Automated prediction of three-way junction topological families in RNA secondary structures

Alexis Lamiable[a,b], Dominique Barth[a], Alain Denise[b,c,e,*], Franck Quessette[a], Sandrine Vial[a], Éric Westhof[d]

[a] *PRiSM, Univ. de Versailles St-Quentin Versailles, France*
[b] *LRI, Univ. Paris-Sud 11 and CNRS, Orsay, France*
[c] *IGM, Univ. Paris-Sud 11 and CNRS, Orsay, France*
[d] *Architecture et Réactivité de l'ARN, Université de Strasbourg, IBMC du CNRS Strasbourg, France*
[e] *INRIA Saclay, France*

Abstract

We present an algorithm for automatically predicting the topological family of any RNA three-way junction, given only the information from the secondary structure: the sequence and the Watson-Crick pairings. The parameters of the algorithm have been determined on a data set of 33 three-way junctions whose 3D conformation is known. We applied the algorithm on 53 other junctions and compared the predictions to the real shape of those junctions. We show that the correct answer is selected out of nine possible configurations 64 % of the time. Additionally, these results are noticeably improved if homology information is used. The resulting software, Cartaj, is available online and downloadable (with source) at: http://cartaj.lri.fr

Keywords: RNA tertiary structure, three-way junctions, prediction

1. Introduction

RNA molecules fold into complex three-dimensional structures in a hierarchical and modular way, with recurring autonomous building blocks being packed together to form the molecule. These modules are also hierarchical: high level modules, like RNA junctions, are made of smaller, lower level modules, in this case of Watson-Crick helices linked together by single strands.

Knowledge of the shape of the lower level modules can give us insight on the shape of the higher level ones, leading to an approximation of the shape of the molecule that can be refined in subsequent steps. Since Watson-Crick helices have a well-defined shape, RNA junctions are the next obvious target (Bindewald et al., 2008; Lescoute et al., 2005; Lescoute and Westhof, 2006;

*Corresponding author
Email address: alain.denise@u-psud.fr (Alain Denise)

Laing and Schlick, 2009). Notably, in Lescoute and Westhof (2006), the authors have showed that the three-way junctions where two helices are approximately stacked can be divided in three families A, B and C, according to the position of the third helix (P3) relatively to the two other helices that are stacked together (P1 and P2). Figure 1 shows a schematic drawing of each of the families.

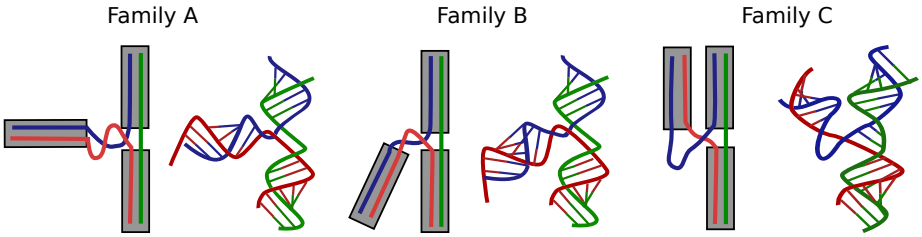

Figure 1: Schematic drawings of the families A, B and C (from Lescoute and Westhof (2006)).

The topology of each of the families is notably due to the different non Watson-Crick interactions that occur within the helices, and between the helices and the other nucleotides of the junction. After a thorough examination of 33 junctions whose three-dimensional structure was known, Lescoute and Westhof gave some hints towards predicting the family of a junction, given its secondary structure.

In this paper, we propose a method for automatically predicting the topological family of any given three-way junction, with only information from sequence and the deduced secondary structure (only Watson-Crick interactions). We also show that the accuracy of the prediction is noticeably improved if homology information is given in addition, that is a set of sequences that are homologous to the input sequence. We evaluate the accuracy of our method on a set of 86 junctions from the structural databases, and we compare it to other possible approaches.

2. Materials and Methods

2.1. Data

We distinguished the following three data sets:

LW: The 33 junctions from Lescoute and Westhof (2006).

FR3D: In order to test our predictions, we automatically extracted the three-way junctions from all molecules in the non-redundant FR3D database (Sarver et al., 2008). We found 86 junctions, among them 53 were new junctions – "new" being defined here as having a secondary structure different from that of the junctions in LW. Details on the extraction process are given in the next subsection.

ALL: This dataset includes all the 86 different junctions from the previous two datasets.

