Bioinformatics, 31(16), 2015, 2660–2667
doi: 10.1093/bioinformatics/btv210
Advance Access Publication Date: 16 April 2015
Original Paper

OXFORD

Structural bioinformatics

Accurate prediction of RNA nucleotide interactions with backbone *k*-tree model

Liang Ding[1,*], Xingran Xue[1], Sal LaMarca[1], Mohammad Mohebbi[1], Abdul Samad[4], Russell L. Malmberg[2,3] and Liming Cai[1,2,*]

[1]Department of Computer Science, [2]Institute of Bioinformatics and [3]Department of Plant Biology, University of Georgia, GA 30602, USA and [4]Department of Computer Science, BUITEMS, Pakistan

*To whom correspondence should be addressed.
Associate Editor: Anna Tramontano

Received on November 3, 2014; revised on April 4, 2015; accepted on April 12, 2015

Abstract

Motivation: Given the importance of non-coding RNAs to cellular regulatory functions, it would be highly desirable to have accurate computational prediction of RNA 3D structure, a task which remains challenging. Even for a short RNA sequence, the space of tertiary conformations is immense; existing methods to identify native-like conformations mostly resort to random sampling of conformations to achieve computational feasibility. However, native conformations may not be examined and prediction accuracy may be compromised due to sampling. State-of-the-art methods have yet to deliver satisfactory predictions for RNAs of length beyond 50 nucleotides.

Results: This paper presents a method to tackle a key step in the RNA 3D structure prediction problem, the prediction of the nucleotide interactions that constitute the desired 3D structure. The research is based on a novel graph model, called a *backbone k-tree*, to tightly constrain the nucleotide interaction relationships considered for RNA 3D structures. It is shown that the new model makes it possible to efficiently predict the optimal set of nucleotide interactions (including the non-canonical interactions in all recently revealed families) from the query sequence along with known or predicted canonical basepairs. The preliminary results indicate that in most cases the new method can predict with a high accuracy the nucleotide interactions that constitute the 3D structure of the query sequence. It thus provides a useful tool for the accurate prediction of RNA 3D structure.

Availability and Implementation: The source package for BkTree is available at http://rna-informatics.uga.edu/index.php?f=software&p=BkTree.

Contact: lding@uga.edu or cai@cs.uga.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

In the past decade, there have been many revelations of the importance of non-coding RNAs to cellular regulatory functions and thus a growing interest in the computational prediction of RNA 3D structure (Laing and Schlick, 2010; Leontis and Westhof, 2012). Nevertheless, RNA 3D structure prediction from a single RNA sequence is a significant challenge. One major unresolved issue is the immense space of tertiary conformations even for a short RNA sequence. Existing methods usually employ random sampling algorithms for computation feasibility, which assemble sampled tertiary motifs into native-like structures (Das and Baker, 2007; Ding *et al.*, 2008; Jonikas *et al.*, 2009; Parisien and Major, 2008; Popenda *et al.*, 2012; Sharma *et al.*, 2008). To reduce the chance to miss native structures, the assembly algorithms have mostly been guided with constraining structural models. For example, MC-Fold/MC-Sym (Parisien and Major, 2008) assumes the 3D structure consists of 4-nt

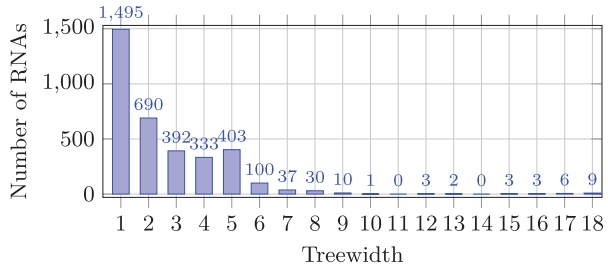

Fig. 1. Treewidth distribution of NIR graphs of more than 3500 RNA chains from the RNA Structure Atlas. The RNAs with treewidth larger than 18 are omitted due to their very small number. These treewidths are actually upper bounds it is likely that the exact treewidths of the NIR graphs may be smaller

cyclic tertiary motifs constructible from the predicted secondary structure. Rosetta (Das and Baker, 2007; Das *et al.*, 2010) *de novo* assembles 3D structure from a database of 3-nt tertiary fragments. Other methods follow samplings that preserve the secondary structure (Bida and Maher, 2012; Popenda *et al.*, 2012; Reinharz *et al.*, 2013). However, these constraining models do not necessarily ensure that native conformations are examined. The state-of-the-art methods have yet to deliver the desired prediction accuracy for RNA sequences of lengths beyond 50 nucleotides (Laing and Schlick, 2010).

In this work, we introduce a novel method to predict nucleotide interactions from known or predicted canonical basepairs as a key step toward accurate prediction of 3D structure. Accurate knowledge of the nucleotide interactions is crucial to predicting the 3D structure of an RNA and subsequently predicting its functional roles. To predict nucleotide interactions, our method is guided by a novel graph model called a *backbone k-tree*, for small integer k, to globally constrain the nucleotide interaction relationships (NIRs) that constitute the 3D structure. In such a k-tree graph, nucleotides are organized into groups of size $k+1$, such that NIRs are permitted only for nucleotides belonging to the same group and groups are connected to each other with a tree topology (see section 2). This model was inspired by our recent discovery of the small treewidth of the NIR graphs for more than 3500 RNA chains extracted from 1984 RNAs whose structures have been resolved (Fig. 1), where the treewidth of each NIR graph is computed by an approximation algorithm (Bodlaender and Koster, 2010). Treewidth is a graph metric, which indicates how much a graph is tree-like (Bodlaender and Koster, 2010; Van Leeuwen, 1990). We have been able to develop dynamic programming (DP) algorithms with $O(n^{k+1})$ time and space complexities, efficient for small k, to compute the optimal backbone k-tree spanning over the nucleotides on the query sequence, given a scoring function (Ding *et al.*, 2014a,b).

