RNA Pseudoknot Folding through Inference and Identification Using TAG_{RNA}

Sahar Al Seesi, Sanguthevar Rajasekaran, and Reda Ammar

Computer Science and Engineering Department, University of Connecticut
{sahar,rajasek,reda}@engr.uconn.edu

Abstract. Studying the structure of RNA sequences is an important problem that helps in understanding the functional properties of RNA. After being ignored for a long time due to the high computational complexity it requires, pseudoknot is one type of RNA structures that has been given a lot of attention lately. Pseudoknot structures have functional importance since they appear, for example, in viral genome RNAs and ribozyme active sites. In this paper, we present a folding framework, TAG_{RNA}Inf, for RNA structures that support pseudoknots. Our approach is based on learning TAG_{RNA} grammars from training data with structural information. The inferred grammars are used to indentify sequences with structures analogous to those in the training set and generate a folding for these sequences. We present experimental results and comparisons with other known pseudoknot folding approaches.

1 Introduction

Many new functional RNAs, such as miRNAs and tmRNAs [3] [20] [34] have been discovered in recent years. This resulted in speeding up RNA structural analysis and determination. Another factor that has led acceleration of RNA structural research is the rise of the RNA World Hypothesis [10] which suggests that the current DNA and protein world have evolved from an RNA based world. Analysis of the structures of RNA sequences is essential in understanding their functional properties. Consequently, it is imperative for creating new drugs and understanding genetic diseases [6] [24]. Computational methods can provide less expensive solutions to structure analysis than other methods such as nuclear magnetic resonance and x-ray crystallography.

Most RNA structure analysis research can be classified into thermodynamic or comparative approaches. Thermodynamic approaches use dynamic programming to compute the secondary structure with the minimum free energy (mfe) [13] [35]. These approaches use experimentally determined parameters for free energies. Comparative approaches are based on aligning a set of homologous sequences and computing the structure based on the alignment [8] [12]. Recently, a new approach for RNA structure analysis based on grammatical formalisms has emerged. This approach was inspired by David Searls work in the early 90's where he studied the linguistics of biological sequences [28]. He suggested the use of formal grammars as a tool to model and analyze DNA, RNA, and proteins. The use of grammars has attracted the attention of many researchers [26] [31] because it can model long range interactions. In

S. Rajasekaran (Ed.): BICoB 2009, LNBI 5462, pp. 90–101, 2009.

addition, grammatical models are concise and easy to understand representation of structures of sequence families.

A secondary structure for a sequence of length n is a list of base-pairs (bps) in the form (i, j) where $1 \le i, j \le n$. Two bps (i, j) and (k, l) are nested if $i < k$ and $l < j$. Two bps are crossing if $i < k$ and $j < l$. Pseudoknot is one type of RNA structures that exhibits crossing bps. It has been proven that predicting RNA structures with pseudoknots using free energy minimization is an NP-complete problem [17]. Also, pseudoknots cannot be modeled with Context Free Grammars (CFG) due to the crossing dependencies of their bps. Consequently, until recently, pseudoknots were ignored in RNA secondary structure analysis. Pseudoknot structures have functional importance since they appear, for example, in viral genome RNAs [19], ribozyme active sites [30], and tmRNA [34]. Among the available research in analyzing pseudoknot structures are the works of Akutsu [2], Dirks and Pierce [9], Rivas and Eddy [23], the iterated loop matching algorithm (ILM) by Ruan *et. al*. [25], and pknotsRG by Reeder and Giegerich [22].

One of the proposed grammatical models that support pseudoknots is TAG_{RNA}. TAG_{RNA} is a submodel of Tree Adjoining Grammars (TAG) [14]. It was proposed by Uemura *et. al*. [31]. They developed a parser for their model, and presented experimental results for using the model to fold RNA sequences with pseudoknot structures. Our solution is based on the TAG_{RNA} model.

Our solution is a grammatical inference approach to RNA structure analysis. Among the research that uses grammatical inference in bioinformatics are the works of Brazma *et. al.* [5], Laxminarayana *et. al.* [16], Takakura *et. al.* have published [29]. Brazma *et. al.* [5] have proposed an approach to discover simple grammars for families of biological sequences, using a subclass of regular grammars. On the use of grammatical inference to analyze RNA structures with Pseudoknots, Laxminarayana *et. al.* [16] presented an inference algorithm for Terminal Distinguishable Even Linear Grammars (TDELG), and they have shown how to use this algorithm in an Infer-Test model for the detection of a pseudoknot structure in an RNA sequence. They address the same problem we addressed in [1]. Takakura *et. al.* [29] use alignment data to infer probabilistic TAG_{RNA}. They use the inferred grammar to find new members of ncRNA families. Sakakibara has published [27] in which he discusses the general merits of using grammatical inference in bioinformatics.

