EFFICIENT ANNOTATION OF NON-CODING RNA STRUCTURES INCLUDING PSEUDOKNOTS VIA AUTOMATED FILTERS

Chunmei Liu* and Yinglei Song
Department of Computer Science, University of Georgia
Athens, Georgia 30602, USA
Email: {chunmei, song}@cs.uga.edu

Ping Hu
Department of Genetics, University of Georgia
Athens, GA 30602, USA
Email: huping@uga.edu

Russell L. Malmberg
Department of Plant Biology, University of Georgia
Athens, GA 30602, USA
Email: russell@plantbio.uga.edu

Liming Cai*
Department of Computer Science, University of Georgia
Athens, Georgia 30602, USA
Email: cai@cs.uga.edu

Computational search of genomes for RNA secondary structure is an important approach to the annotation of non-coding RNAs. The bottleneck of the search is sequence-structure alignment, which is often computationally intensive. A plausible solution is to devise effective filters that can efficiently remove segments unlikely to contain the desired structure patterns in the genome and to apply search only on the remaining portions. Since filters can be substructures of the RNA to be searched, the strategy to select which substructures to use as filters is critical to the overall search speed up. Such an issue becomes more involved when the structure contains pseudoknots; approaches that can filter pseudoknots are yet available. In this paper, a new effective filtration scheme is introduced to filter RNA pseudoknots. Based upon the authors' earlier work in tree-decomposable graph model for RNA pseudoknots, the new scheme can automatically derive a set of filters with the overall optimal filtration ratio. Search experiments on both synthetic and biological genomes showed that, with this filtration approach, RNA structure search can speed up 11 to 60 folds whiling maintaining the same search sensitivity and specificity of without the filtration. In some cases, the filtration even improves the specificity that is already high.

1. INTRODUCTION

Non-coding RNAs (ncRNAs) do not encode proteins yet they play fundamental roles in many biological processes including chromosome replication, RNA modification, and gene regulation [7, 19, 28]. Due to the explosive growth of fully sequenced genome data, homologous searching using computational methods has recently become an important approach to annotating genomes and identifying new ncRNAs [16, 21, 22]. In general, a computational searching tool scans through a genome and aligns its sequence segments to an RNA profile. Since secondary structure generally determines the biological functions of an ncRNA and is preserved across its homologs, a profile needs to include both sequence conservation and secondary structure information. For example, compared with profiling models based on Hidden Markov Models (HMMs) [14], Covariance models (CMs) [6] contain additional emission states that can emit base pairs to generate stems. CMs can thus be used as structural profiles to model RNA families. However, for *pseudoknots*, which contain at least one pair of crossing stems, the sequence-structure alignment is computationally intractable. RNA structure search

*To whom correspondence should be addressed.

in genomes or large databases thus remains difficult.

Search on genomes can be speeded up with filtration methods. With simpler sequence or structural models, it is possible to efficiently remove genome segments unlikely to contain the desired pattern. A few filtration methods Ref. 2, 16, 27 have been developed to improve the search efficiency. For example, in tRNAscan-SE [16], two efficient tRNA detection algorithms are used as filters to preprocess a genome and remove most parts that are unlikely to contain the searched tRNA structure. The remaining part of the genome is then scanned with a CM to identify the tRNA. FastR [2] considers the structural units of an RNA structure. It evaluates the specificity of each structural unit and construct filters based on the specificities of these structural units. In [27], an algorithm is developed to safely "break" the base pairs in an RNA structure and automatically select filters from the resulting HMM models. These approaches have significantly improved the computational efficiency of genome annotation. However, all of them have yet been applied to search for structures that contain pseudoknots.

Filters, like the structure to be searched, need to be profiled with appropriate models. Most of the existing searching tools [3, 13, 15, 16] use CMs to profile the secondary structure of an ncRNA. While CM based searching tools can achieve high accuracy, they are incapable of modeling pseudoknots. In addition, the time complexity for optimally aligning a sequence segment to a CM profile is too high for a thorough search of a genome [13]. A few models [4, 20, 23, 26] based on stochastic grammar systems have been proposed to profile pseudoknot structures. However, for all these models, the computation time and memory space costs needed for optimal structure-sequence alignment are $O(N^5)$ and $O(N^4)$ respectively. In practice, these models cannot be directly used for searching. Heuristic approaches [3, 8, 15] can significantly improve the search efficiency for pseudoknots. These approaches either cannot guarantee the search accuracy [8] or have the same drawback in computation efficiency as CM based approaches [3, 15].