To ensure that the computed optimal k-tree can actually yield the set of nucleotide interactions that constitutes the native 3D structure, our method proposes to identify detailed patterns of nucleotide interactions for every group of $k+1$ nucleotides found in known RNA 3D structures and to score every such pattern. We consider nucleotide interactions from the established geometric nomenclatures and families (Leontis and Westhof, 2001; Leontis *et al.*, 2002; Stombaugh *et al.*, 2009), including base–base, base–phosphate, base–ribose (Zirbel *et al.*, 2009; Zirbel, 2011), base–stacking interactions as well as the phosphodiester bonds between two neighbouring nucleotides on the backbone. To test our method, we adopted an improved 3-tree model, and pre-computed candidates of interaction patterns for every group of 4 given nucleotides (see Supplementary Fig. S1 in the Supplementary Material). These annotated atom-level interaction patterns have been extracted from resolved 3D structures of RNAs in the RNA Structure Atlas (Sarver *et al.*, 2008). To avoid overfitting, only nucleotide interactions from RNAs of length ≤ 100 nucleotides were selected. To score such patterns, we trained artificial neural networks (ANNs) to compute the confidence of every admissible nucleotide interaction pattern for every group of 4 given nucleotides. We filtered out unlikely interaction patterns and kept only those with high confidences. With this 3-tree model, our algorithm efficiently predicts an optimal set of nucleotide interactions from the query sequence (along with canonical base pairs) within computational time $O(n^3)$. We have implemented the algorithm into a program called BkTree as a part of a 3D structure prediction framework (Supplementary Fig. S2 in the Supplementary Material).

We evaluated our methods through testing BkTree on two sets of data. First, the performance of nucleotide interaction prediction was measured on a set of 43 RNAs of lengths ranging from 26 to 128 nucleotides, a benchmark used by the survey of state-of-the-art 3D structure prediction methods (Laing and Schlick, 2010). The resolved, atom-level interactions of these high resolution RNAs were extracted with FR3D (Sarver *et al.*, 2008). Sensitivities (STY), positive predictive values (PPV) and Matthews correlation coefficients (MCC) (Laing and Schlick, 2010) of the nucleotide interactions predicted by BkTree were calculated for these 43 RNAs. The overall performance was also compared with previous programs MC (Parisien and Major, 2008), Rosetta (Das and Baker, 2007), and NAST (Jonikas *et al.*, 2009). Second, performance of nucleotide interaction prediction was also evaluated by testing BkTree on a set of 13 single RNA chains from PDB (Berman *et al.*, 2000) of lengths between 100 and 200 nucleotides, whose nucleotide interaction patterns have not been extracted for the construction of ANNs. To evaluate the significance of our method to 3D structure prediction, we modeled 3D conformations from predicted nucleotide interactions for all 43 RNAs. Both root-mean-square deviation (RMSD) values and *deviation index* (DI; Parisien *et al.*, 2009) values were calculated from the modeled 3D conformations. RMSD comparisons on about a dozen representative RNAs were also made with previous methods MC, Rosetta, NAST, and RNA–MoIP (Reinharz *et al.*, 2013). These evaluations show that BkTree predicted nucleotide interactions with high accuracies across the tested RNAs, including RNAs whose interaction patterns were not used for training. Our method also impressively outperformed the other methods on the overwhelming majority of the tested RNAs and showed a great potential for handling RNAs beyond short lengths.

2 Model and methods

In this work, we consider RNA nucleotide interactions of atomic-resolution of all known types, Supplementary Table S1 in the Supplementary Material summarizes these nucleotide interactions by their geometric families.

We use notation $\langle X, Y, t\rangle$ for a type t interaction between nucleotides X and nucleotide Y (from 5' to 3'), where $X, Y \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}$. Let $I_{XY} = \{\langle X, Y, t\rangle : t \text{ is an interaction type}\}$ and $I = \bigcup_{X,Y\in\{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{U}\}} I_{XY}$.

2.1 Backbone *k*-tree model

Let $S = S_1 S_2 \ldots S_n$ be an RNA sequence, in which $S_i \in \{A, C, G, U\}$, for $1 \leq i \leq n$. The *nucleotide interaction relation* (NIR) *model* for the native structure of S is a pair $\langle G; A\rangle$, where $G = (V, E)$ is called the *NIR graph* of S with vertex set $V = \{1, 2, \ldots, n\}$, and A is an *association* such that for every pair

$i<j$, $A(i,j) \subseteq I_{S_i,S_j}$ is the set of interactions between nucleotides S_i and S_j in the native structure and $A(i,j)=\emptyset$ implies $(i,j) \in E$. Note that because of phosphodiester bonds between neighboring nucleotides, the NIR graph G always contains the Hamiltonian path $(i,i+1)$, $i=1,2,\ldots,n-1$; these edges are named *backbone edges*.

In our recent investigation (Ding *et al.*, 2014a,c), we constructed NIR graphs for all RNAs whose 3D structures were known from RNA Structure Atlas (Reinharz *et al.*, 2013). We discovered that an overwhelming majority of these RNAs are of small treewidths (Fig. 1). Theoretically, if a graph has treewidth bounded by k, any clique obtained by deleting vertices and edges and contracting edges of the graph can contain at most $k+1$ vertices (Arnborg *et al.*, 1990). Thus the distribution of treewidths suggests that NIRs in the RNA 3D structures are in general not arbitrarily complex.

The concept of treewidth is closely related to, and may be better explained with the notion of *k-tree*, which is central to this work.