Table 1 shows the number of junctions from each family in each of the datasets.

Set	A	B	C	Total
LW	10	7	16	33
FR3D	16	13	24	53
ALL	26	20	40	86

Table 1: Number of junctions from families A, B, and C in the data sets.

2.2. Extraction of junctions

Starting with the PDB files taken from the non redundant FR3D Database (Sarver et al., 2008), we extracted the secondary structure of the molecules with RNAView (Yang et al., 2003) and removed the pseudoknots with K2N (Smit et al., 2008) using the default method. Then, we extracted all three-way junctions. We defined a three-way junction as a junction between three helices, considering as in Waugh et al. (2002) that an helix must contain at least two consecutive (Watson-Crick) base-pairs.

A special attention was given to what we call *homologous* junctions. We say that two junctions are *homologous* if they appear at the same place in the three-dimensional structures of two homologous RNAs. This is the case, for example, for the the junctions 1J5E_001 and 2AVY_29 that appear, respectively, in the 16S RNAs of *T. thermophilus* and *E. coli* (see Supplementary Material). We kept every instance of *homologous* junctions that differ in sequence or secondary structure.

Finally, we excluded the junction from 3E5C (SAM-III riboswitch) because of a ligand that changes its configuration. We also excluded one junction from 2A64 (bacterial RNase P) because one of its unpaired strands formed a pseudoknot with another part of the molecule.

Altogether, we found 86 junctions that can be clustered into 39 classes of *homologous* junctions. Table 4 gives the 26 classes that contain more than one junction. More details can be found in the Supplementary Material where all the junctions are grouped by families (A, B or C) then by *homology*, and their sequences and secondary structures are given.

2.3. Prediction workflow

Figure 2 gives the general outline of the prediction workflow. For a given three-way junction there are three possible stackings and, for each stacking, three possible families. Hence, we have to choose between nine different configurations. We assign a score to each of these configurations. This score is denoted s_c for any given configuration c. It is a linear combination of four

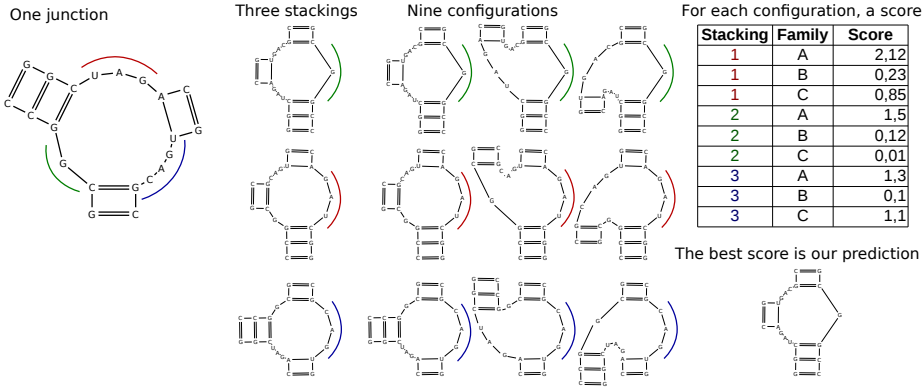

Figure 2: Prediction workflow. A given junction can have three stackings; for each stacking, the junction can be in three families (A, B or C), depending on the angle of the third helix. This gives nine configurations. We compute a score for each configuration (see figure 3), and the configuration with the best score is our prediction.

partial scores, denoted $s_{c,i}$ for $i \in \{1, 2, 3, 4\}$. Each of these partial scores is computed according to a specific criterion. Indeed, according to Lescoute and Westhof (2006), classification of three-way junctions strongly depends of a small number of parameters. We consider only four parameters of sequence and (secondary) structure, as shown in figure 3, excluding any tertiary structure information. Finally, we take the configuration with the best score as our best putative prediction.

Here are the four criteria and how their associated scores are computed, roughly. The precise formulas are given in the supplementary material.

