BIOINFORMATICS

Vol. 19 no. 1 2003
Pages 108–116

Finding the common structure shared by two homologous RNAs

O. Perriquet, H. Touzet* and M. Dauchet*

LIFL, UPRESA CNRS 8022, Bâtiment M3, Université des Sciences et Technologies de Lille, 59655 Villeneuve d'Ascq Cedex, France

Received on February 14, 2002; revised on May 7, 2002; June 27, 2002; accepted on July 23, 2002

ABSTRACT

Motivation: CARNAC is a new method for pairwise folding of RNA sequences. The program takes into account local similarity, stem energy, and covariations to produce the common folding. It can handle all RNA types, and has also been adapted to align a new homologous sequence along a reference structured sequence.

Results: Using different data sets, we show that CARNAC provides a good partial prediction for a wide range of sequences (16S ssu rRNA, RNase P RNA, viruses) with only two sequences. In presence of a whole family of sequences, we also show that CARNAC can be used to detect whether the sequences actually share the same structure.

Availability: CARNAC is available at the URL http://www.lifl.fr/∼perrique/rna/

contact: perrique@lifl.fr; touzet@lifl.fr

INTRODUCTION

Analysis of RNA sequences can be considered at different levels: the *primary structure* (ground level) is the sequence of bases A, U, C, G, the *secondary structure* is the graph of weak interactions between bases that can be drawn in a half plane with no crossing edge, the *tertiary structure* is the complete graph of interactions, and the *steric structure* is the gift of exact atomic 3D positions. Intermediate stages can be defined between secondary and tertiary structure, in particular *secondary structure with pseudoknots*, where intertwined pairings are allowed.

At a first glance, secondary structure prediction methods can be divided in two great families: *Minimization of Free Energy* and *Comparative Sequence Analysis*. Energy minimization is the favourite method to fold a single RNA sequence: the famous program MFOLD, conceived by Mathews and colleagues (Mathews *et al.*, 2000), uses a set of experimentally determined energy parameters (Mathews *et al.*, 1999) to compute the structure of a given RNA. The underlying algorithm is an improvement of the Nussinov algorithm (Nussinov and Jacobson, 1980) adapted for the nearest neighbour energy model. As the free energy of the best structure is supposed to be *close* to its minimum, MFOLD predicts suboptimal structures too (Zuker, 1989). In the large set of suboptimal structures, many correct stems are often found, but there is no 'level of confidence' information about those pieces of structure. Comparative sequence analysis applies when we have a large family of homologous RNA sequences: one aligns them and looks for covariations (compensated mutations that conserve structure). The method has been implemented using chi-square (Chiu and Kolodziejczak, 1991) or Stochastic Context Free Grammars (Eddy and Durbin, 1994). The alignment must be refined by successive iterations, and the results strongly depend on the quality of the initial alignment. It is worth noting that alignment requires enough sequence homology to be reliable, whereas covariation hunting asks for sequences to be different enough to covary. So if the sequences are too close, a great number of homologous RNAs will be needed, and if they are too different, they will be unalignable.

A third family of prediction methods is now appearing, trying to use both the intrinsic information of a single sequence, and the mutual information of several homologs. The idea is that function is strongly related to structure, and sequences sharing the same function should share the same structure. The recent programs use a small set of sequences to find the common folding: Juan and Wilson (1999) use five homologs Knudsen and Hein (1999) four homologs, and Wang and Zhang (1999) only three homologs. Sankoff (1985) proposed an algorithm to align and fold two sequences at the same time, but the average complexity $O(n^6)$ in time, and $O(n^4)$ in space, made it impracticable. Some adaptations, like DYNALIGN (Mathews and Turner, 2002) or FOLDALIGN (Gorodkin *et al.*, 1997), apply successfully to small sequences.

In this paper, we present a new method, called CARNAC (*Computer Alignment of RNA by Cofolding*), that belongs to that last family: it has been devised to find the common structure shared by two homologous sequences, which may be viewed as the *intersection* of their sets of potential low energy structures. The program builds

*To whom correspondence should be addressed.

a common folding, taking into account sequence local similarity, energy and covariations. CARNAC does not always recover the whole structure but provides a good partial prediction for a wide range of sequences.

The paper is organized as follows. We first describe the algorithm implemented in CARNAC, then we present results on two families of structured RNAs (RNase P RNA, ssu rRNA). We show that two RNAs are enough to predict structure. We introduce a variation of CARNAC adapted to inverse folding. In the last section, we apply CARNAC to non-structured or partially structured RNAs, such as viruses: if we have a family of homologs, we can compute an index of confidence on the structures found. It is then possible to discriminate between coding and non-coding RNAs.

THE ALGORITHM

CARNAC takes as input two unaligned sequences in FASTA format, and produces a secondary structure in CONNECT format. It proceeds in four steps:

- first, it scans each sequence for the best candidate stems;
- second, it searches for the regions of high similarity between the two sequences, called *anchor points*;
- then, it performs a pairwise selection of stems, using information furnished by anchor points and covariations;
- finally, it constructs the common folding by energy minimization from the set of pre-selected stems.