The use of grammatical inference to automate the grammar building step is essential in facilitating the use of grammatical formalism by biologists. Otherwise, the biologist will always be dependent on a grammar expert. In [1], we presented a grammatical inference engine for TAG_{RNA}. We also presented a structure identification framework, where the inferred grammars can be used to answer the question of whether an RNA input sequence exhibits a certain structure or not. In this paper, we present a modification of the framework which is capable of folding[1] an RNA sequence with identification as a first step in folding. We test our solution on RNA sequences from Pseudobase [4], Rfam [11] , and the tmRNA database [34], and we compare our results with a representative subset of the available tools that are capable of folding RNA sequences including pseudoknots. We compare our results with ILM

[1] We use the terms structure prediction and folding interchangeably.

and, pknotsRG. PknotsRG is an algorithm for folding RNA sequences under the mfe model. It requires $O(n^4)$ time and $O(n^2)$ space. The ILM algorithm is based on the loop matching algorithm [18], and it also utilizes thermodynamic parameters. The worst case time complexity of ILM is $O(n^4)$ and the space complexity is $O(n^2)$. We also compare our results with TAG_{RNA} which our solution is based on. The folding approach provided in [31] using TAG_{RNA} is based on single generic grammar. Our approach is different because we infer specific grammars and use them to do identification and folding. TAG_{RNA} has time and space complexity of $O(n^5)$ and $O(n^4)$ respectively.

2 TAG and TAG_{RNA}

Tree Adjoining Grammars (TAGs) were introduced by Joshi *et. al.* [14]. Uemura *et. al.* [31] defined a subclass of TAGs, $TAG_{RNA,}$ suitable for modeling RNA pseudoknot structures. In this section, we describe TAG and TAG_{RNA}.

A Tree Adjoining Grammar (TAG) is defined to be a 5-tuple $(T \cup \{\varepsilon\}, N , I, A, S)$, where T is a set of terminal symbols, N is a set of non-terminal symbols, ε is the empty string symbol, and S is the starting symbol. I and A are defined as follows:

I (initial trees): A finite set of finite trees with the internal nodes' labels belonging to $N \cup \{S\}$, the leaves' labels belonging to $T \cup \{\varepsilon\}$, and the root is labeled with S.

A (auxiliary trees): A finite set of finite trees with the internal nodes' labels belonging to $N \cup \{S\}$, and the leaves' labels belonging to $T \cup \{\varepsilon\}$ except one leaf node which has the same label as the root. This special leaf node is called a foot node.

$I \cup A$ constitutes the set of elementary trees. An operation called the adjoining operation can be used to compose two trees, resulting in a derived tree. The adjoining operation composes an auxiliary tree β with a foot node labeled X with any other tree α, elementary or derived, that has some internal node with the same label X. The adjoining operation works as follows: starting with the tree α, extract the sub-tree rooted at the internal node labeled with X (let that sub-tree be γ), and replace it with β. Then at the foot node of β, γ is reinserted. The adjoining operation is illustrated in Fig. 1. Let $T = \{ t : \exists i \in I$ s.t. t can be derived from i$\}$, then L(TAG) consists of the yield of all the trees in T.

In [31], Simple Linear TAG (SLTAG) and Extended Simple Linear TAG (ESLTAG) are defined to be two subclasses of TAG with adjoining constraints [33]. In these two subclasses, the adjoining operation can occur only at internal nodes tagged

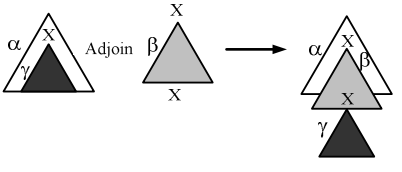

Fig. 1. The Adjoining Operation

with the symbol *, and the number of these nodes is restricted to one in SLTAG and two in ESLTAG. TAG_{RNA} is a sub-class of ESLTAG where only five types of elementary trees are allowed (Fig. 2). Each type of tree is responsible for a specific kind of branching or structural form that an RNA sequence can have.