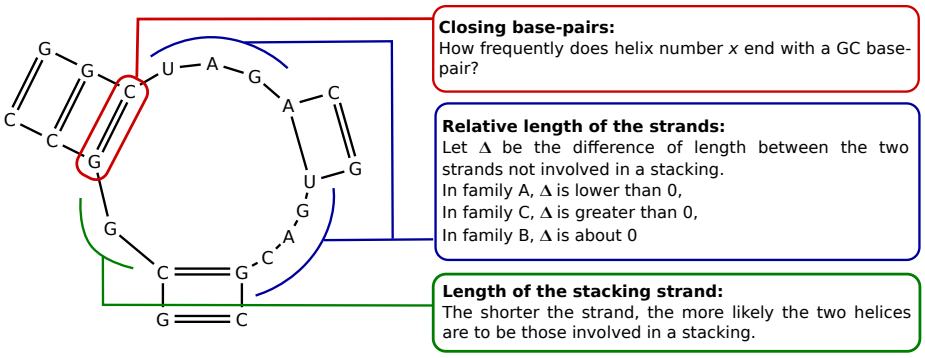

Figure 3: The different criteria used to evaluate a configuration. Each criterion is given a numerical score, and those scores are aggregated together in order to produce a total score for the considered configuration.

$s_{c,1}$: Length of the stacking strand. When two helices are coaxially stacked, it is usually those connected by the shortest strand. Hence, for a given configuration, the shortest the strand of the two stacked helices, the better the score. If a strand has length zero, we give it an even better score, as there is no counter-example in the Lescoute-Westhof data set.

$s_{c,2}$: Relative length of the strands. In a given configuration, one strand is involved in the stacking. Let the other two strands be called *below* and *above*. In family A, the *below* strand is longer than the *above* one, and vice-versa in family C. In family B, the two strands are about the same length. We compute Δ, the difference between the length of *above* and that of *below*. The value of Δ determines the score for a given configuration.

$s_{c,3}$: Closing base-pairs of the helices. We numbered the helices of a junction according to the transcription order (the first helix is the one that contains the lowest nucleotide id). We then computed the frequency of each type of closing base-pair in the helices of each family, and built a score matrix for each base pair. The score $s_{c,3}$ depends on this matrix.

$s_{c,4}$: Bonus criteria. We take into account some possibilities of tertiary links, as the presence of two consecutive adenines in a strand that could form an A-minor motif, as is often observed in family C.

2.4. Score computation

For a given criterion i, the nine configurations can be ranked by decreasing scores $s_{c,i}$. Let $r_{c,i}$ be the rank of configuration c for the criterion i. The global score for configuration c is then defined as:

$$s_c = \frac{\sum_i w_i s_{c,i}}{\sum_i r_{c,i}}$$

where w_i is a constant weight assigned to each criterion i.

We divide the sum of the scores by the sum of the ranks as a way to ensure that, in order to have a high total score, a configuration must be good in several tests, relatively to the other possible configurations. For example, for a given configuration, if the sum of its scores is 42 and its ranks are 1, 3, 3 and 3, it gets a total score of 4.2, but if its ranks are 1, 1, 1 and 1, it gets a total score of 10.5. That way, a good score in one test doesn't outweight mediocre scores in the other tests.

The values of the weights w_i have been fixed as follows. We explored all possible combinations of the w_i's between 0 and 4 by step of 0.5. We picked the combination that provided the best results on the Lescoute-Westhof data set. This gave: $w_{c,1} = w_{c,2} = w_{c,4} = 1$ and $w_{c,3} = 2$.

3. Results and Discussion

3.1. Prediction result

For the three data sets described in section 2.1, we computed how many times we predicted the correct answer in the first position, or in the first three

positions among the nine possible ones (see table 2). Randomly selecting a configuration would produce the correct answer in the first position in 11 % of the cases, and in the first three positions in 33 % of the cases. We predict the

Data set	Size	First position	First 3 positions
LW	33	20 (60.6 %)	31 (93.4 %)
FR3D	53	35 (66.0 %)	44 (83.0 %)
ALL	86	55 (64.0 %)	75 (87.2 %)

Table 2: Quality of the predictions on the three datasets. The correct configuration gets the best score 64 % of the time. Moreover, 87 % of the time, the correct configuration is among the best 3 scores.

correct configuration in the first position 64 % of the time, and in the first three positions 87 % of the time.

Looking at these results, one could think that the first test conclusively predicts the stacking, and that the difficulty lies in predicting the family, because the first three configurations correspond to the families A, B and C of that stacking. This is not the case; the correct answer being in the second or third position does not imply that the first three configurations have the same stacking.