The method has been implemented in C.

Finding the best stems

A stem is any set of stacked pairs of bases. For each sequence, CARNAC searches for the best stems, using stacking energies and tetraloop bonuses found by Mathews and colleagues (Mathews *et al.*, 1999), allowing some (penalized) mismatches. The energy matrix is computed and every stem whose energy is greater than a given threshold is selected.

As we want to find most of the *good* stems, without being drowned by a flood of noisy stems, the threshold tries to take into account the fact that a stem can appear *by chance*. It is not straightforward, in the stacking energy model, to evaluate the probability to find by chance a complementary sequence to close the stem. However it is intuitively clear that, given a small piece of sequence, if we find complementary bases rather far away in the sequence, it may be by chance, whereas if we find it closer it may not. For this reason, we consider a piecewise affine threshold that depends on the distance between the opening and the closure of a stem. This idea is also compatible with the principle of hierarchical folding (Tinoco and Bustamante, 1999). Threshold parameters have been empirically optimized with in sight the idea that they should not depend on the sequence type (see Figure 1).

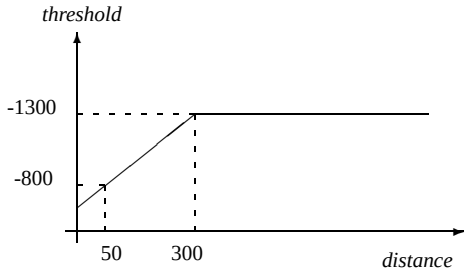


Fig. 1. Piecewise affine threshold depending on the distance between the opening and the closure of a stem. The value -800 is usually the energy of a 3 or 4 bases long stem, whereas -1300 is the energy of a 6 or 7 bases long stem. The threshold can be slightly modified by the user to take into account bias in GC content ($threshold = threshold \times (1 + (50 - GC\%)/50)$).

Finding anchor points

CARNAC searches for highly conserved regions between the two sequences, in order to use them as *anchor points*. These regions are detected using classical recursions for sequence alignment, except that we forbid indels: we credit 1 for a match and -2 for a mismatch. The selection of regions among the best diagonals of the matrix is made greedily, using a probability based measure selection.

If sequences were generated at random, following a Bernoulli model where $p_A = p_U = p_C = p_G$, the probability that two sequences of the same length l share at least $k \leqslant l$ common bases would follow a Binomial law

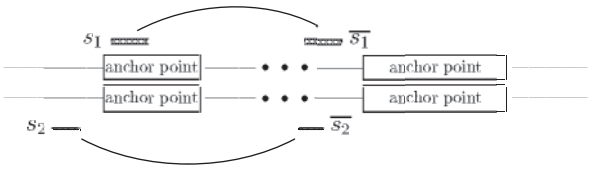
$$B(k,\ l) = \sum_{i \leqslant k} \binom{l}{i} p^i (1-p)^{l-i},$$

where $p = \frac{1}{4}$ according to the assumption of equirepartition of bases. Thus if we are given a subsequence of length l, the expectation of finding the same subsequence within a sequence of length $l' \geqslant l$ with at most m mismatches may be approximated by:

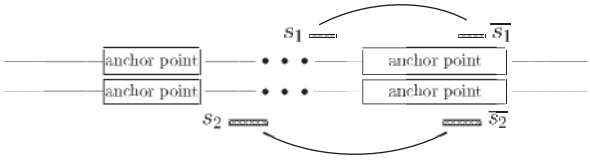
$$E(l,\ l', m) = (l' - l) \times B(l - m,\ l).$$

We consider that a region is conserved if $E(l, l + rel\_dist, m) \leqslant 10^{-8}$, where rel_dist is the relative shift distance between the two pieces of sequence (the reference origin being the end of the last conserved region). In practice, the threshold is high enough to neglect bias in nucleotide repartition and prevent conflicts between anchor points.

case 1 – anchor point violation:
If $(s_1, \overline{s_1})$ and $(s_2, \overline{s_2})$ were folded together, the alignment of s_1 and s_2 would contradict the anchor point alignment.



case 2 – shift too large outside an anchor point:
When the opening or the closure of a stem falls between two anchor points, a variable shift is allowed, which depends on the difference of distance between the anchor points. Here the shift between s_1 and s_2 is too large.



case 3 – shift too large within an anchor point:
When the opening or the closure of a stem falls within an anchor point, no shift is allowed. Here there is a slight shift for $\overline{s_1}$ and $\overline{s_2}$.

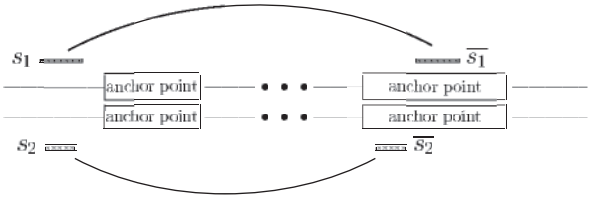


Fig. 2. Unmatchable stems.