3 The Structure Identification/Prediction Framework

In [1], we presented TAG_{RNA}Inf; an RNA structure identification framework. By structure identification we mean, given an RNA sequence, we answer the question of whether it exhibits a certain structure or not. TAG_{RNA}Inf, has a training phase in which a grammatical inference engine is fed with a positive training set with structural information. The inference engine will generate a grammar for each unique structure pattern in the sample. Then, the same sample along with a negative sample and the grammar(s) generated by the inference algorithm will go through an ESLTAG parser. For each input sequence the parser will output a score. We use the maximum number of base pairs as the scoring function. The scores will be the input to a threshold function inference module. This module infers a score threshold function Th(l) = p. A sequence s of size l is considered to have the RNA structure represented by a grammar G iff the parser accepts s under G, with score $p_s \geq p$. Th(l) is a step function defined as follows:

$$Th(l) = p \qquad , \; i \leq l < j \tag{1}$$

The threshold function inference module infers a function that maximizes the sum of sensitivity and specificity for the training data using dynamic programming. The time and space complexity for inferring the threshold function are $O(n^3m^2)$ and $O(n^2m^2)$, where n is the maximum sequence size and m is maximum reported score for the training data set. While inferring the threshold function, this module also selects the most informative grammars. As mentioned earlier, the grammatical inference engine generates a grammar for each unique structure pattern it encounters. Here, nearly redundant grammars or grammars rarely used in the training set are eliminated. This enhances the time performance for the identification phase by reducing the number of grammars representing a training set. Furthermore, the number of grammars can be restricted to a preset maximum. Fore more details refer to [1].

The identification module consists of an ESLTAG parser and sets of inferred grammars coupled with their threshold functions. Each grammar set represents a certain structure. Depending on the training set fed to the inference engine, these structures could be as general as a pseudoknot structure, more specific as an H-type pseudoknot structure, or as specific as the structure of Antizyme RNA frameshifting stimulation element, for example. Given an input RNA sequence, the user can select to check it against a certain set of grammars. The identification module will answer the question of whether it belongs to the structure defined by this set of grammars.

The Identification/Prediction Phase

In this paper, we present a variant of TAG_{RNA}Inf which can be used to fold RNA sequences. In the new framework, the training phase remains unchanged. The Identification phase is replaced with an identification/prediction phase. In this phase, the

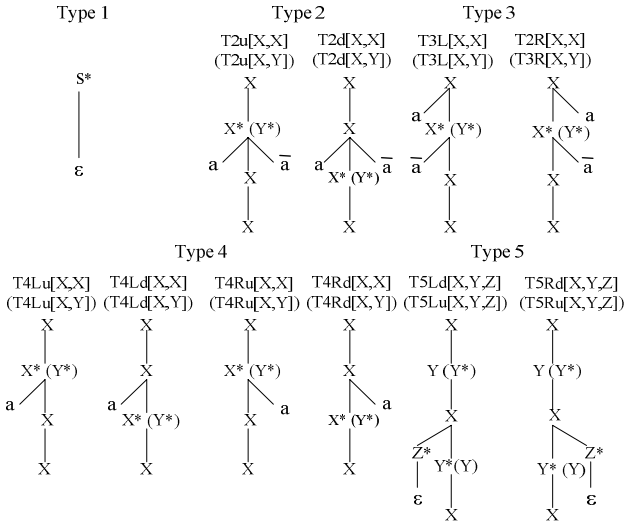

Fig. 2. TAG_{RNA}

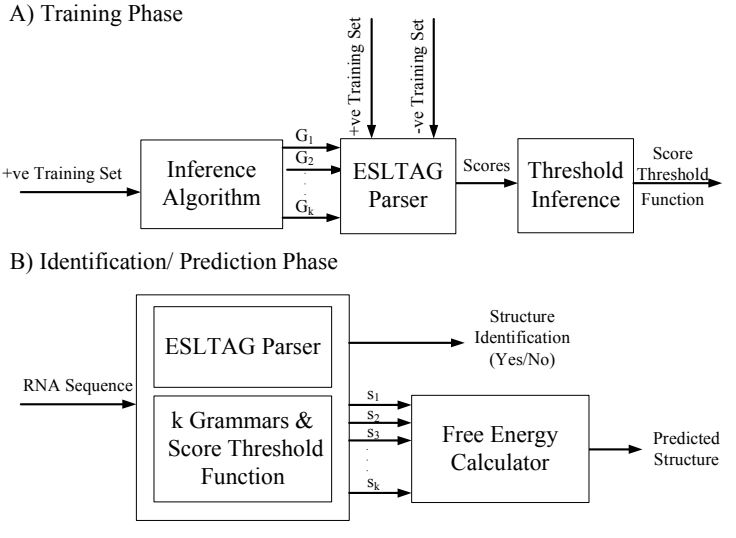

Fig. 3. TAG_{RNA}Inf : RNA structure identification/prediction framework

identification question is answered as before. Additionally, if the input sequence is identified to have the structure represented by the selected set of grammars, the sequence will be folded, and TAG_{RNA}Inf will output its structure. If the sequence was accepted by more than one grammar in the set, the structure resulting in minimum free energy is selected. To calculate the mfe for a certain structure, we use RNAeval tool from the Vienna suite [13]. RNAeval does not support pseudoknots. We