Definition 1: (Patil, 1986) Let integer $k \geq 1$. The class of *k-trees* are graphs defined by the following inductive steps:

1. A k-tree of $k+1$ vertices is a clique of $k+1$ vertices;
2. A k-tree of n vertices, for $n > k+1$, is a graph consisting of a k-tree G of $n-1$ vertices and a vertex v, which does not occur in G, such that v forms a $(k+1)$-clique with some k-clique already in G.

Supplementary Figure S3a and b in the Supplementary Material show a 3-tree of 7 vertices. It is well known that for any $k \geq 1$, a graph is of treewidth $\leq k$ if and only if it is a subgraph of some k-tree (Arnborg and Proskurowski, 1989). This also suggests that every graph of treewidth $\leq k$ can be augmented with additional edges into a k-tree. Thus, given a NIR model $\langle G; A\rangle$, where NIR graph G is of a treewidth $\leq k$, one can augment G to a k-tree with additional edges, and for each newly added edge (i,j), let $A(i,j)=\emptyset$. Since such k-trees contain all backbone edges, they are called *backbone k-trees*.

Definition 2: A *backbone k-tree model* is a NIR model $\langle G; A\rangle$ in which NIR graph G is a backbone k-tree.

This allows us to conclude that for small values of k, backbone k-tree models exist for an overwhelmingly majority of RNAs whose native structures are known. Supplementary Figure S3c in the Supplementary Material shows a backbone 3-tree as the NIR graph for a short sequence consisting of 7 nucleotides.

To predict the set of nucleotide interactions from a query sequence $S = S_1 S_2, \ldots S_n$, we propose to identify a backbone k-tree model $\langle G; A\rangle$, where $G=(V,E)$ and $A(i,j) \subseteq I_{S_i,S_j}$ such that $A(i,j)=\emptyset \Rightarrow (i,j) \in E$. To ensure the identified model actually corresponds to the set of nucleotide interactions that constitute the native structure of the query sequence, we will quantify nucleotide interactions for the optimization computation of such a backbone k-tree model.

2.2 Quantification of nucleotide interactions

Definition 3: Let q be a $(k+1)$-clique in a backbone k-tree of query sequence S. An *interaction pattern* (ip) for clique q is a set $A(q)$ of nucleotide interactions, for some association A, such that $A(q) = \cup_{i,j \in q} A(i,j)$.

Given an ip $A(q)$ for clique q, the *subgraph of q induced by A(q)*, denoted with $H_{q,A(q)} = (q, E_{q,A(q)})$, is such that $(i,j) \in E_{q,A(q)}$ if and only if $A(i,j) \neq \emptyset$. Supplementary Figure S1 in the Supplementary Material illustrates the examples of a $(k+1)$-clique q, two ips of q and their induced subgraphs.

Definition 4: Let q be a $(k+1)$-clique in a backbone k-tree of query sequence S. The *confidence* of a given ip $A(q)$ for clique q is defined as

$$f(q, A(q), S) = \sum_{(i,j) \in E_{q,A(q)}, \langle S_i, S_j, t\rangle \in A(i,j)} c^{(i,j),t}_{q,H_{q,A(q)}} \quad (1)$$

where $c^{(i,j),t}_{q,H_{q,A(q)}}$ is the *confidence* of interaction $\langle S_i, S_j, t\rangle$ given q and the subgraph $H_{q,A(q)}$ induced by ip $A(q)$.

In Section 3, we will introduce ANNs that compute confidence $c^{(i,j),t}_{q,H_{q,A(q)}}$.

For every clique q, with $\mathcal{P}(q)$, we denote the finite set of all ips for q. In the practical application, we may only include those ips in $\mathcal{P}(q)$ which have 'high' confidences (e.g. above certain threshold).

Definition 5: Let k be any fixed integer ≥ 2. The *nucleotide interaction prediction* problem NIP(k) is, given an input query sequence S, to identify a backbone k-tree model $\langle G^*; A^*\rangle$, such that

$$(G^*; A^*) = \arg \max_{\langle G; A\rangle} \{ \sum_{q \text{ in } G, A(q) \in \mathcal{P}(q)} f(q, A(q), S)\} \quad (2)$$

2.3 Overview of the method

To solve the NIP(k) problem, our method consists of three major components.

1. Data repositories include NIPDB and NIPCTable. NIPDB is a database of all possible interaction patterns. To build the database, we first extracted a set $\mathcal{P}(q)$ of nucleotide interaction patterns for every $(k+1)$-clique q, which were found in the known 3D structures of RNAs with length ≤ 100 nucleotides. Then an unique identifier was assigned to each such clique by taking into account both the nucleotides and their backbone distances. See Supplementary Figure S1 in Supplementary Material for examples.

   NIPCTable is a matrix for compatibility between every pair of ips for two cliques that share all but one vertex (nucleotide). To compute for the optimization problem formulated with (2), for every two $(k+1)$-cliques q_1 and q_2 that are adjacent in the k-tree G, $A(q_1)$ and $A(q_2)$ are required to be compatible in the sense that the two interaction sets among the k common nucleotides of q_1 and q_2 are identical. The compatibility of all pairs of ips in NIPDB forms a binary matrix. For the efficiency, the compatibility can be precomputed before the prediction program is executed. The nucleotides in an ip are ordered from 5′ to 3′. Given two ips I_1 and I_2 each with $k+1$ nucleotides, we enumerate all $(k+1)^2$ ways of mapping k nucleotides of I_1 to k nucleotides of I_2. For each of the mappings, if the selected two sets of k nucleotides are not identical, two ips are not compatible; otherwise, we further verify the interactions among the two sets of k nucleotides and the compatibility holds when they are identical.
2. A set of ANNs; each computes the confidence for a specific interaction between two given nucleotides on the query sequence.
3. A DP algorithm that computes the solution to Equation (2).