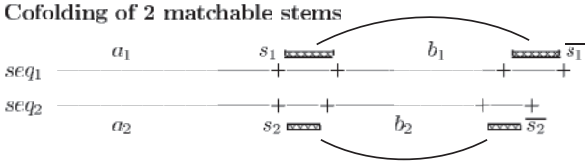
Pairwise selection of matchable stems

Pairs of matchable stems are created, checking for consistency with anchor points and for covariations. A stem of the first sequence is said to be *matchable* with a stem of the second sequence if these two conditions are fulfilled:

- the alignment of opening and closure induced when we close the two stems at the same time does not contradict the alignment induced by anchor points (unmatchable stems cases are pictured in Figure 2);
- there is at least one covariation. In our context, two positions are said to *covary* if both columns in the pairwise alignment of the stems are substitutions. This requirement may lead to underpredict the structure, but most of the time it discards false positive stems.

If a stem has no matchable partner, then it is rejected. It is possible that a stem has several matchable partners.

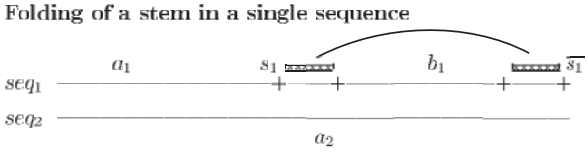
Cofolding of 2 matchable stems



$$\alpha(seq_1, seq_2) = min \begin{cases} \alpha(a_1, a_2) + \alpha(b_1, b_2) + \\ \quad 10 * (energy(s_1, \overline{s_1}) + energy(s_2, \overline{s_2})) \\ \alpha(a_1 s_1 b_1 t_1,\ a_2 s_2 b_2 t_2) \end{cases}$$

α is the energy of the common folding,
$(s_1, \overline{s_1})$ and $(s_2, \overline{s_2})$ are two matchable stems,
t_1 and t_2 are the largest proper prefixes of $\overline{s_1}$ and $\overline{s_2}$.

Folding of a stem in a single sequence



$$\alpha(seq_1, seq_2) = min \begin{cases} \alpha(a_1, a_2) + energy(s_1, \overline{s_1}) \\ \alpha(a_1 s_1 b_1 t_1, a_2) \end{cases}$$

$(s_1, \overline{s_1})$ is a potential hairpin (distance between s_1 and $\overline{s_1}$ should be less than 8 bases), and t_1 is the largest proper prefix of $\overline{s_1}$.

Fig. 3. Recursions used for the construction of the common folding.

Common folding

The last step of the algorithm consists in selecting *coherent* pairs of matchable stems to constitute the final structure.

According to the usual definition of secondary structure (Wang and Zhang, 1999), two stems *of the same sequence* are said to be coherent if they are nested or juxtaposed. As a natural generalization, two pairs of matchable stems are said to be coherent if the members are *both* nested or *both* juxtaposed. This definition explicitly excludes pseudoknots. The main reason is that allowing pseudoknots is too expensive from a computational point of view, even for a single sequence: the Zuker algorithm is $O(n^3)$ in time, while the algorithm of Rivas and Eddy (1999), where pseudoknots are allowed, is $O(n^6)$. Nevertheless, as secondary structure seems to be quite robust, and well conserved through evolution, it is possible to treat tertiary interactions as a *complement* of secondary structure, bringing eventually some perturbations in yet established secondary interactions (Wu and Tinoco, 1998).

The common folding of two sequences is thus defined as the subset of coherent pairs of matchable stems with lowest energy. CARNAC builds this subset from the whole set of pairs of matchable stems using dynamic programming (see Figure 3). The recursion formula is reminiscent of the formula introduced by Sankoff (1985). The first difference is that we consider an RNA sequence as a sequence of potential stems, not a sequence of nucleotides. This introduces some adaptations for the

recursion formula. We also allow the creation of a stem on a single sequence, with no counterpart in the other sequence, provided that this potential stem is a hairpin.

The brute recursion formula gives $O(n^6)$ for the time and $O(n^4)$ for the space complexity, where n is the length of the sequences to be folded. We optimize the algorithm by taking advantage of the preprocessing of the sequences: we only consider pairs of matchable stems with respect to anchor points and covariations. As we work with stems, instead of nucleotides, for each stem closure, there is only *one* opening, so we managed to adapt the data structure: for the implementation, we use a dynamic graph for storing the recursive calls, instead of a usual dynamic programming table. This last point is crucial for the efficiency of the program, and it is not applicable in Sankoff's initial approach. This permits us to significantly decrease the search space and, in practice, we empirically observe that we manage to gain about $O(n^2)$ in time and in space. CARNAC is thus quite efficient for all real examples we tried. We give more details in the next sections.

EXPERIMENTAL RESULTS

In this section we display and analyze the results of CARNAC on different data sets.

We show that CARNAC can handle poorly conserved RNAs (e.g. RNase P RNA) as well as large well conserved sequences (e.g. 16S ssu rRNA). For those sequences, we evaluate the quality of our results by comparing the predictions with a selection of other methods. We ran CARNAC on a Pentium III PC, 1 GHz, with 256 Mb RAM. Finally, we present a slight adaptation of CARNAC to perform a quick inverse folding along a known structure. All the sequences used are referenced in the Appendix.