A tree decomposable graph model has been introduced in our previous work [25]. In particular, the secondary structure of RNAs is modeled as a conformational graph, while a queried sequence segment is modeled with an image graph with valued vertices and edges. The sequence-structure alignment can be determined by finding in the image graph the maximum valued subgraph that is isomorphic to the conformational graph. Based on a tree decomposition of the conformational graph with tree width t, a sequence-structure alignment can be accomplished in time $O(k^t N^2)$ [25], where k is a small parameter (practically $k \leq 7$), and N is the size of the conformational graph. The tree width t of the RNA conformational graph is very small, e.g, $t = 2$ for pseudoknot-free RNAs and can only increase slightly for pseudoknots. Experiments have shown that this approach is significantly faster than CM based searching approaches while achieving an accuracy comparable with that of CM.

In this paper, based on the tree decomposable model, we develop a novel approach of filtration. In particular, based on the profiling model in our previous work, a subtree formed by tree nodes containing either of the two vertices that form a stem can be used as a filter. A filter can thus be constructed for each vertex in the conformational graph. Based on the intersection relationship among the subtrees of filters, we are able to construct a filter graph. In the graph each vertex represents a maximal subtree and two vertices are connected with an edge if the corresponding subtrees intersect. We associate every vertex in the filter graph a weight, which is the filtration ratio of the filter that can be measured based on randomly generated sequences. We thus select filters that correspond to the maximum weighted independent set in the graph. A filter graph is a chordal graph and we thus are able to compute its maximum weighted independent set in time $O(n^2)$, where n is its number of vertices. Filters can thus be selected in time $O(n^2)$.

We have implemented this filter selection algorithm and combined it with the original tree decomposition based searching program to improve its computational efficiency. To test its accuracy and computational efficiency, we used this combined search tool to search for RNA structures inserted into random generated sequences. Our testing results showed that, compared with the original searching program, this filtering approach is significantly faster and can achieve improved specificity. Specifically, it achieved

20 to 60 fold speed up for pseudoknot-free RNAs and 11 to 45 fold speedup for RNAs containing pseudoknots. In addition, for some tested structures, this approach is able to achieve an improvement in specificity from about 80% to 92%. We then used this combined searching tool to search a few biological genomes for ncRNAs. Our testing results showed that this combined program can accurately determine the locations of these ncRNAs with significantly reduced computational time, e.g, compared with the original searching program, it achieved 6 to 142 fold speed up for genome searchings for pseudoknots.

2. ALGORITHMS AND MODELS

2.1. Tree Decomposable Graph Model

In our previous work [25], the consensus secondary structure of an RNA family was modeled as a topological relation among stems and loops. The model consists of two components: a *conformational graph* that describes the relationship among all stems and loops and a set of simple statistical profiles that model individual stems and loops. In the conformational graph, each vertex defines one of the base pairing regions of a stem. The graph contains both directed and undirected edges. Each undirected edge connects two vertices that form the pairing regions of a stem. In addition, the vertices for two base regions are connected with a directed edge (from 5' to 3') if the sequence part between them is a loop. Technically, two additional vertices s (called *source*) and t (called *sink*) are included in the graph. Figure 1(a) and (b) show the consensus structure of an RNA family and its conformational graph. In general, we can construct a consensus structure from the multiple structural alignment of a family of RNAs. In this model, in addition to the conformational graph, individual stems are profiled with the Covariance Model (CM) [6], and loops are profiles with HMM [14].

To align the structure model to a target sequence, we first preprocess the target sequence to identify all possible matches to each individual stem profile. All pairs of regions with statistically significant alignment score, called the *images* of the stem, are identified. Then an *image graph* is constructed from the set of images for all stems in the structure. In particular, each vertex represents an image for one pairing region of a stem; two vertices for the base pairing regions of a stem are connected with a non-directed edge. In addition, a directed edge connects the vertices for two non-overlapping base regions (5' to 3'). To reduce the complexity of the graph, a parameter k is used to define the maximum number of images that a stem can map to. It can be computed based on a statistical cut-off value and its value is generally small in nature. Figure 1(c) and (d) illustrate the mapping from stems to their images and the corresponding image graph constructed.

The optimal structure-sequence alignment between the structure model and the target sequence thus corresponds to finding in the image graph a maximum weighted subgraph that is isomorphic to the conformational graph. The weight is defined by the alignment score between vertices (stems) and edges (loops) in the conformational graph and their counterparts in the image graph. The subgraph isomorphism problem is NP-hard. Interestingly, the conformational graph for the RNA secondary structure is tree decomposable; efficient isomorphism algorithms are possible.