Given the query sequence S and the known or predicted canonical basepairs on S, our method first employs ANNs to compute $c^{(i,j),t}_{q,H_{q,A(q)}}$, the confidence of the interaction $\langle S_i, S_j, t\rangle$ in the interaction pattern

$A(q)$ for $(k+1)$-clique q that involves vertices i and j, where $H_{q,A(q)}$ is the subgraph induced by ip $A(q)$. This is done for every pair of $i<j$, every interaction type t, every $(k+1)$-clique q and every ip $A(q) \in \mathcal{P}(q)$ for q. Then it computes the confidence score $f(q, A(q), S)$ for every ip $A(q) \in \mathcal{P}(q)$ of every $(k+1)$-clique q, using formula (1). Afterward, it runs the DP algorithm to solve Equation (2).

3 Algorithms

3.1 ANNs for computing interaction confidence

We constructed ANNs that compute confidences of nucleotide interactions, one ANN for every specific nucleotide interaction $\langle S_i, S_j, t\rangle$ contained in a given specific interaction pattern $A(q)$ of a given $(k+1)$-clique q. We use $\mathcal{N}^{(i,j),t}_{q,H_{q,A(q)}}$ to denote such an ANN and $c^{(i,j),t}_{q,H_{q,A(q)}}$ for the confidence score that the ANN computes. Each ANN $\mathcal{N}^{(i,j),t}_{q,H_{q,A(q)}}$ consists of an input layer, two hidden layers (with 8 and 16 nodes, respectively), and an output layer (Supplementary Fig. S4 in the Supplementary Material). The output layer is a single unit producing a confidence value for interaction $\langle S_i, S_j, t\rangle$. The input layer consists of input units representing the selected global and local features shown in Supplementary Table S2 in Supplementary Material. The features included the sequence length and the distance between the involved nucleotides as well as neighboring nucleotide types. In addition, we included the information of *assumed* canonical base pairs within the query sequence.

We adopted conventional methods to construct and train each ANN (Mitchell, 1997), typically the technique of back-propagation with gradient descent, using a fixed-size network. The learning rate 0.03 were the values that yielded the best results for some representative ANNs. The training data for the ANNs were from RNA Structure Atlas. We removed all RNAs of lengths larger than 100 nucleotides and RNAs with missing nucleotides. This resulted in a subset of 895 RNAs of single chains. Then all the $(k+1)$-cliques q along with their features of every RNA in the subset were enumerated to form a whole set T. Due to different number of features, the number of $(k+1)$-cliques associated with different q, interaction pattern $A(q)$, and subgraph $H_{q,A(q)}$ in T vary considerably. As a result, for most of the ANNs, only a small portion of 895 RNAs were used for training and testing. A 10-fold cross validation was used to avoid over-fitting.

3.2 Algorithm for NIP(k) problem

Our algorithm solves the NIP(k) problem by producing a pair $\langle G^*; A^*\rangle$ satisfying Equation (2) for the query sequence. In particular, the backbone k-tree G^* constrains the NIR topology, together with the association A^* of nucleotide patterns with all $(k+1)$-cliques in G^*, to achieve the maximum confidence score. The algorithm maximizes the confidence score of a backbone k-tree spanning over the query sequence nucleotides by a DP process. To derive recurrences for the DP, we followed the basic process of creating k-trees given in Definition 1. The inclusion of backbone edges in the k-trees disallows introducing edges in arbitrary order and thus makes the search space much smaller. We briefly explain this algorithm in the following paragraphs.

By *interval* $[i..j]$, for $i \le j$, we mean the set of consecutive integers between i and j, inclusive. Two intervals $[i..j]$ and $[h..l]$ are *non-overlapping* if either $j \le h$ or $l \le i$. Let the query sequence be S of length n and q be a $(k+1)$-clique formed by $k+1$ vertices drawn from $\{1, 2, \ldots, n\}$. Let C be a set of non-overlapping intervals and $A(q) \in \mathcal{P}(q)$ be an ip for clique q. We define function $M(q, C, A(q), S)$ to be the maximum confidence of a k-tree constructed beginning from clique q, which includes backbone edge $(i, i+1)$ for every pair of integers i and $i+1$ both contained in some interval in C. Then we obtain the following recurrence:

$$\begin{aligned} &M(q, C, A(q), S) \\ &= \max_{x \in q,\, y \notin q,\, y \in [i..j] \in C,\, p = q|^x_y} \\ &\quad \{ \max_{A(p) \in \mathcal{P}(p),\, \mathcal{R}(C_1, C_2),\, \mathcal{Q}(A(p), A(q))} \{ M(p, C_1, A(p), S) \\ &\quad + M(q, C_2, A(q), S) + f(q, A(q), S) \} \} \end{aligned} \tag{3}$$

where abbreviations $q|^x_y = q \cup \{y\} \setminus \{x\}$, $\mathcal{Q}(A(p), A(q))$ asserts that the chosen ip $A(p)$ be compatible with the ip $A(q)$, and $\mathcal{R}(C_1, C_2)$ represents the choices of two sets of intervals, C_1 and C_2, which satisfy the following constraints

1. $\{[i..y], [y..j]\} \subseteq C_1$, $\{[w..x], [x..z]\} \subseteq C_2$, for applicable w and z; and
2. $C_1 \cup C_2 = C \cup \{[i..y], [y..j]\} \setminus \{[i..j]\}$, and $C_1 \cap C_2 = \emptyset$.