RNase P RNA

Our first example is concerned with Delta/Epsilon Purple Bacteria RNase P RNA sequences that are available in the database developed by Brown (1999). The sequences are about 350 bases long and they share a common structure, in spite of their weak sequence conservation (60% of identity in average). The reference structure in the database (*D.desulfuricans*) has around 15 stems, plus 2 pseudoknots (4 and 8 bases long), but some stems are not always present in the structure of the others.

We constituted our test set by rejecting partial or redundant sequences: we finally obtained 5 sequences. We folded each sequence against each other. The results are diplayed in Table 1. Despite the poor sequence similarity, the pseudoknots and the variations in structure, more than half of the structure is predicted, with 85% correct in average. The computation time is less than a second for each folding.

We now try to evaluate how CARNAC performs in comparison with other existing similar programs: MFOLD (Mathews *et al.*, 2000), RNAGA (Chen *et al.*, 2000), DYNALIGN (Mathews and Turner, 2002) and FOLDALIGN (Gorodkin *et al.*, 1997). We do not compare with pure phylogenetic approaches, that use covariations only, since these methods are not applicable to only two sequences, or even to a few number of sequences.

We chose *D.desulfuricans* to be the reference organism. This corresponds to the first column of Table 1. We used DYNALIGN and FOLDALIGN exactly in the same way as CARNAC: we computed all pairwise foldings. For MFOLD, we had to fold each sequence separately, and for RNAGA we folded all sequences simultaneously. The results are summarized in Table 2.

MFOLD: We tried the standard thermodynamic method on each sequence separately, and we display the results for the best of the different suboptimal structures proposed by MFOLD. Note that even if the set of suboptimal structures contains a structure close to the true one, there is no straightforward way to know which one or which piece of structure is correct.

RNAGA: RNAGA is a genetic algorithm that builds the common structure from a set of unaligned sequences. We tried it with the whole set of RNase P RNA. By default, RNAGA makes ten proposals for each folding. For each sequence, we display the results for the best prediction amongst the ten proposed by RNAGA. We observe that all the predicted stems are hairpins. They are sometimes stacked, but there is no branching structure.

FOLDALIGN: The program FOLDALIGN searches for local conserved motifs in a set of RNA sequences. The core algorithm is an adaptation of Sankoff recursions: the authors managed to gain complexity by forbidding branching structures, and by considering a bounded shift between the two sequences. This method implies that only hairpins can be found. We ran FOLDALIGN for pairs of sequences. In this context, the output of FOLDALIGN is constituted of a single hairpin. As we did not manage to carry out the folding of full sequences, we prepared a sample set by cutting correctly (according to the structure) the left hand part and the right hand part of the RNase P RNA in order to have around 250 bases long sequences. In our examples, all hairpins predicted by FOLDALIGN are globally correct: at least one pairing in the hairpin is correct.

DYNALIGN: DYNALIGN is also based on Sankoff recursions, restricted by a tunable bounded shift between the sequences, so that it becomes tractable if the shift is small. We ran the program with different values for the shift parameter *max separation* (max sep). Of course the quality of the prediction increases as *max sep.* becomes larger, but, as expected, the running time too. With *max sep* $= 3$, the folding lasts about 15 min and with

Table 1. CARNAC output for Delta/Epsilon Purple Bacteria RNase P RNA

	D.desulfuricans	*D.vulgaris*	*G.sulfurreducens*	*C.jejuni*	*H.pylori*
D.desulfuricans	—	65 93% 44%	88 81% 34%	93 80% 31%	77 94% 33%
D.vulgaris	59 93% 49%	—	81 86% 35%	65 89% 46%	65 73% 56%
G.sulfurreducens	87 85% 33%	93 81% 31%	—	84 80% 39%	69 85% 47%
C.jejuni	61 88% 41%	52 78% 55%	63 84% 42%	—	35 94% 64%
H.pylori	51 92% 45%	54 70% 55%	48 89% 49%	42 88% 56%	—

The table should be read like this: the organism of the row is the current sequence, being folded with the sequence of the organism of the column. For each folding, the first number indicates the total number of predicted base pairings, then we give the percentage of correct base pairs in the prediction, and the percentage of false negatives.