Therefore, having additional knowledge about the stacking would help us to decide between those cases. We tried using the approach from Tyagi and Mathews (2007) to predict the stacking. One shortcoming of that method is that it can only work on junctions with a relatively low number of unpaired bases; a strand must contain less than two unpaired nucleotides for the nearest-neighbor model to be able to predict a stacking. In our application, this approach correctly predicted 8 stackings (25 % of the stackings) in the Lescoute-Westhof data set, and failed to predict the remaining 26 ones. Those 26 false negative are due in part to the limitation mentioned previously, in part to errors of the method, and in part to differences in our definitions of stacked helices (see the discussion in Tyagi and Mathews (2007, p944)).

In the 8 positive cases, we already predicted the correct stacking with our simple strand length criterion. Using stacking predictions from Tyagi and Mathews did not improve our prediction, suggesting that our simple criterion is "good enough" in first approximation.

We also tried to use MC-Fold (Parisien and Major, 2008) to predict tertiary interactions inside our junctions. Our approach does not take tertiary interactions into account, but we hoped that they could at least help us predict the stacking by reducing the lengths of the unpaired strands. We took the junctions from the LW dataset, constrained their Watson-Crick interactions, and applied MC-Fold on them.

The first observation was that we found the exact real junction among the best 10 candidates given by MC-Fold only once, because the original junction was very constrained and contained few tertiary interactions. Therefore, we

cannot expect MC-Fold to produce a perfect tertiary structure for the junctions, but we can hope that it provides enough hints to help us decide on the stacking.

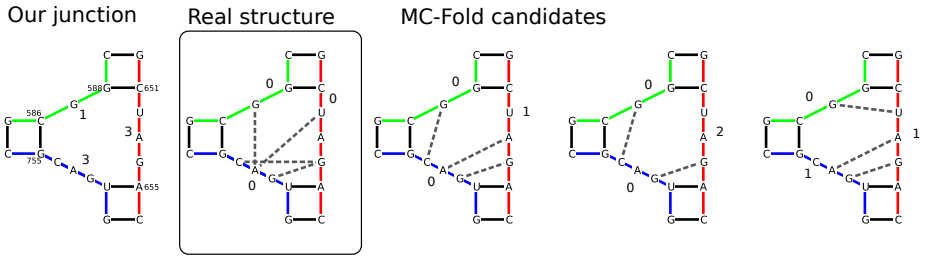

Figure 4: Secondary structure of junction 1J5E_001, its tertiary structure extracted from the PDB, and three of the first MC-Fold candidate structures. The numbers indicate the lengths of the unpaired strands that we used to compute the stacking scores. We consider that MC-Fold predictions help us when the length of the strand involved in the stacking is lowered, comparatively to the other lengths. In the example above, the length of the red strand is lowered from 3 to 2 or 1, but the length of the green strand is lowered to 0, producing a much higher score for that strand.

We put 20 junctions through MC-Fold and manually looked at the best 3 candidates for each of them. Instead of having only Watson-Crick helices, we assumed that tertiary interactions were part of the helices, and then we checked if using those "extended" helices improved the stacking score of the correct configuration, comparatively to the other configurations. When that was the case, then we said that tertiary information "helped" us, whether our final prediction was correct or not (see figure 4). In those 20 junctions, using the tertiary information extracted from the PDB files helped us 6 times, using the first MC-Fold candidate helped us 1 time, and using any one of the best 3 MC-Fold candidates helped us 3 times, but it was assumed that we knew which candidate to pick. A common problem was that MC-Fold often produced several strands of lengths 0, thus predicting all of them being a good stacking candidate.

These results suggest that MC-Fold does not improve our approach by helping us on the strand lengths criterion. It is possible that it might help us if we introduced tertiary criteria, which was not within the scope of our work, but we still would have to choose among the several candidates.

In table 3, we consider our classification method as binary yes/no tests that determine whether a junction belongs to a given family. We show the positive predictive value (PPV), specificity and sensitivity of those tests:

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

Family	PPV	Specificity	Sensitivity
A	0.60	0.67	0.85
B	0.44	0.94	0.20
C	0.83	0.58	0.74

Table 3: Positive predicting value (PPV), specificity and sensitivity of our classification, considering binary tests such as "does this junction belong to family A?", on the complete dataset. Random choice between the 3 families would give a PPV of 0.33, a specificity of 0.66 and a sensitivity of 0.33.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

where TP stands for "true positives", TN for "true negatives", FP for "false positives" and FN for "false negatives".