approximate the free energy of a pseudoknot by the sum of the free energies for its two stem-loops as calculated by RNAeval. If a sequence was accepted by the parser, resulting in a non-zero score, but was rejected by the threshold function, the user may choose to fold the sequence despite its rejection. However, the confidence level for this structure is not expected to be high. The framework is illustrated in Fig. 3.

The bottleneck for the computational complexity of our approach lies in the parser. We currently use an implementation of the SLTAG and ESLTAG parses described in [31]. If the set of grammars used do not have any TYPE5 trees (see Fig 2) the SLTAG parser is used; otherwise, the ESLTAG parser is used. The time complexity of the SLTAG and the ESLTAG parsers are $O(n^5)$ and $O(n^4)$ respectively. Both parsers have $O(n^4)$ space complexity.

4 Experimental Results

To test the accuracy of folding for TAG_{RNA}Inf, we evaluate the sensitivity and specificity of the predicted structures for a set of H-type pseudoknot sequences. The folding sensitivity and specificity are defined as follows:

$$Folding\_Sensitivity = \frac{TP}{ref\_bps} \text{ and } Folding\_Specificity = \frac{TP}{predicted\_bps}$$

Where *TP*, *ref_bps*, and *predicted_bps* are the number of correctly predicted bps, number of bps in the actual structure, and total number of predicted bps respectively.

For the training phase of this experiment, we used 105 H-type RNA sequences as the +ve training set and 107 non-pseudoknot sequences as the –ve training set. The +ve training data set was collected from Pseudobase [4], the tmRNA database [34], and Rfam database [11]. We arbitrarily selected sequences from tmRNA and extracted PK1, PK2, and PK4 from them. The negative training set was driven from the non-pseudoknot families in Rfam database, taking into consideration that the lengths of these sequences would be in the same range as the positive population. The +ve training set resulted in 6 grammars. The test set consists of 36 H-type pseudoknot sequences. The test set was driven from the same sources as the +ve training set. It includes 4 sequences from Rfam, 20 sequenes from Pseudobase, and 12 from the tmRNA database.

We ran three comparative experiments on the test data. The first was an identification/structure prediction experiment. In this experiment, the test set was fed to the identification engine to check if the structure of each sequence belongs to any of the inferred grammars. The identification sensitivity and specificity for the training data set are 87.4 and 84.4 respectively. The sensitivity and specificity for identification are defined as:

Table 1. Folding sensitivity and specificity for the 31 sequenced accepted by the H-type pseudoknot grammars on TAG_{RNA}Inf

	Sensitivity	Specificity
ILM	69.1	67.7
pknotsRG (enf)	75.9	77.9
TAG_{RNA}	83.7	79.3
TAG_{RNA}Inf	83.4	87.4

Table 2. . Folding sensitivity and specificity for the whole 36 sequence test set

	Sensitivity	Specificity
ILM	68.5	67.7
pknotsRG (enf)	75.5	77.0
TAG_{RNA}	80.0	75.5
TAG_{RNA}Inf	79.5	85.3

$$Identification\_Sensitivity = \frac{TP}{TP + FN} \text{ and } Identification\_Specificity = \frac{TP}{TN + FP}$$

where *TP*, *TN*, *FP*, and *FN* are the number of true positives, true negatives, false positives, and false negatives respectively.

Out of the 36 input sequences, 31 were accepted and 5 were rejected. The structure generated by TAG_{RNA}Inf for the accepted 31 sequences were compared with the actual structures for these sequences, taken from the source databases, and *folding_sensitivity* and *folding_specificity* were calculated. We also generated structures for the same set of 31 sequences by ILM, pknotsRG, and TAG_{RNA}. pknotsRG was tested in enforce mode (enf), which enforces a pseudoknot in the predicted structure. Also, ***Use extended helix plot score*** option was set for ILM as recommended by the ILM website for single sequence structure prediction. All other options for all tools were set to their defaults. The grammars generated by TAG_{RNA}Inf have default minimum stem length of 2 and maximum bulge loop length of 2. Table1 lists the comparative results for this experiment. TAG_{RNA}Inf results in the best specificity with a big margin and the second best sensitivity after TAG_{RNA} with a very small difference. The high specificity of TAG_{RNA}Inf is expected because the grammars used for prediction are H-type pseudoknot grammars. Sequences whose structures are not expected to follow any of the inferred grammars are excluded in the identification phase, as will be illustrated further in the following two experiments. Figure 4 illustrate an example where TAG_{RNA}Inf gives more accurate structure prediction than ILM and pknotsRG.