Table 2. Comparison of CARNAC with other prediction methods

Name	Max sep	Suboptimal	Rank	bp	True positive	False negative
			CARNAC			
D. vulgaris	–	–	–	59	55 (93%)	49%
G. sulfurreducens	–	–	–	87	74 (85%)	33%
C. jejuni	–	–	–	61	54 (88%)	41%
H. pylori	–	–	–	51	47 (92%)	45%
			MFOLD			
D. desulfuricans	–	18	4	113	75 (66%)	31%
D. vulgaris	–	22	2	112	80 (71%)	26%
G. sulfurreducens	–	15	2	112	90 (80%)	19%
C. jejuni	–	6	1	86	59 (68%)	36%
H. pylori	–	9	4	95	56 (58%)	34%
			RNAGA			
D. desulfuricans	–	–	1	90	64 (71%)	41%
D. vulgaris	–	–	2	91	53 (58%)	51%
G. sulfurreducens	–	–	4	102	67 (66%)	39%
C. jejuni	–	–	10	79	43 (55%)	52%
H. pylori	–	–	1	84	40 (48%)	52%
			FOLDALIGN			
D. vulgaris	–	–	–	12	5 (41%)	95%
G. sulfurreducens	–	–	–	12	5 (41%)	95%
C. jejuni	–	–	–	13	5 (38%)	94%
H. pylori	–	–	–	10	5 (50%)	94%
			DYNALIGN			
			[max sep = 3, 5 and 10]			
	3	–	–	101	39 (38%)	64%
D. vulgaris	5	–	–	101	53 (52%)	51%
	10	–	–	108	85 (78%)	21%
	3	–	–	96	81 (84%)	27%
G. sulfurreducens	5	–	–	105	91 (86%)	18%
	10	–	–	105	94 (89%)	15%
	3	–	–	47	0 (0%)	100%
C. jejuni	5	–	–	67	0 (0%)	100%
	10	–	–	64	37 (57%)	60%
	3	–	–	53	3 (5%)	97%
H. pylori	5	–	–	41	14 (34%)	84%
	10	–	–	58	31 (53%)	64%

We study the folding of each sequence with the sequence of the reference organism *D. desulfuricans*, except for MFOLD (each sequence was folded separately) and RNAGA (the five sequences were folded simultaneously). For each folding, we mention the number of predicted base pairings *(bp)*, the number and the percentage of true positives, the purcentage of false negatives. For MFOLD, we display the number of suboptimal structures, and we selected the best structure. For RNAGA, we selected the best structure amongst the ten proposed by RNAGA. The column *rank* is the rank of this structure.

Table 3. Results of CARNAC for 16S ssu rRNA The table should be read like in Table 1

	Archaea		*Bacteria*	
	M.jannaschii	*S.solfataricus*	*E.coli*	*B.subtilis*
M. jannaschii	—	288 81% 49%	323 71% 50%	312 69% 53%
S. solfataricus	278 84% 50%	—	336 75% 45%	306 74% 51%
E. coli	308 76% 50%	299 74% 53%	—	237 87% 57%
B. subtilis	307 75% 51%	284 78% 52%	237 87% 56%	—

Table 4. Results of MFOLD (with default parameters) on the same set of 16S ssu rRNA

Name	Suboptimal	Rank	Predicted bp	True positive	False negative
M. jannaschii	14	9	494	294 (59%)	35%
S. solfataricus	16	7	495	287 (57%)	38%
E. coli	27	21	476	340 (71%)	28%
B. subtilis	17	5	489	271 (55%)	42%

We display the number of suboptimal structures found, and, *for the best one*, its rank within the suboptimal set, the number of predicted base pairs, and the percentages of true positives and false negatives

max sep = 10, it runs more than 8 h, while CARNAC gives its results in less than a second. The bad results of DYNALIGN concerning *C. jejuni* and *H. pylori* may be explained by the variations between the compared structures: one hairpin of *D. desulfuricans* is deleted in *C. jejuni* and in *H. pylori*, inducing a large shift in the alignment. CARNAC is more robust in this case.

As a conclusion, we see that all methods usually predict more correct pairings than CARNAC, but also more wrong pairings: CARNAC is more *specific*, while other programs tend to be more *sensitive*. As for FOLDALIGN, its purpose is clearly different from CARNAC, since it does not try to recover the whole structure, but it searches for local conserved hairpins.

16S ssu rRNA

All 16S ssu rRNA share a common structure, but the consensus structure is stronger when sequences belong to the same domain. We chose on purpose well known species in two different domains in order to test the robustness of CARNAC against variations in the overall common structure. The four rRNAs were taken from the databank of Woese and colleagues (Woese *et al.*, 1980) and pairwise folded. The sequences are about 1500 bases long and the consensus structure has around 80 stems.

Table 3 summarizes the results of CARNAC. For the computation time, 2/3 of the foldings took less than 20 s, and 1/3 less than a minute. We observe that the running time is also strongly amplified by the bias in GC content, since sequences that are rich in GC produce more potential stems. The last factor is the sequence similarity: the number of pairs of matchable stems is larger when the sequences are divergent, due to the weaker quality of anchor points.

As expected, the quality of the prediction seems to be related to sequence similarity. The closer the sequences are in the phylogenetic tree, the more the structure is shared, and thus the better the prediction is: *E. coli* versus *B. subtilis* (77% of identity) gives 87% of correctness while *M. jannaschii* versus *B. subtilis* (67% of identity) gives only 69% for *M. jannaschii*. For this last folding, most of the mispredicted stems appear close to regions where the structure is less conserved: stems may be different in branching, length, or contain many mismatches. A part of those wrong stems are only shifts, and do not contradict the overall branching structure of the molecule.