Recurrence (3) offers a bottom-up process to compute $M(q, C, A(q), S)$. Intuitively, the idea is to create a new clique p from q by introducing a new nucleotide vertex y. There may be one or more sub-k-trees, some stemming from p while others from q (but not including vertex y). Since these sub-k-trees will never join together again, interval sets are used to ensure backbone edges will be properly created. In particular, the set of backbone edges in the k-tree corresponding to the value of function $M(q, C, A(q), S)$ contains only those edges between consecutive indexes specified in the intervals in C. Initially, C may include intervals allowing all backbone edges.

The confidence score of the produced k-tree is computed as the sum of confidence scores of ips chosen for all involved $(k+1)$-cliques. The chosen ips need to be compatible across the cliques when they share nucleotide interactions or even just nucleotides. This is ensured by the assertion $\mathcal{Q}(A(q), A(p))$ by looking up table NIPCTable. In addition, any pattern of interactions between a single nucleotide and multiple others has to exist in the structure database.

To complete the recurrence (3), we need the following base case:

$$M(q, C, A(q), S) = 0 \qquad \text{if } C = \emptyset$$

which will be first computed in a bottom-up DP strategy.

3.3 Improved algorithms

Implementation of the above outlined DP algorithm would require $O(n^{k+1})$ memory space and $O(n^{k+2})$ computation time for every fixed value of k. Following the same idea but creating $(k+1)$-cliques from k-cliques instead has lead to an improved algorithm, with a few more sophisticated steps to navigate through k-cliques. The improved algorithm uses $O(n^k)$ memory and $O(n^{k+1})$ time for every fixed value of k (Ding *et al.*, 2014b).

For $k=3$, the time efficiency can be further improved to $O(n^3)$ with a constrained backbone k-tree model that requires every $(k+1)$-clique to contain at least one backbone edge $(i, i+1)$ for some i. Testing has shown that the constrained backbone k-tree model did not weaken the capability to account for sophisticated nucleotide interactions as the 'standard' backbone k-tree model. The constrained model may reduce biologically unfavorable interaction patterns, e.g. those not involving locally related nucleotides.

3.4 Implementation

We have implemented the new method into a prototype system. The NIPDB database construction was coded in Python, where the

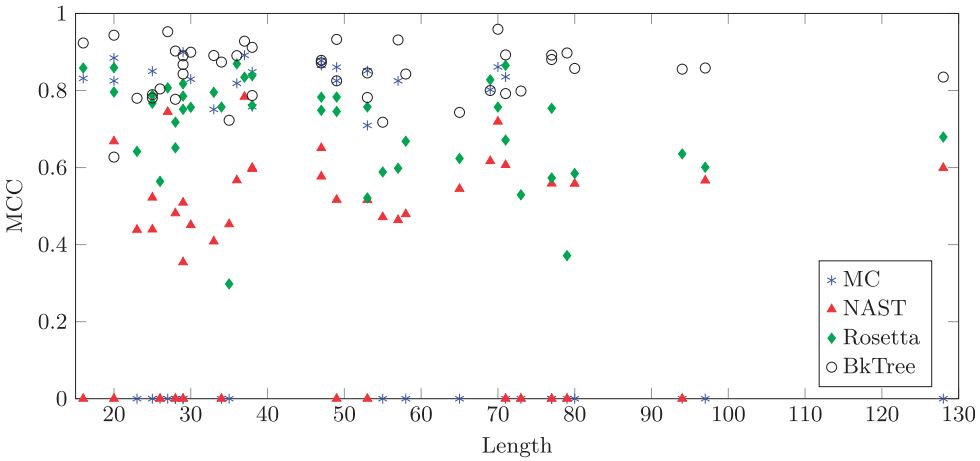

Fig. 2. Comparison of the MCC generated by MC, NAST, Rosetta and BkTree. The MCC of the 43 RNAs are calculated by including canonical base pairs in the results and sorted by their lengths. The plot was derived by merging the results obtained by BkTree and the data computed in the survey (Laing, 2014; Laing and Schlick, 2010). In that survey, the 3D structure predictions with the other 3 methods were based on resolved secondary structures and the secondary structures were included in the calculations. Therefore, the canonical base pairs were also been added to the prediction results by BkTree

Prody package (Bakan *et al.*, 2011) was adopted to search the RNA Structure Atlas. NIPCTable, the matrix for ip consistence was developed using Python. Building and training of all ANNs were realized with WEKA package (Hall *et al.*, 2009) in nearly a month. Programs were coded in Java to compute confidences of ips admissible for every $(k+1)$-clique in the query sequence.

We implemented in C++ the DP algorithm into a program called BkTree based on the constrained backbone 3-tree model. We ran the evaluation tests on a Red Hat 4.8.2-7 server with 4 Intel Quad core X5550 Xeon Processors, 2.66 GHz 8 M Cache and 70 GB Memory. The server runs nearly an hour for predicting a sequence of 100 nucletodies.

4 Performance evaluation

4.1 Test data

We implemented our method in the program BkTree. We evaluated our method through testing BkTree on two sets of RNAs of high resolution structures. One was a list of 43 RNAs that had been used as a benchmark set in the survey of state-of-the-art 3D structure prediction methods (Laing and Schlick, 2010). Eighteen of the RNA sequences are of length $\geq$50 nucleotides. In developing the ANNs for computing interaction confidences, 7 of these RNAs were not included in set *T*. The second set was 13 high resolution (3.5 Åor better) single chain RNAs of lengths from 101 to 174 nucleotides, none of which was included in *T*.

Given the recent progress made in RNA secondary structure prediction (Laing and Schlick, 2010; Reinharz *et al.*, 2013), we believe that canonical base pairs may be routinely predicted with a fair accuracy. Therefore, we have allowed the program BkTree to accept known or predicted canonical base pairs along with the query sequence as input. Note that the knowledge of canonical base pairs does not necessarily imply the whole secondary structure, which is often a part of input to most of the existing RNA 3D prediction methods. In our test, we extracted canonical base pairs of a RNA from FR3D analyzed interactions (Sarver *et al.*, 2008).