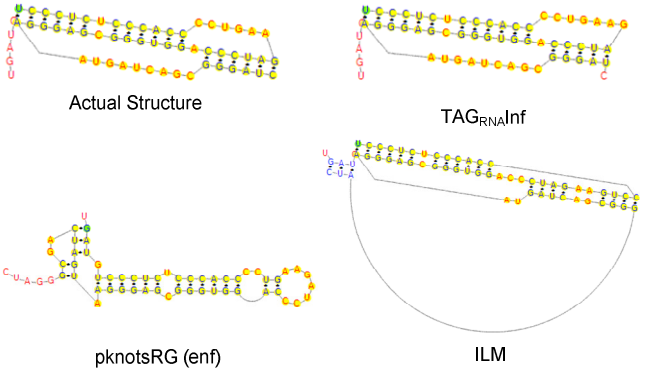

Fig. 4. Actual [11] and predicted structures for an Antizyme RNA frameshifting regulating sequence. Structure images are generated using Pseudoviewer [7]

When we compared the structures predicted by TAG_{RNA} and TAG_{RNA}Inf, we found out that in most cases where TAG_{RNA}Inf gave better predictions than TAG_{RNA}, the structures predicted by TAG_{RNA} had one bp stems in them. As mentioned earlier the default setting for the minimum stem size in TAG_{RNA}Inf grammars is two. TAG_{RNA} has an option to change the minimum stem size, but we tested all tools under their default settings. It is worthy to mention here, that the reported results for TAG_{RNA} take advantage of the identification phase performed by TAG_{RNA}Inf, and these results will endure a drop when we eliminate this phase in the second experiment.

In the second experiment, we included all 36 sequences when calculating folding sensitivity and specificity, ignoring the results of the identification phase. The numbers in Table 2 show a drop in the results, compared to Table 1, across all prediction tools, except for the specificity of ILM. If we look closer to the prediction results for the 5 rejected sequences, listed in table 3, we will observe the following: 2 out of the 5 rejected sequences (BSBV2 and BYDV-NY-RPV) give low sensitivity and specificity values across all prediction tools. Additionally, TRV-PSG and oligo-PK5 give low sensitivity on TAG_{RNA}Inf, This explains the general improvement achieved when these sequences are removed from the test set, specially for TAG_{RNA}Inf. Note that according to the framework we are presenting, identification must be performed prior to structure prediction. Thus, if a sequence was not identified to have a structure that belongs to the family of grammars under consideration, the structure predicted for this sequence by TAG_{RNA}Inf, if any, is considered to have a low confidence level.

In addition to the previous two experiments, we ran a third experiment in which we added two non-pseudoknot RNA grammars to the inferred six grammars. The two added grammars were for hair-pin RNA structures. The aim of this experiment was to test the effect of broadening the search space beyond the structures of the training set and compare it with the other approaches. We realize that the search space of the other tools is much broader. It is worth noting here that our approach is structure prediction through grammar learning. In this experiment, the two grammars were added, without going through the learning phase. Also, a threshold function was not inferred, and no identification was done before the structure prediction.

Table 4 includes the results for predicting the structure of the 36 input sequences with and without the added non-pseudoknot grammars. It also includes the results for pknotsRG in enforce pseudoknot mode and mfe mode. The mfe mode predicts the minimum free energy structure without trying to enforce a pseudoknot structure. We report sensitivity, specificity, and number of sequences whose predicted structure differed due to introducing the non-pseudoknot grammars for TAG_{RNA}Inf, or switching to mfe mode in the case of pknotsRG.