We compare the results of CARNAC with the results of MFOLD (Table 4). Other methods do not apply on that set, because the sequences are too long.

CARNAC and inverse folding

The *Inverse Folding Problem* consists in finding the structure of the second sequence when the structure of the first one is known (we assume they share the same structure). This known structure may be the consensus structure of an alignment. In that case, inverse folding is simply adding a sequence to a structural alignment (Corpet and Michot, 1994). *ReverseCARNAC* is an adaptation of CARNAC in order to achieve inverse folding: we give as input the two sequences in FASTA format and a list of

Table 5. Results of ReverseCARNAC on 16S ssu rRNA: number of predicted base pairs, percentage of true positives, and percentage of false negatives

reference structure ⟶	*Archaea* *M.jannaschii*	*S.solfataricus*	*Bacteria* *E.coli*	*B.subtilis*
M.jannaschii	—	371 86% 29%	343 80% 39%	347 75% 42%
S.solfataricus	348 87% 35%	—	393 82% 30%	337 80% 41%
E.coli	272 90% 48%	308 78% 49%	—	319 90% 39%
B.subtilis	297 87% 44%	297 87% 45%	302 93% 40%	—

base pairings for the first sequence, in a CONNECT file format. For inverse folding, we consider stems that fold in both sequences only, and we do not take into account covariations.

The common folding computed by ReverseCARNAC is of course more accurate and somewhat faster. We folded once again the four 16S ssu rRNA sequences of the previous section, and gathered the new results in Table 5. A rather constant number of stems was predicted and the coverage and accuracy were quite better: about 85% of correctness, and only 40% of the structure is missed. The twelve foldings were achieved in less than 10 s.

IS IT POSSIBLE TO DETECT NON-CODING RNAS USING CARNAC ?

In the following, we try to evaluate whether CARNAC can be used to discriminate between families of functional structured RNAs and non-structured RNAs. When we have even a small set of homologous sequences, it is possible to use CARNAC to get an index of confidence for each stem found. This index can be used to check if these homologous molecules *actually* share a common structure. We applied the method to mRNA and enteroviruses.

Cytochrome mRNA versus RNase P RNA

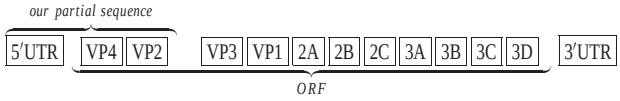
We have selected two distinct sets of sequences, of about the same length (300–350 bases). The first one (15 mRNA coding for cytochrome) is supposed to be unstructured, while the second one (15 RNase P RNA sequences from *Bacteria, Gamma subdivision*) is known to be structured (Brown, 1999). For each set, $\binom{15}{2} = 105$ pairwise comparisons were performed. The quality of anchor points was about the same, and also the number of potential stems, if we take into account bias in GC content.

There is no obvious way to distinguish mispredicted structure by looking only at the pairwise alignments of folded stems. But the use of a family (more than two sequences) permits us to compute, for each stem of a given sequence, its frequency *f*, i.e. the number of times it has been folded ($0 \leqslant f \leqslant 14$). Then we can look at the distribution of those frequencies: if the sequences share a common structure, some stems are expected with a frequency close to the maximum. This frequency provides each stem with an index of confidence: the higher the frequency, the more reliable the stem is. Figure 4 displays the result.

The distribution is clearly flatter for RNase P RNA, indicating that CARNAC detects an *overall* common structure: 63% of the stems appear in more than half of the foldings. On the contrary, for cytochrome mRNA, less than 1% of the predicted stems appear in more than half of the foldings. Thus CARNAC does not detect a strong common structure for mRNAs.

Enteroviruses

The enterovirus ORF is divided into different parts coding for a polyprotein. The 5′UTR has been shown to be structured (Witwer *et al.*, 2001; Le and Zuker, 1990), whereas structure is not expected within the ORF. We built a set of 12 enteroviruses partial sequences by keeping only the 5′UTR and the two first regions VP4 and VP2 (about 1800 bases long).



The sequences were pairwise folded and the frequencies were calculated as described before. Then we compared the frequencies of stems with closure either before or after the start codon (see Figure 5). As for the last comparison, the distribution of 5′UTR stem frequencies is flatter, indicating that a common structure is found whereas no *common* structure is detected in the remainder of the sequence.