Definition 2.1 ([24]). *Let $G = (V, E)$ be a graph, where V is the set of vertices in G, E denotes the set of edges in G. Pair (T, X) is a tree decomposition of graph G if it satisfies the following conditions:*

(1) $T = (I, F)$ defines a tree, the sets of vertices and edges in T are I and F respectively,
(2) $X = \{X_i | i \in I, X_i \subseteq V\}$, and $\forall u \in V$, $\exists i \in I$ such that $u \in X_i$,
(3) $\forall (u, v) \in E$, $\exists i \in I$ such that $u \in X_i$ and $v \in X_i$,
(4) $\forall i, j, k \in I$, if k is on the path that connects i and j in tree T, then $X_i \cap X_j \subseteq X_k$.

The tree width of the tree decomposition (T, X) is defined as $\max_{i \in I} |X_i| - 1$. The tree width of the graph G is the minimum tree width over all possible tree decompositions of G.

Figure 2 provides an example for a tree decomposition of a given graph. Tree decomposition is a technique rooted in the deep graph minor theorems [24]; it provides a topological view on graphs. Tree width of a graph measures how much the graph is "tree-like". Conformational graphs for the RNA secondary structure have small tree width. For example,

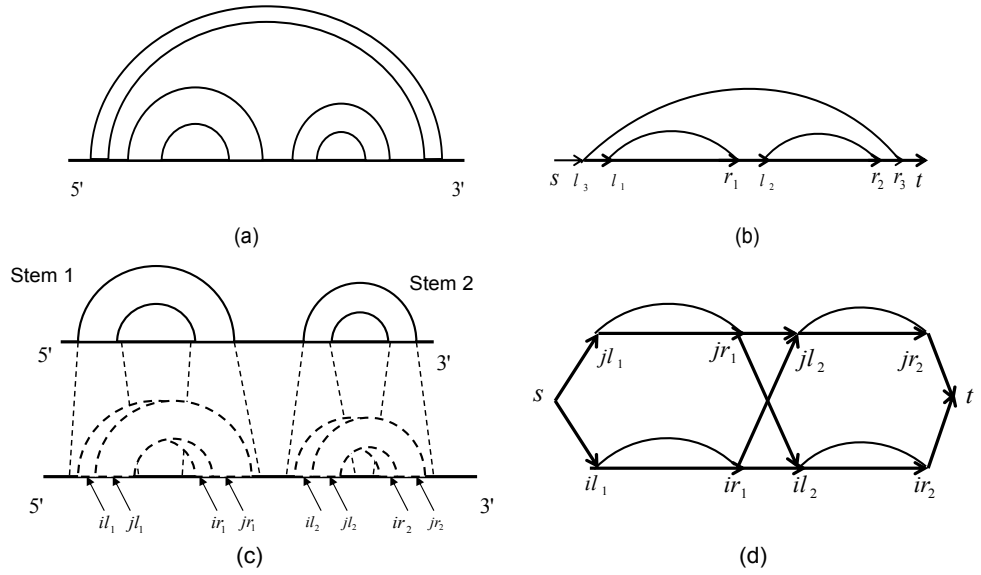

Fig. 1. (a) An RNA structure that contains both nested and parallel stems. (b) The corresponding conformational graph. (c) A secondary structure (top), and the mapped regions and images for its stems on the target sequence (bottom). The dashed lines specify the possible mappings between stems and their images. (d) The image graph formed by the images of its stems on a target sequence. (il_1, ir_1) and (jl_1, jr_1) for stem 1, and (il_2, ir_2) and (jl_2, jr_2) for stem 2.

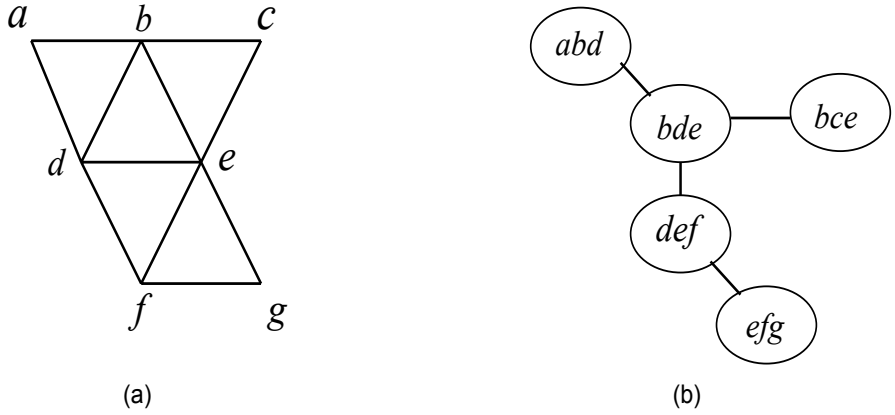

Fig. 2. (a) An example of a graph. (b) A tree decomposition for the graph in (a).

the tree width is 2 for the graph of any pseudoknot-free RNA and it can only increase slightly for all known pseudoknot structures [25]. For instance, the conformational graph shown in Figure 5 for sophisticated bacterial tmRNAs has tree width 5.