Table 3. Folding sensitivity and specificity for the five sequences rejected by the H-type pseudoknot grammars on TAG_{RNA}Inf

Sequence	ILM		pknotsRG(enf)		TAG_{RNA}		TAG_{RNA}Inf	
	Sen	Spec	Sen	Spec	Sen	Spec	Sen	Spec
TRV-PSG [4][32]	100	92.9	100	100	38.5	35.7	61.5	100
BSBV2 [4][15]	25.0	27.3	41.7	38.5	83	83	41.7	55.6
BYDV-NY-RPV[4]	0.0	0.0	22.0	22.0	0.0	0.0	0.0	0.0
Oligo-PK5 [4]	100	100	100	100	100	100	75.0	100
Azoacus_BH72(PK1) [34]	100	100	100	100	56.0	45.0	89.0	100

Table 4. Folding sensitivity and specificity for the whole 36 sequence test set on pknotsRG in both enforce mode (enf) and mfe mode and on $TAG_{RNA}Inf$ using the inferred H-type pseudoknot grammars (PK) and with the added two hair-pin grammars (nonPK)

	Sensitivity	Specificity	Affected sequences n/36
pknotsRG (enf)	75.5	77.0	-
pknotsRG (mfe)	75.0	76.7	5
$TAG_{RNA}Inf$ (PK)	79.5	85.3	-
$TAG_{RNA}Inf$ (nonPK)	78.0	85.7	4

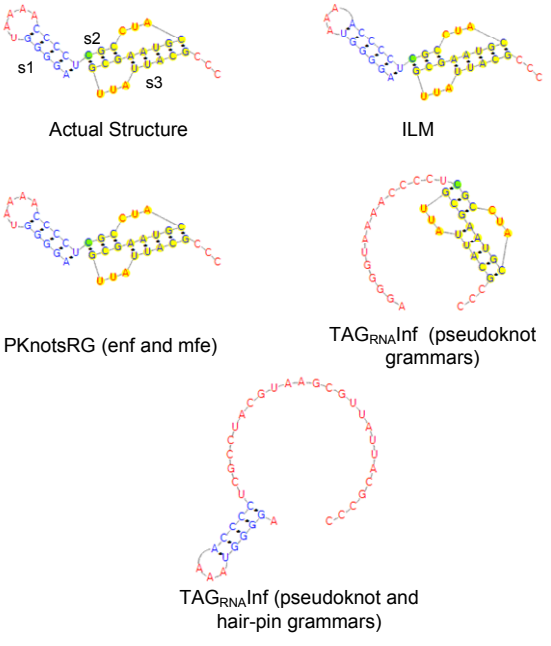

Fig. 5. Actual [32] and predicted structures for TRV-PSG RNA sequence. Structure images are generated using Pseudoviewer [7].

We notice more stability in the sensitivity and specificity of pknotsRG than $TAG_{RNA}Inf$. We found that 5 out of the 36 structures were affected in the case of pknotsRG and 4 in the case of $TAG_{RNA}Inf$. There was no overlap between the sequences affected for each approach. We also found that 3 out of the 4 affected sequences in the case of $TAG_{RNA}Inf$ belong to the set of rejected (unidentified) sequences reported in Table 3. This once more suggests the benefit of using the identification before prediction idea as an indicator of the confidence level of the predicted structure.

To illustrate the advantage of this methodology further, we will present an example, TRV-PSG, which gave considerably worse sensitivity and specificity on $TAG_{RNA}Inf$, compared to some of the other approaches. The actual structure for

TRV-PSG [32], illustrated in Fig. 5, consists of a hair-pin concatenated to a H-type pseudoknot. PknotsRG gave perfect structure prediction of TRV_PSG in both enf and emf modes. ILM gave almost perfect structure prediction where it predicted one additional bp in the hair-pin stem (s1). TAG_{RNA}Inf with the H-type pseudoknot grammar set totally missed the hair-pin structure (s1). On the other hand, with the added two hair-pin grammars, TAG_{RNA}Inf predicted a hair-pin structure for TRV-PSG, and the pseudoknot (s2 and s3) was missed. Notice once more that the identification phase was able to recognize that the structure of TRV-PSG does not belong to the structures represented by the inferred set of grammars. TAG_{RNA}Inf could have been able to predict the correct structure for TRV-PSG if the training data set included a sequence that had a similar structure, a hair-pin concatenated to the pseudoknot.

5 Conclusion and Future Directions

In this paper we presented an RNA structure identification/prediction framework, TAG_{RNA}Inf capable of handling pseudoknot structures. The frame work is a variant of our previous work [1]. In this framework, if a certain structure is identified in a sequence, a folding is computed for the sequence. Our experimental results show the advantage of the identification step to the folding performed by TAG_{RNA}Inf as well as other folding approaches.

In this paper and in [1], we discussed two problems related to RNA structure analysis, which are structure identification and structure prediction. In future work, we plan to address the problem of structural classification. Our preliminary results showed that the use of grammars alone would not be sufficient for classification. We plan to investigate the coupling of the grammatical methods with sequence based methods to do structural classification.

