretta J, Morillon A: **Pervasive transcription constitutes a new level of eukaryotic genome regulation.** *EMBO Rep* 2009, **10**:973–982.
2. Ponjavic J, Ponting CP, Lunter G: **Functionality or transcriptional noise? Evidence for selection within long noncoding RNAs.** *Genome Res* 2007, **17**:556–565.
3. Pheasant M, Mattick JS: **Raising the estimate of functional human sequences.** *Genome Res* 2007, **17**:1245–1253.

4. Ponting CP, Hardison RC: **What fraction of the human genome is functional?** *Genome Res* 2011, **21**:1769–1776.
5. Menzel P, Gorodkin J, Stadler PF: **The tedious task of finding homologous non-coding RNA genes.** *RNA* 2009, **15**:2075–2082.
6. Mosig A, Zhu L, Stadler PF: **Customized strategies for discovering distant ncRNA homologs.** *Brief Funct Genomic Proteomic* 2009, **8**:451–460.
7. Piskol R, Stephan W: **Selective constraints in conserved folded RNAs, of drosophilid and hominid genomes.** *Mol Biol Evol* 2011, **28**:1519–1529.
8. Washietl S, Hofacker IL, Stadler PF: **Fast and reliable prediction of noncoding RNAs.** *Proc Natl Acad Sci USA* 2005, **102**:2454–2459.
9. Pedersen JS, Meyer IM, Forsberg R, Simmonds P, Hein J: **A comparative method for finding and folding RNA secondary structures within protein-coding regions.** *Nucleic Acids Res* 2004, **32**:4925–4936.
10. Gesell T, Washietl S: **Dinucleotide controlled null models for comparative RNA gene prediction.** *BMC Bioinformatics* 2008, **9**:248.
11. Burge SW, Daub J, Eberhardt R, Tate J, Barquist L, Nawrocki EP, Eddy SR, Gardner PP, Bateman A: **Rfam 11.0: 10 years of RNA families.** *Nucleic Acids Res* 2013, **41**:D226–D232.
12. Freyhult EK, Bollback JP, Gardner PP: **Exploring genomic dark matter: a critical assessment of the performance of homology search methods on noncoding RNA.** *Genome Res* 2007, **17**:117–125.
13. Nawrocki EP, Kolbe DL, Eddy SR: **Infernal 1.0: inference of RNA alignments.** *Bioinformatics* 2009, **25**:1335–1337.
14. Gautheret D, Lambert A: **Direct RNA motif definition and identification from multiple sequence alignments using secondary structure profiles.** *J Mol Biol* 2001, **313**:1003–1011.
15. Klein RJ, Eddy SR: **RSEARCH: Finding homologs of single structured RNA sequences.** *BMC Bioinformatics* 2003, **4**(44):1471–2105.
16. Bussotti G, Raineri E, Erb I, Zytznicki M, Wilm A, Beaudoin E, Bucher P, Notredame C: **BlastR-fast and accurate database searches for non-coding RNAs.** *Nucleic Acids Res* 2011, **39**:6886–6895.
17. Gardner PP, Daub J, Tate J, Moore BL, Osuch IH, Griffiths-Jones S, Finn RD, Nawrocki EP, Kolbe DL, Eddy SR, Bateman A: **Rfam: Wikipedia, clans and the “decimal” release.** *Nucleic Acids Res* 2011, **39**:D141–D151.
18. Rivas E, Klein RJ, Jones TA, Eddy SR: **Computational identification of noncoding RNAs in *E. coli* by comparative genomics.** *Curr Biol* 2001, **11**:1369–1373.
19. Washietl S, Hofacker IL, Lukasser M, Huttenhofer A, Stadler PF: **Mapping of conserved RNA secondary structures predicts thousands of functional non-coding RNAs in the human genome.** *Nat Biotech* 2005, **23**:1383–1390.
20. Pedersen JS, Bejerano G, Siepel A, Rosenbloom K, Lindblad-Toh K, Lander ES, Kent J, Miller W, Haussler D: **Classification of conserved RNA secondary structures in the human genome.** *PLoS Comput Biol* 2006, **2**:e33.
21. Torarinsson E, Sawera M, Havgaard J, Fredholm M, Gorodkin J: **Thousands of corresponding human and mouse genomic regions unalignable in primary sequence contain common RNA structure.** *Genome Res* 2006, **16**:885–889.
22. Will S, Missal K, Hofacker IL, Stadler PF, Backofen R: **Inferring non-coding RNA families and classes by means of genome-scale structure-based clustering.** *PLoS Comp Biol* 2007, **3**:e65.
23. Gräf S, Strothmann S, Kurtz S, Steger G: **HyPaLib: a database of RNAs and RNA structural elements defined by hybrid patterns.** *Nucleic Acids Res* 2001, **29**:196–198.
24. Macke TJ, Ecker DJ, Gutell RR, Gautheret D, Case DA, Sampath R: **RNA motif, an RNA secondary structure definition and search algorithm.** *Nucleic Acids Res* 2001, **29**(22):4724–4735.
25. Höchsmann T, Höchsmann M, Giegerich R: **Thermodynamic Matchers: strengthening the significance of RNA folding energies.** In *Computational Systems Bioinformatics, CSB 2006*. Edited by Markstein P, Xu Y. Singapore: World Scientific; 2006:111–121.
26. The Athanasius FbompfünereRNA Consortium; Backofen R, Flamm C, Fried C, Fritsch G, Hackermüller J, Hertel J, Hofacker IL, Missal K, Mosig SJ, Axel Prohaska, Rose D, Stadler PF, Tanzer A, Washietl S, Sebastian W: **RNAs everywhere: Genome-wide annotation of structured RNAs.** *J Exp Zool B: Mol Dev Evol* 2007, **308B**:1–25.