4.2 Overall performance

We evaluated the quality of the predicted nucleotide interactions by the sensitivity (STY) and positive predictive value (PPV) against the FR3D-analyzed interactions (Sarver *et al.*, 2008). In order to take into account the effects of both true positive and false positive rates in one measure, the *Matthews correlation coefficient* (MCC), defined in (Laing and Schlick, 2010) as $\text{MCC} := \sqrt{\text{PPV} \times \text{STY}}$, was also calculated.

Supplementary Table S4 in Supplementary Material summarizes the overall performance of BkTree on the benchmark set. On a large majority of RNAs, the sensitivity is decently high. Note that the STY and PPV calculations excluded the canonical base pairs. The sensitivity result indicates that our method has a high accuracy in identifying non-canonical interactions that may be crucial to 3D structures. This is true even for those longer RNAs. We further note that for the 7 RNAs that are not in *T*, BkTree also performed very well.

We point out that in Supplementary Table S4 in Supplementary Material almost all of the relatively low MCC values (below 0.8) were caused by relatively low sensitivity (STY) values. These low sensitivity values were due to that the backbone 3-tree is too weak to model RNAs of structures more complex than helices and junctions, such as pseudoknots. This is evident by the column EdgeDiff, which is the ratio of total number of edges in the NIR graph of the RNA to the number of edges that the constrained 3-tree model is able to include. *k*-tree models, with higher *k* values, can include all edges of the NIR graph and thus is expected improve the performance of prediction (see Section 5 for more discussions on how such *k*-tree models can be efficiently implemented).

4.3 Performance comparison with other methods

We compared our program BkTree with the programs MC, Rosetta and NAST on the capability to predict nucleotide interactions. These other state-of-the-art methods had been surveyed and evaluated in (Laing and Schlick, 2010) based on their ability to identify both base pairing and base stacking interactions only. We removed base-phosphate and base-ribose interactions from our prediction results. We incorporated the canonical base pairs into our results because these other methods include all interactions from the input secondary structure.

Figure 2 shows the MCC plots for MC, Rosetta, NAST, and BkTree on the benchmark set of RNAs. Data of RNAs failed by a program were not included in the calculation. We note that for every RNA, these other programs produced more than one conformation so the results were averaged for these comparisons. The figure demonstrates that BkTree overall outperformed the other three programs in predicting non-canonical base pairing and base stacking interactions.

In addition, Supplementary Table S3 in Supplementary Material gives comparisons on average performance across the 43 RNAs between the four methods. In general, Bktree produced much better average results than Rosetta and NAST, and comparable average results with MC, for which BkTree shows better average STY value than MC, whereas MC gives better average PPV. On MCC values, BkTree had an average over MC. On RNAs of length $\geq$50 nucleotides, BkTree maintained almost the same average MCC as it did on the whole set.

4.4 Performance on long RNAs

We also evaluated performance of BkTree on 13 longer RNAs with diverse structures, including riboswitches, pseudoknots, synthetic RNAs and RNAs containing multi-way junctions. These RNAs have lengths from 101 to 174 nucleotides and thus they were not included in the training data for ANNs. Supplementary Table S5 in Supplementary Material shows that the performance on these long RNAs is comparable to that of the 43 benchmark RNAs. In particular, the average MCC of the predictions is 0.787, suggesting the capability of our method to predict nucleotide interactions for RNAs of lengths beyond 100 nucleotides.

4.5 Significance to 3D conformation prediction

To evaluate the significance of our method to prediction of 3D structure, we have also developed a 3D conformation modeling program that can be pipelined with BkTree for 3D structure prediction (Xue *et al.*, in preparation; also see Supplementary Fig. S2 of Supplementary Material). Here we briefly summarize the underlying method of the 3D modeling. The method takes as input the predicted nucleotide interactions along with the predicted backbone k-tree that organizes the nucleotides into $(k+1)$-cliques. It assigns one geometric motif candidate to the set of predicted nucleotide interactions within every clique. The candidates were first extracted from RNA Structure Atlas (Sarver *et al.*, 2008) with the set of interactions considered. Then the geometry of each candidate was collected from PDB (Berman *et al.*, 2000). Since there are usually more than one motif candidate for every clique, motifs are selected to achieve the highest consistency across all cliques in the k-tree. The consistency between two motifs selected for two respective 'neighboring' cliques is measured with the root-mean-square-deviation (RMSD) on the two motif geometries involving the k common nucleotides shared by the two cliques. The predicted 3D structure consists of a collection of motifs selected for all the cliques, which has the lowest sum of the RMSDs across all pairs of 'neighboring' cliques. The input k-tree enables a DP algorithm to compute the RMSD sum very efficiently, in particular, in time $O(C^2N)$, where C is the number motif candidates considered for each clique, and N is the number of cliques in the k-tree, linearly proportional to the length n of the query RNA sequence. In addition, the method assumes an option to use Amber (Saloman-Ferrer *et al.*, 2012) for structure refinement.

For each of the all 43 RNAs, the 3D modeling program produced an optimally predicted 3D structure whose RMSD was calculated against the resolved structure. The *deformation index* (DI) (Parisien *et al.*, 2009), a measure that accounts for both RMSD and MCC, the quotient between them, was also calculated (see Supplementary Table S7 in Supplementary Material). Supplementary Figure S6 in the Supplementary Material plots the RMSD values for all the 43 RNAs. These data suggest a great potential of our method for RNAs beyond short lengths. In particular, 12 out of the 18 RNAs with lengths exceeding 50 nucleotides achieved RMSD values below 10 Å; Supplementary Figure S5 in Supplementary Material shows some of such examples. On the other hand, the sharp increase of RMSD values for some of longer RNAs in Supplementary Figure S6 in the Supplementary Material by no means indicates the failure of the backbone k-tree model or the proposed 3D modeling method on these RNAs. Most of them appear to require a backbone 4-tree model, beyond the capability of the 3-tree based BkTree program (see Section 5).