As mentioned earlier, the parser used within TAG_{RNA}Inf's identification/prediction phase is an implementation of the SLTAG/ESLTAG parsers described in [31]. These algorithms use a $(n+1)^4$ matrix and they can be described as metric-centric. Independent of the elementary trees in the given grammar, the algorithms step through each entry in this matrix to check if any trees can be placed in this entry. This means that even if no tree will ever be placed in a matrix entry, some work is done corresponding to this entry. In [21], we describe new parsing algorithms for SLTAG and ESLTAG which are tree-centric. These algorithms are expected to be more time efficient in practice. Additionally, since the matrix used by the parsers is usually sparse, we intend to use other data structures that will result in reducing the space requirements as well. We plan to provide comparative results to prove the vantage of the new algorithms in practice. Additionally, we will work on designing a parallelized version of these algorithms.

References

1. Al Seesi, S., Rajasekaran, S., Ammar, R.: Pseudoknot Identification through Learning TAG_{RNA}. In: Chetty, M., Ngom, A., Ahmad, S. (eds.) PRIB 2008. LNCS (LNBI), vol. 5265, pp. 132–143. Springer, Heidelberg (2008)
2. Akutsu, T.: Dynamic Programming Algorithms for RNA Secondary Structure Prediction with Pseudoknots. Discrete Applied Mathematics 104, 45–62 (2000)

3. Ambros, V., Bartel, B., Bartel, D.P., Burge, C.B., Carrington, J.C., Chen, X., Dreyfuss, G., Eddy, S.R., Griffiths-Jones, S., Marshall, M., Matzke, M., Ruvkun, G., Tuschl, T.: A Uniform System for microRNA Annotation. RNA 9(3), 277–279 (2003)
4. Batenburg, Dvan, F.H., Gultyaev, A.P., Pleij, C.W.A., Ng, J., Oliehoek, J.: Pseudobase: a Database with RNA Pseudoknots. Nucl. Acids Res. 28(1), 201–204 (2000)
5. Brazma, A., Jonassen, I., Vilo, J., Ukkonen, E.: Pattern Discovery in Biosequences. In: Honavar, V.G., Slutzki, G. (eds.) ICGI 1998. LNCS, vol. 1433, pp. 255–270. Springer, Heidelberg (1998)
6. Buratti, E., Dhir, A., Lewandowska, M.A., Baralle, F.E.: RNA Structure is a Key Regulatory Element in Pathological ATM and CFTR Pseudoexon Inclusion Events. Nucl. Acids Res. 35(13), 4369–4383 (2007)
7. Byun, Y., Han, K.: PseudoViewer: Web application and Web service for Visualizing RNA Pseudoknots and Secondary structures. Nucl. Acids Res. 34, W416–W422 (2006)
8. Chiu, D., Kolodziejczak, T.: Inferring consensus structure from nucleic acid sequences. Comput. Appl. Biosci. 7, 347–352 (1991)
9. Dirks, R.M., Pierce, N.A.: A Partition Function Algorithm for Nucleic Acid Secondary Structure Including Pseudoknots. J. Comput. Chem 24(13), 1664–1677 (2003)
10. Gilbert, W.: The RNA World. Nature 319, 618 (1986)
11. Griffiths-Jones, S., Moxon, S., Marshall, M., Khanna, A., Eddy, S.R., Bateman, A.: Rfam: Annotating Non-coding RNAs in Complete Genomes. Nucl. Acids Res. 33, D121–D124 (2005)
12. Gulko, B., Haussler, D.: Using multiple alignments and phylogenetic trees to detect RNA secondary structure. In: Proc. Pac. Symp. Biocomput. vol. 1, pp. 350–367
13. Hofacker, I.L., Fontana, W., Stadler, P.F., Bonhoeffer, S., Tacker, M., Schuster, P.: Fast Folding and Comparison of RNA Secondary Structures. Monatshefte f. Chemie 125, 167–188 (1994)
14. Joshi, A.K., Levy, L., Takahashi, M.: Tree Adjunct Grammars. Journal of Computer and System Sciences 10, 136–163 (1975)
15. Koenig, R., Commandeur, U., Loss, S., Beier, C., Kaufmann, A., Lesemann, D.-E.: Beet Soil-borne Virus RNA 2: Similarities and Dissimilarities to the Coat Protein Gene-carrying RNAs of other Furoviruses. J. Jen. Virol. 78, 469–477 (1997)
16. Laxminarayana, J.A., Nagaraja, G., Balaji, P.V.: Identification of Pseudoknots in RNA Secondary Structures: A Grammatical Inference Approach. In: Mukherjee, D.P., Pal, S. (eds.) Proceedings of 5th International Conference on Advances in Pattern Recognition (2003)
17. Lyngso, R., Pedersen, C.: RNA pseudoknot prediction in energy-based models. J. Comput. Biol. 7, 409–427 (2000)
18. Nussinov, R., Pieczenik, G., Griggs, J., Kleitman, D.: Algorithms for loop matchings. SIAM J. Appl. Math. 35, 68–82 (1978)
19. Paillart, J.C., Skripkin, E., Ehresmann, B., Ehresmann, C., Marquet, R.: In vitro Evidence for a Long Range Pseudoknot in the 5'-Untranslated and Matrix Coding regions of HIV-1 Genomic RNA. J. Biol. Chem. 277, 5995–6004 (2002)
20. Pedersen, J.S., Bejerano, G., Siepel, A., Rosenbloom, K., Lindblad-Toh, K., Lander, E.S., Kent, J., Miller, W., Haussler, D.: Identification and Classification of Conserved RNA Secondary Structures in the Human Genome. Public Library of Science. Computational Biology 2(4), e33 (2006)
21. Rajasekaran, S., Al Seesi, S., Ammar, R.: Improved Algorithms for Parsing ESLTAG: a grammatical model suitable for RNA pseudoknots. In: International Symposium on Bioinformatics Research and Applications, ISBRA (submitted, 2009)