27. Höchsmann M, Töller T, Giegerich R, Kurtz S: **Local similarity in RNA secondary structures.** In *Proc of the Computational Systems Bioinformatics Conference, Stanford, CA, August 2003 (CSB 2003)*; 2003:159–168.
28. Siebert S, Backofen R: **MARNA: multiple alignment and consensus structure prediction of RNAs based on sequence structure comparisons.** *Bioinformatics* 2005, **21**:3352–3359.
29. Sankoff D: **Simultaneous solution of the RNA folding, alignment, and proto-sequence problems.** *SIAM J Appl Math* 1985, **45**:810–825.
30. Gorodkin J, Heyer LJ, Stormo GD: **Finding the most significant common sequence and structure motifs in a set of RNA sequences.** *Nucleic Acids Res* 1997, **25**:3724–3732.
31. Hull Havgaard JH, Lyngsø R, Stormo GD, Gorodkin J: **Pairwise local structural alignment of RNA sequences with sequence similarity less than 40%.** *Bioinformatics* 2005, **21**:1815–1824.
32. Mathews DH, Turner DH: **Dynalign: an algorithm for finding the secondary structure common to two RNA sequences.** *J Mol Biol* 2002, **317**:191–203.
33. Bradley RK, Pachter L, Holmes I: **Specific alignment of structured RNA: stochastic grammars and sequence annealing.** *Bioinformatics* 2008, **24**:2677–2683.
34. Hofacker IL, Bernhart SH, Stadler PF: **Alignment of RNA base pairing probability matrices.** *Bioinformatics* 2004, **20**:2222–2227.
35. McCaskill JS: **The equilibrium partition function and base pair binding probabilities for RNA secondary structure.** *Biopolymers* 1990, **29**:1105–1119.
36. Heyne S, Will S, Beckstette M, Backofen R: **Lightweight comparison of RNAs based on exact sequence-structure matches.** *Bioinformatics* 2009, **25**:2095–2102.
37. Schmiel C, Möhl M, Heyne S, Amit M, Landau GM, Will S, Backofen R: **Exact pattern matching for RNA structure ensembles.** In *Proceedings of the 16th International Conference on Research in Computational Molecular Biology (RECOMB 2012), Volume 7262 of LNCS*. Edited by Chor. B. Heidelberg: Springer-Verlag; 2012:245–260.
38. Backofen R, Will S: **Local sequence-structure motifs in RNA.** *J Bioinf Comput Biol* 2004, **2**:681–698.
39. Backofen R, Siebert S: **Fast detection of common sequence structure patterns in RNAs.** *J Discr Alg* 2007, **5**:212–228.
40. Will S, Yu M, Berger B: **Structure-based whole-genome realignment reveals many novel noncoding RNAs.** *Genome Res*. 2013, **Jun**;23(6):1018–27. doi:10.1101/gr.137091.111. Epub 2013 Jan 7.
41. Will S, Miladi CSM, Möhl M, Backofen R: **SPARSE: Quadratic time simultaneous alignment and folding of RNAs without sequence-based heuristics.** In *Proceedings of the 17th International Conference on Research in Computational Molecular Biology (RECOMB 2013), Volume 7821 of LNCS*. Edited by Deng M, Jiang R, Sun F, Zhang X. Heidelberg: Springer-Verlag; 2013:289–290.
42. Will S, Joshi T, Hofacker IL, Stadler PF, Backofen R: **LocARNA-P: Accurate boundary prediction and improved detection of structural RNAs.** *RNA* 2012, **18**:900–914.
43. Nawrocki EP: **Structural RNA homology search and alignment using covariance models.** *PhD thesis*. Washington University, Saint Louis 2009.
44. Gruber AR, Bernhart SH, Hofacker IL, Washietl S: **Strategies for measuring evolutionary conservation of RNA secondary structures.** *BMC Bioinformatics* 2008, **9**:122.
45. Bompfünere AF, Backofen R, Bernhart SH, Hertel J, Hofacker IL, Stadler PF, Will S: **Variations on RNA folding and alignment: Lessons from Benasque.** *J Math Biol* 2008, **56**:129–144.
46. Kolbe DL, Eddy SR: **Local RNA structure alignment with incomplete sequence.** *Bioinformatics* 2009, **25**:1236–1243.
47. Bernhart S, Hofacker IL, Stadler PF: **Local RNA base pairing probabilities in large sequences.** *Bioinformatics* 2006, **22**:614–615.
48. Lange SJ, Maticzka D, Mohl M, Gagnon JN, Brown CM, Backofen R: **Global or local? Predicting secondary structure and accessibility in mRNAs.** *Nucleic Acids Res* 2012, **40**(12):5215–5226.
49. Hertel J, de Jong D, Marz M, Rose D, Tafer H, Tanzer A, Schierwater B, Stadler PF: **Non-coding RNA annotation of the genome of *Trichoplax adhaerens*.** *Nucleic Acids Res* 2009, **37**:1602–1615.
50. Wan Y, Kertesz M, Spitale RC, Segal E, Chang HY: **Understanding the transcriptome through RNA structure.** *Nat Rev Genet* 2011, **12**:641–655.

doi:10.1186/1748-7188-8-14

Cite this article as: Will et al.: LocARNAscan: Incorporating thermodynamic stability in sequence and structure-based RNA homology search. *Algorithms for Molecular Biology* 2013 **8**:14.