CONCLUSION

CARNAC has been devised to handle any type of RNA homologous sequences. By looking for pairings at the level of stems, and by using anchor points, we managed to overcome the intrinsic complexity of the problem that is a usual limitation of many methods. The privilege of efficiency over accuracy, leaves some problems to solve. For instance, we discard noisy potential stems in the preselection step, to drastically reduce the huge search space. We also prohibit small overlaps between our maximal stems, while such overlaps actually arise, and

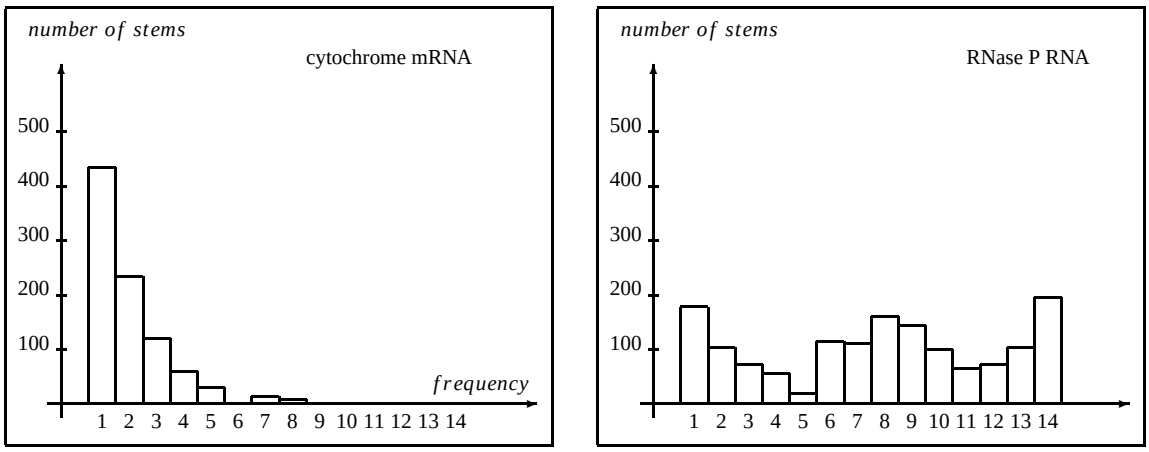


Fig. 4. Compared frequencies of RNase P RNA (right) and mRNA (left). Stems are despatched into classes corresponding to their frequency. For each bar of the histogram, the area is equal to the number of stems of that frequency.

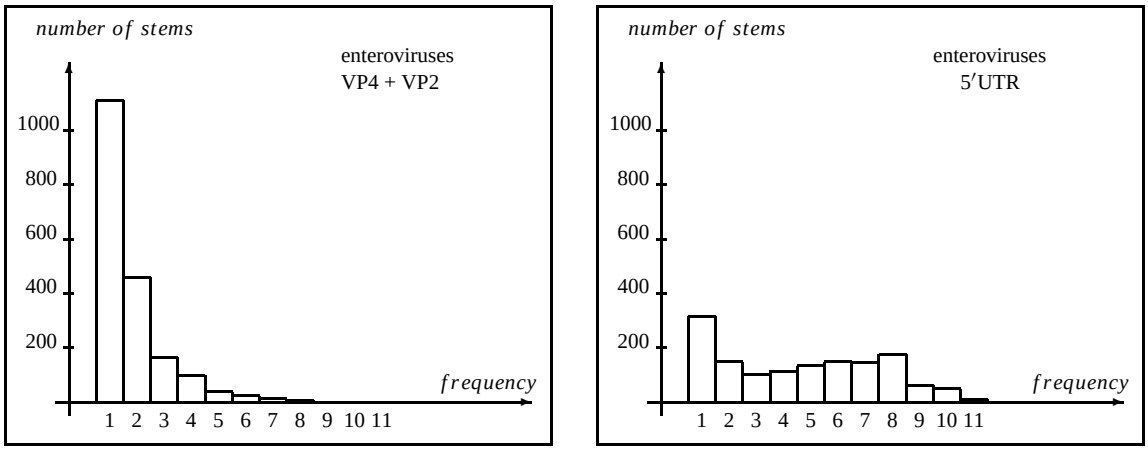


Fig. 5. Compared frequencies for enteroviruses: like in Figure 4, stems are despatched into frequency classes.

we consider only mismatch error types. Those restrictions imply that CARNAC is not able to recover the whole structure. However we believe that CARNAC is of great interest in a 'multi-step' strategy for structure discovering. At a first step, run CARNAC to get a reliable set of common stems. This partial prediction may then be used as a guide for any other complex prediction approach.

ACKNOWLEDGEMENTS

We would like to thank Daniel Gautheret and Cédric Notredame for the fruitful discussions we had about RNA folding and alignment. We are also grateful to the reviewers for their constructive remarks.

REFERENCES

Brown,J.W. (1999) The Ribonuclease P database. *NAR*, **27**, http://www.mbio.ncsu.edu/RNaseP/.

Chen,J.H., Le,S.Y. and Maizel,J.V. (2000) Prediction of common secondary structures of rnas; a genetic algorithm approach. *NAR*, **28**, 991–999.

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

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

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

Gorodkin,J., Heyer,L.J. and Stormo,G.D. (1997) Finding the most significant common sequence and structure motifs in a set of RNA sequences. *NAR*, **25**, 3724–3732.

Juan,V. and Wilson,C. (1999) RNA secondary structure prediction based on free energy and phylogenetic analysis. *JMB*, **289**, 935–947.

Knudsen,B. and Hein,J. (1999) RNA secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics*, **15**, 446–454.

Le,S.Y. and Zuker,M. (1990) Common structures of the 5′ noncoding RNA in enteroviruses and rhinoviruses. *JMB*, **216**, 729–741.