To compare with other 3D structure prediction methods, Supplementary Table S6 in Supplementary Material presents the performance values on the 4 representative RNAs chosen in (Laing and Schlick, 2010) which typically contain two hairpins and two junctions. Since both MC and Rosetta allow prediction of multiple optimal or suboptimal folds, we chose to use the best and the averaged values of their solutions. For every one of the 4 RNAs, the RMSD achieved by BkTree is significantly smaller than both the best and the averaged values achieved by MC and Rosetta.

The more recently developed 3D structure prediction program RNA-MoIP (Reinharz *et al.*, 2013) shows some remarkable improvements of the geometries with an integer programming method to fit tertiary motifs into the predicted secondary structure. In Supplementary Table S7 of Supplementary Material, BkTree is compared with RNA-MoIP on all the 9 RNAs tested by RNA-MoIP, all of lengths exceeding 50 nucleotides. This is not an ideal comparison as we have input to BkTree program the known canonical Watson-Crick base pairs instead of the predicted secondary structure. It shows that for 5 out of the 9 RNAs, our method achieved a RMSD value even smaller than the minimum RMSD achieved by RNA-MoIP on each of the 5 RNAs. On the 6th RNA 2HOJ, our RMSD is 4.024 Å, close to the minimum RMSD 3.19 Å but much smaller than the average 7.19 Å achieved by RNA-MoIP. On two slightly longer RNA sequences 1LNG and 1MFQ, our predictions yielded RMSDs larger than what RNA-MoIP produced. We will further discuss the performance issue in next section.

5 Discussion and conclusion

Our method is a non-conventional framework to predict RNA nucleotide interactions (of all known types) without simultaneous prediction of 3D structure. The underlying backbone k-tree model drastically reduces the space of plausible nucleotide interaction relations, permitting not only efficient but also effective prediction of nucleotide interactions. The evaluation test results have highlighted the potential of our method as a viable step toward accurate 3D structure prediction of RNA sequences beyond short lengths.

Our work has taken advantage of the recent growth of knowledge in the rich, high-resolution nucleotide interaction data. In particular, our method predicts the most plausible set of interactions based on confidence scores of individual interactions computed with ANNs. The neural networks were trained and tested with a small subset of single chain RNAs (all of lengths $\leq$ 100 nucleotides) extracted from the established database RNA Structure Atlas. For each ANN that computes an individual interaction, a 10-folds cross validation was used to avoid overfitting. Indeed, test results on RNAs not in the training data, especially those of lengths $>$ 100, have apparently justified the rationale of the proposed confidence scores.

The evaluation tests have revealed that our method is robust in the sense that only choices of k for the k-tree model may affect the prediction results, especially on 3D modeling performance. Supplementary Table S8 in Supplementary Material shows that nucleotide interaction prediction by BkTree has allowed 3D modeling to achieve RMSD values $\leq$6.7Å for all but seven in the set of 43

RNAs. For those RNAs not in set T, all but one RNAs achieved the same low RMSDs. A careful look at Supplementary Table S4 in Supplementary Material shows that these RNAs are of more complex structures and their NIRs are actually beyond the capability of the backbone 3-tree model built in the program BkTree. In particular, column EdgeDiff of Table 4 shows that all these seven RNAs have more than a few edges in their NIR graphs which cannot be included by even the best backbone 3-tree model. For example, the 3-tree model can miss 6 and 9 edges in the NIR graphs of the long RNAs 1LNG and 1MFQ, respectively. These edges correspond to some important nucleotide interactions including those between the hairpins of two helices in these two signal recognition particle RNAs. Failure to predict these crossing interactions has resulted in the considerably high RMSD values in their 3D models. Supplementary Figure S5 in Supplementary Material shows the modeled 3D structure of 1LNG deviates from its native structure due to those missed interactions across the two hairpins.

Therefore, backbone k-tree models, for $k > 3$ are expected to improve performance for RNAs of complex structures. However, one major concern with such a model is the possibly impractical complexity $O(n^{k+1})$, for $k \geq 4$, of implementation with the developed DP algorithm. Our more recent study reveals that such obstacle can be surmounted by taking advantage of some inherent properties of backbone k-trees constructed from known RNA structures. For example, our survey on more than 600 RNAs with known 3D structures (data not shown) suggests that in backbone 4-tree models of these RNAs 5-cliques are basically of two types. One type of clique is that the 5 nucleotides can be partitioned into at most 3 groups, each containing vertices close to each other on the backbone. The other type of clique models 4–5 sporadic nucleotides as a part of a tertiary motif connecting 3 or 4 small regions of the backbone. Both type of 5-cliques are thus of number $O(n^3)$ in total, potentially leading to an $O(n^3)$-time (and space) implementation of the DP algorithm with the backbone 4-tree model for nucleotide interaction prediction.

The implemented neural networks for interaction pattern scoring were designed and trained exclusively for the constraint backbone 3-tree model. Thus to implement the backbone k-tree model with a higher k, an updated knowledge-based scoring system is needed. Using the same idea of the neural networks, due to the increased number of features, a feature selection scheme will need to be developed. Then the neural networks have to be retrained.

Acknowledgement

We thank Christian Laing for the provided original data used in the survey (Laing and Schlick, 2010) and Robert W. Robinson for his comments on an earlier version of the manuscript.

Funding

This work was supported in part by National Science Foundation (IIS 0916250).

Conflict of Interest: none declared.