The overall results are good for families A and C, with, in particular, a PPV much higher than what could be expected with a random choice. The high specificity and low sensitivity of the test for family B shows that we are rarely wrong, but also rarely right, meaning that we rarely predict a junction as being in family B at all. It is the least well-defined of the three families and we lack decisive criteria: it has no bonus score $s_{c,4}$, but trying to compensate by adding a fixed bonus score, or by ignoring the bonus criteria completely does not improve the overall predictions. It is interesting to note that Family C is the most prevalent family and Family B the rarest one.

3.2. Improving the predictions by using sequence homology

We consider here the problem where the entry data are not a single junction, but a set of n *homologous* junctions $\{J_0, \ldots, J_n\}$. therefore they are supposed to be in the same topological family.

In this case we compute a global score for the set, as follows. Let $s_{c,i}(k)$ the score for configuration c and criterion i of the junction J_k. Then we set:

$$S_{c,i} = \frac{\sum_j s_{c,i}(j)}{n}$$

and the global combined score for a configuration c is:

$$S_c = \frac{\sum_i w_i S_{c,i}}{\sum_i r_{c,i}}$$

In other words, we compute, for each test, the mean of the scores of each junction, and then compute the final score as we did previously.

We grouped the junctions in clusters according to homology, as detailed in Section 2.2. We considered only the 26 clusters containing more than one junction. Those clusters are given in table 4. The column "Sep." shows how many junctions have been correctly predicted, separately from the others, by using the classical scoring function. In 8 clusters (31 %), one junction is predicted

Id	Junctions	Fam.	Sep.
1	1J5E_001, 2AVY_29	A	**1/2**
2	1J5E_002, 2AVY_32	A	**1/2**
3	1J5E_003, 2AVY_35	A	2/2
4	1J5E_004, 2AVY_47	A	**1/2**
5	1J5E_006, 2AVY_40	B	0/2
6	1J5E_008, 2AVY_3	C	2/2
7	1J5E_011, 2AVY_49	C	2/2
8	1J5E_007, 2AVY_42	B	2/2
9	1S72_001, 2AW4_5, 2ZJR_1	A	3/3
10	1S72_002, 2AW4_10, 2ZJR_4	A	3/3
11	1S72_003, 2AW4_50, 2ZJR_47	A	3/3
12	1S72_004, 1FG0_7, 2AW4_51	A	**2/3**
13	2ZJR_101, 2AW4_109	B	2/2
14	1S72_007, 2AW4_28, 2ZJR_24	B	0/3
15	1S72_008, 2AW4_52, 2ZJR_49	B	0/3
16	1S72_010, 2AW4_21, 2ZJR_0	C	3/3
17	1S72_011, 2AW4_18, 2ZJR_10	C	**2/3**
18	1S72_012, 2AW4_27, 2ZJR_21	B	0/3
19	1S72_013, 2AW4_99, 2ZJR_92, 1FG0_14	C	4/4
20	1HC8_001, 2AW4_44, 1S72_47, 1QA6_0, 1MMS_0, 2ZJR_41	C	6/6
21	1S72_005, 2AW4_86, 2ZJR_82	B	3/3
22	1S72_014, 2AW4_0, 2ZJR_106, 1UN6_0	C	**1/4**
23	1E8O_001, 1MFQ_001, 1LNG_0	C	**2/3**
24	1U8D_001, 1Y26_0, 2EET_0	C	**2/3**
25	1MME_001, 2QUS_0, 2OEU_0	C	3/3
26	3D2G_0, 2HOJ_0	B	0/2

Table 4: The 86 junctions clustered by *homology*. Each line contains a cluster of homologous junctions (clusters containing a single junction are not shown). The "Sep." column shows how many of the junctions in that cluster are separately predicted correctly, separately; the value is in bold face for the *contradictory* clusters. Clusters that are correctly predicted by the global scoring function S_c are shown with a grey background. Note that there is only one contradictory cluster without a gray background (ie. badly predicted), number 23.

in a wrong configuration despite the fact that we got the correct configuration for some others. Let us us call these clusters the *contradictory* clusters. In the remaining 18 clusters, all the junctions are either correctly predicted or badly predicted. It is to be noted that the 5 clusters that we completely fail to predict are in family B, confirming that our classification lacks decisive criteria for that family.

Rows with a gray background in the table show a good prediction result for the cluster when the global scoring function S_c defined above is used. Seven out of the eight contradictory clusters are now subject to the right prediction; the remaining cluster is now badly predicted. Comparison in prediction results with or without homology information is provided in table 5.