22. Reeder, J., Giegerich, R.: Design, Implementation and Evaluation of a Practical Pseudoknot Folding Algorithm Based on Thermodynamics. BMC Bioinformatics 5, 104 (2004)
23. Rivas, E., Eddy, S.: A dynamic programming algorithm for RNA structure prediction including pseudoknots. J. Mol. Biol. 258, 2053–2068 (1999)
24. Robertson, M.P., Igel, H., Baertsch, R., Haussler, D., Ares, M., Scott Jr., W.G.: The Structure of a Rigorously Conserved RNA Element within the SARS Virus Genome. Public Library of Science: Biology 3(1), e5 (2004)
25. Ruan, J., Stormo, G.D., Zhang, W.: An Iterated loop matching approach to the prediction of RNA secondary structures with pseudoknots. Bioinformatics 20(1), 58–66 (2004)
26. Sakakibara, Y., Brown, M., Hughey, R., Mian, I.S., Sjolander, K., Underwood, R.C., Haussler, D.: Stochastic Context-Free Grammars for tRNA Modeling," Nucl. Acids Res. 22, 5112–5120 (1994)
27. Sakakibara, Y.: Grammatical Inference in Bioinformatics. IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 1051–1062 (2005)
28. Searls, D.: The Linguistics of DNA. Am. Scient 80, 579–591 (1992)
29. Takakura, T., Asakawa, H., Seki, S., Kobayashi, S.: Efficient Tree Grammar Modeling of RNA Secondary Structures from Alignment Data. In: Proceedings of posters of RECOMB 2005, pp. 339–340 (2005)
30. Tanaka, Y., Hori, T., Tagaya, M., Sakamoto, T., Kurihara, Y., Katahira, M., Uesugi, S.: Imino Proton NMR Analysis of HDV Ribozymes: Nested Double Pseudoknot Structure and Mg2+ Ion-Binding Site Close to the Catalytic Core in Solution. Nucl. Acids Res. 30, 766–774 (2002)
31. Uemura, Y., Hasegawa, A., Kobayashi, S., Yokomori, T.: Tree Adjoining Grammars for RNA Structure Prediction. Theoretical Computer Science 210(2), 277–303 (1999)
32. van Belkum, A., Cornelissen, B., Linthorst, H., Bol, J., Pley, C., Bosch, L.: tRNA-like Properties of Tobacco Rattle Virus RNA. Nucl. Acids Res. 15(7), 2837–2850 (1987)
33. Vijay-Shanker, K., Joshi, A.K.: Some Computational Properties of Tree Adjoining Grammars. In: 23 rd Meeting of the Association for Computational Linguistics, pp. 82–93 (1985)
34. Williams, K.P.: The tmRNA Website: Invasion by an Intron. Nucl. Acids Res. 30(1), 179–182 (2002)
35. Zuker, M., Stiegler, P.: Optimal computer folding of large RNAsequences using thermodynamics and auxiliary information. Nucleic Acids Res. 9, 133–148 (1981)