References

Arnborg,S. and Proskurowski,A. (1989) Linear time algorithms for NP-hard problems restricted to partial k-trees. *Discrete Appl. Math.*, **23**, 11–24.

Arnborg,S. *et al.* (1990) Forbidden minors characterization of partial 3-trees. *Discrete Math.*, **80**, 1–19.

Bakan,S. *et al.* (2011) ProDy: protein dynamics inferred from theory and experiments. *Bioinformatics*, **27**, 1575–1577.

Berman,H.M. *et al.* (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

Bida,J.P. and Maher,L.J. III. (2012) Improved prediction of RNA 3D structure with insights into native state dynamics. *RNA*, **18**, 385–393.

Bodlaender,H.L. and Koster,A.M.C.A. (2010) Treewidth computations I. Upper bounds. *Inf. Comput.*, **208**, 259–275.

Das,R. and Baker,D. (2007) Automated *de novo* prediction of native-like RNA 3D structures. *Proc. Natl Acad Sci*, **104**, 14664–14669.

Das,R. *et al.* (2010) Atomic accuracy in predicting and designing noncanonical RNA structure. *Nat. Methods*, **7**, 291–294.

Ding,F. *et al.* (2008) *Ab initio* RNA folding by discrete molecular dynamics: From structure prediction to folding mechanisms. *RNA*, **14**, 1164–1173.

Ding,L. *et al.* (2014a) Stochastic k-tree grammar and its application in bio-molecular structure modeling. *Lect. Notes Comput. Sci.*, **8370**, 308–322.

Ding,L. *et al.* (2014b) Polynomial-time algorithms for maximum spanning k-tree constrained by Hamitonian path, in press.

Ding,L. *et al.* (2014c) *Ab initio* prediction of RNA nucleotide interactions with backbone k-tree model. In: *Proceedings of ECCB'14 Workshop on Computational Methods for Structural RNAs*, Strasbourg, pp. 25–42.

Gendron,P. *et al.* (2001) Quantitative analysis of nucleic acid three-dimensional structures. *J. Mol. Biol.*, **308**, 919–936.

Jonikas,M.A. *et al.* (2009) Coarse-grained modeling of large RNA molecules with knowledge-based potentials and structure filters. *RNA*, **15**, 189–199.

Jossinet,F. *et al.* (2010) Assemble: An interactive graphical tool to analyze and build RNA architectures at the 2D and tertiary levels. *Bioinformatics*, **26**, 2057–2059.

Laing,C. (2014) Personal communication.

Laing,C. *et al.* (2013) Predicting helical topologies in RNA junctions as tree graphs. *PLoS ONE*, **8**, e71947.

Laing,C. and Schlick,T. (2010) Computational approaches to tertiary modeling of RNA. *J. Phys. Condens. Matter*, **22**, 283101.

Leontis,N.B. and Westhof,E. (2001) Geometric nomenclature and classification of RNA base pairs. *RNA*, **7**, 499–512.

Leontis,N.B. and Westhof,E., (2012) *RNA 3D structure analysis and prediction*. Springer, Berlin Heidelberg.

Leontis,N.B. *et al.* (2002) The non-Watson–Crick base pairs and their associated isostericity matrices. *Nucleic Acids Res.*, **30**, 3497–531.

Hall,M. *et al.* (2009) The WEKA data mining software: an update. *ACM SIGKDD Expl. Newsl.*, **11**, 10–18.

Martinez,H.M. *et al.* (2008) RNA2Dtertiary: a program for generating, viewing, and comparing 3-dimensional models of RNA. *J. Biomol. Struct. Dyn.*, **25**, 669–683.

Mitchell,T. (1997) *Machine Learning*. McGraw Hill, New York, NY.

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

Parisien,M. *et al.* (2009) New metrics for comparing and assessing discrepancies between RNA 3D structures and models. *RNA*, **15**, 1875–1885.

Patil,H.P. (1986) On the structure of k-tree. *J. Comb. Inf. Syst. Sci.*, **11**, 57–64.

Pinhas,T. *et al.* (2014) Efficient edit distance with duplications and contractions. *Algor. Mol. Biol.*, **8**, 27.

Popenda,M. *et al.* (2012) Automated 3D structure composition for large RNAs. *Nucleic Acids Res.*, **40**, e112.

Reinharz,V. *et al.* (2013) Towards 3D structure prediction of large RNA molecules: an integer programming framework to insert local tertiary motifs in RNA secondary structure. *Bioinformatics*, **28**, i207–i214.

Salomon-Ferrer,R. *et al.* (2012) An overview of the Amber biomolecular simulation package, *WIREs Comput. Mol. Sci.*, **3**, 198–210

Sarver,M. *et al.* (2008) FR3D: finding local and composite recurrent structural motifs in RNA 3D structures. *J. Math. Biol.*, **56**, 215–252.

Sharma,S. *et al.* (2008) iFoldRNA: three-dimensional RNA structure prediction and folding. *Bioinformatics*, **24**, 1951–1952.

Stombaugh,J. *et al*. (2009) Frequency and isostericity of RNA base pairs. *Nucleic Acids Res*., **37**, 2294–2312.

Van Leeuwen,J. (1990) Graph algorithms. *Handbook of Theoretical Computer Science, A: Algorithms and Complexity theory*, MIT Press Cambridge, MA, USA.

Zirbel,C.L. *et al*. (2009) Classification and energetics of the base-phosphate interactions in RNA. *Nucleic Acids Res*., **37**, 4898–4918.

Zirbel,C.L. *et al*. (2011) FR3D list of base-phosphate and base-ribose interactions in 1EHZ http://rna.bgsu.edu/FR3D/AnalyzedStructures/1EHZ/1EHZ_base_phosphate.html.