Mathews,D.H. and Turner,D.H. (2002) Dynalign: an algorithm for finding the secondary structure common to two RNA sequences. *JMB*, in press.

Mathews,D.H., Sabina,J., Zuker,M. and Turner,D.H. (1999) Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. *JMB*, **288**, 911–940.

Mathews,D.H., Turner,D.H. and Zuker,M. (2000) RNA secondary structure prediction. *Current Protocols in Nucleic Acid Chemistry*, **11**, 1–10.

Nussinov,R. and Jacobson,A.B. (1980) Fast algorithm for predicting the secondary structure of single stranded RNA. *PNAS*, **77**, 6309–6313.

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

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

Tinoco,I. and Bustamante,C. (1999) How RNA folds JMB. **293**, 271–281.

Wang,Z. and Zhang,K. (1999) Finding common RNA secondary structures from RNA sequences. *Lecture Notes in Computer Science, 1645*. pp. 258–269.

Witwer,C., Rauscher,S., Hofacker,I.L. and Stadler,P.F. (2001) Conserved RNA secondary structures in picornaviridae genomes. *NAR*, **29**, 5079–5089.

Woese,C.R., Magrum,L.J., Gupta,R., Siegel,R.B., Stahl,D.A., Kop,J., Crawford,N., Brosius,J., Gutell,R., Hogan,J.J. and Noller,H.F. (1980) Secondary structure model for bacterial 168 ribosomal RNA: phylogenetic enzymatic and chemical evidence. *NAR*, **24:8**, 2275–2293. http://www.rna.icmb.utexas.edu/.

Wu,M. and Tinoco,I. (1998) RNA folding causes secondary structure rearrangement. *PNAS*, **95**, 11555–11560.

Zuker,M. (1989) On finding all suboptimal foldings of an RNA molecule. *Science*, **244**, 48–52.

APPENDIX: SEQUENCE REFERENCES

name (acc. number, when available)

16S ssu rRNA *M. jannaschii* (U67517), *S. solfataricus* (X03235), *E. coli* (J01695), *B. subtilis* (K00637).

*RNase P RNA (*Delta/Epsilon Purple Bacteria subd.*)* *D. desulfuricans* (M59357), *D. vulgaris*, *G. sulfurreducens*, *C. jejuni* (AL139075), *H. pylori* (AE000573).

*RNase P RNA (*Gamma Purple Bacteria subd.*)* *A. ferrooxidans* (X16580), *Buchnera-APS* (AP001118), *C. vinosum* (M59356), *H. influenza* (U32848), *K. pneumoniae* (M32719), *E. coli* (V00338), *E. agglomerulans* (M33657), *P. aeruginosa* (AF295983), *P. fluorescens* (M19024), *S. typhi*, *S. typhimurium* (M10889), *S. marcescens* (M33658), *S. putrefaciens*, *Y. pestis*, *X. fastidiosa* (AE003849).

cytochrome mRNA *Arabidopsis thaliana* (X59459), *Candida glabrata* (X58249), *Chlamydomonas reinhardtii* (Z99829), *Curvularia lunata* (AY034827), *Drosophila melanogaster* (X01761), *Emericella nidulans* (M83141), *Fusarium oxysporum* (AB033762), *Fritillaria agrestis* (AF031540), *Gallus gallus* (K02303), *Helianthus annuus* (L77113), *Homo sapiens* (BC009602), *Mus musculus* (AK005336), *Neurospora crassa* (X05506), *Rice (Oryza sativa)* (D12634), *Saccharomyces cerevisiae* (L22173), *Tigriopus californicus* (AF091460).

Enteroviruses *POLIOS1* (V01150), *POL2LAN Poliovirus type 2 Lansing strain* (M12197), *PIPOLS2 Poliovirus type 2* (X00595), *PI3L37 Poliovirus P3/Leon/37 type 3* (K01392), *PIP03XX Poliovirus type 3 strain 23127* (X04468), *HPO293918 Human poliovirus type 3* (AJ293918), *CA05876 Human coxsackievirus A16* (U05876), *Human coxsackievirus A16* (AF177911), *CXB1G Coxsackievirus B1* (M16560), *Coxsackievirus B2 strain Ohio* (AF081485), *Edwards CB4 human strain* (S76772), *Coxsackievirus B5 strain Faulkner* (AF114383), *Coxsackievirus B6 strain Schmitt* (AF039205), *CXB9CG Coxsackievirus A9 strain Griggs* (D00627), *Echovirus 1 strain Farouk* (AF029859), *Echovirus 5* (AF083069), *EC12TCGWT Echovirus type 12* (X79047), *EV70CG Enterovirus 70* (D00820), *BEVVG527 Bovine enterovirus* (D00214), *Bovine enterovirus isolate K2577* (AF123432), *Bovine enterovirus isolate SL305* (AF123433), *BENTRM2 Bovine Enterovirus RM-2* (X79369), *PEV9XX Porcine Enterovirus 9* (Y14459), *A-2 plaque virus A2* (AF201894),