In most real cases, we are not given a set of junctions with the secondary

	Without homology	With homology
All correct	17	24
Contradictory	8	0
None correct	5	6
Junctions correct	50	57

Table 5: Using homology knowledge improves the predictions where homologous junctions were not predicted in a consistent way, but does not help when they were predicted consistently wrong.

structures deduced from the crystallographic tertiary structures, but most generally an approximate sequence alignment providing a consensus structural arrangement hopefully close to the real secondary structure. How does our approach handle these cases? To assess this, we used alignments from the Comparative RNA Web site (Cannone. et al., 2002). For each ribosomal junction J of our dataset, we randomly picked 500 aligned sequences, used the refined secondary structure from J for all these alignments, and applied our prediction method using the average score S_c. We ensured that the number of sequences we picked was big enough to be representative of all the available sequences in the alignments. Out of the 5 contradictory clusters of ribosomal junctions, 2 or 3 of them are now correcly predicted, depending on the structure we used when we had several choices.

4. Conclusion

We described an automated method for predicting the topological family of three-way RNA junctions from their secondary structure only. We showed that this approach works well on single junctions, and is improved either by having sequence alignements for that junction or, even better, a set of homologous 2D structures deduced from crystallographic data. Among the three topological families, Family B, the rarest one, is the less well predicted by our method, showing that new criteria have to be found for that family.

The Cartaj software (http://cartaj.lri.fr) that implements our method can be used as it. It is also meant for being part of RNA modelling softwares and platforms.

Acknowledgement

We warmly thank Julie Bernauer and Fabrice Jossinet for fruitful discussions. We also thank Christian Cadéré and Vincent Reinhard for their help at an early stage of the work.

This research was supported in part by the Digiteo project PASAPAS, by the ANR project AMIS ARN ANR-09-BLAN-0160, and by the UniverSud Paris project PASAPRES.

References

E. Bindewald, R. Hayes, Y. Yingling, W. Kasprzak, and B. A. Shapiro. RNA-Junction: A Database of RNA Junctions and Kissing Loops For Three-Dimensional Structural Analysis and Nanodesign. *Nucleic Acids Research*, 36:392–397, January 2008.

J.J. Cannone., S. Subramanian, M.N. Schnare, J.R. Collett, L.M. D'Souza, Y. Du, B. Feng, N. Lin, L.V. Madabusi, K.M. Müller, N. Pande, Z. Shang, N. Yu, and R.R. Gutell. The Comparative RNA Web (CRW) Site: An Online Database of Comparative Sequence and Structure Information for Ribosomal, Intron, and Other RNAs. *BioMed Central Bioinformatics*, 3(2), 2002.

C. Laing and T. Schlick. Analysis of four-way junctions in RNA structures. *J. Mol. Biol.*, 390:547–559, 2009.

A. Lescoute and E. Westhof. Topology of three-way junctions in folded RNAs. *RNA*, 12:83–93, 2006.

A. Lescoute, N. B. Leontis, C. Massire, and E. Westhof. Recurrent structural RNA motifs, Isostericity Matrices and sequence alignments. *Nucleic Acids Research*, 33(8):2396–2409, April 2005. doi: 10.1093/nar/gki535.

M. Parisien and F. Major. The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature*, 452(7183):51–55, March 2008.

M. Sarver, C. L. Zirbel, J. Stombaugh, A. Mokdad, and N. B. Leontis. FR3D: Finding Local and Composite Recurrent Structural Motifs in RNA 3D Structures. *Journal of Mathematical Biology*, 56:215–252, 2008.

S. Smit, K. Rother, J. Heringa, and R. Knight. From knotted to nested RNA structures: A variety of computational methods for pseudoknot removal. *RNA*, 14(3):410–416, March 2008. doi: 10.1261/rna.881308.

R. Tyagi and D. H. Mathews. Predicting helical coaxial stacking in RNA multibranch loops. *RNA*, pages 939–951, July 2007. doi: 10.1261/rna.305307.

A. Waugh, P. Gendron, R. Altman, J. W. Brown, D. Case, D. Gautheret, S. C. Harvey, N. Leontis, J. Westbrook, E. Westhof, and et al. RNAML: A standard syntax for exchanging RNA information. *RNA*, 8:707–717, 2002.

H. Yang, F. Jossinet, N. Leontis, L. Chen, J. Westbrook, H. Berman, and E. Westhof. Tools for the automatic identification and classification of RNA base pairs. *Nucl. Acids Res.*, 31(13):3450–3460, July 2003.