We showed in our previous work [25] that given a tree decomposition of the conformational graph with tree width t, the maximum weighted subgraph isomorphism can be efficiently found in time $O(k^t N^2)$, where N is the length of the structure model and k is the maximum number of images that a stem can map to.

2.2. Automated Structure Filter

We observe that any subtree in a tree decomposition of a conformational graph induces a substructure and is thus a structure profile of smaller size. It can be used as a filter to preprocess a genome to be annotated. In particular, the left and right regions of any stem s_i in an RNA structure have two corresponding vertices v_i^l and v_i^r in its conformational graph. In the tree decomposition of the conforma-

tional graph, these two vertices induce a maximal connected subtree T_i, in which every node contains either of the vertices. We choose subtrees with this maximal property since each of them contains the maximum amount of structural information associated with the stem. This is also to ensure that when the RNA structure contain a simple pseudoknot, the pseudoknot will be included in some filter.

This way, we thus can obtain up to $O(N)$ such subtrees, where N is the size of the conformational graph. However, subtrees may intersect and it would be more desirable to select a set of disjoint subtrees to preprocess the genome. For this, we construct a *filter graph* as follows. In the graph each vertex represents a maximal subtree defined above and two vertices are connected with an edge if the corresponding subtrees intersect. Figure 3 shows an example for the filter graph of a given RNA structure.

We associate every vertex in the filter graph a weight, which is the *filtration ratio* of the filter resulted from the corresponding subtree. The filtration ratio of a filter is defined as the percentage of nucleotides that pass the corresponding filtration process and it is obtained as follows. For each filter, we randomly generate a sequence of sufficient length and compute the distribution of the scores of alignment between the filter profile and all the sequence segments in the generated sequence. For a filter with filtration ratio f, we assign a weight of -ln f to its corresponding vertex. To achieve a minimum filtration ratio, we need to find the maximum weighted independent set in the filter graph. We show in the following that this independent set can be found easily.

According to [10], the filter graph constructed from a tree decomposition is actually a chordal graph, in which any cycle with length larger than 3 contains a chord. Also for any chordal graph, there exists a tree decomposition for the graph such that the vertices contained in every tree node induce a clique and the tree decomposition can be found in time $O(|V|^2)$, where V is the vertex set of the chordal graph [9]. Then given such a tree decomposition, a simple dynamic programming algorithm can be developed to find the maximum weight independent set.

Theorem 2.1. *For an RNA secondary structure that contains n stems, there exists an algorithm of time $O(n^2)$ that can select a set of disjoint filters with the maximum filtration ratio.*

2.3. Filter-Sequence Alignment

For a given filter F, the vertices contained in the tree bags of its corresponding subtree induce a subgraph in the conformational graph; such an induced subgraph is its *filter conformational graph*. An alignment between a structural filter profile and a target sequence is essentially an isomorphism between its filter conformational graph H and some subgraph of the image graph G for the target sequence. To find such an isomorphism, we adopt the general dynamic programming technique [1] over the tree decomposition of H. However, since the general technique can only be directly applied to a subgraph isomorphism on small fixed graph H and graph G of a small tree width [17], we introduce some additional techniques to solve the problem in our setting. We present a summary and some details of the new optimal alignment algorithm in the following.

The dynamic programming over the tree decomposition to find an optimal alignment is based on the maintenance of a dynamic programming table for each node in the tree. An entry in a table includes a possible combination of images of vertices in the corresponding tree node and the validity and partial optimal alignment score associated with the combination. The table thus contains a column allocated for each vertex in the node and two additional columns V and S to maintain validity and partial optimal alignment scores respectively.

In a bottom up fashion, the algorithm first fills the entries in the tables for all leaf nodes. Specifically, for vertices in a leaf node, a combination of their images is valid if the corresponding mapping satisfies the first two conditions for isomorphism (see section 2) and the partial optimal alignment score for a valid combination is the sum of the alignment scores of loops and stems induced by images of vertices that are only contained in the node. For an internal node X_i in the tree, without loss of generality, we assume X_j and X_k are its children nodes. For a given combination e_i of images of vertices in X_i, the algorithm checks the first two conditions for isomorphism (section 2 in [25]) and sets e_i to be in-

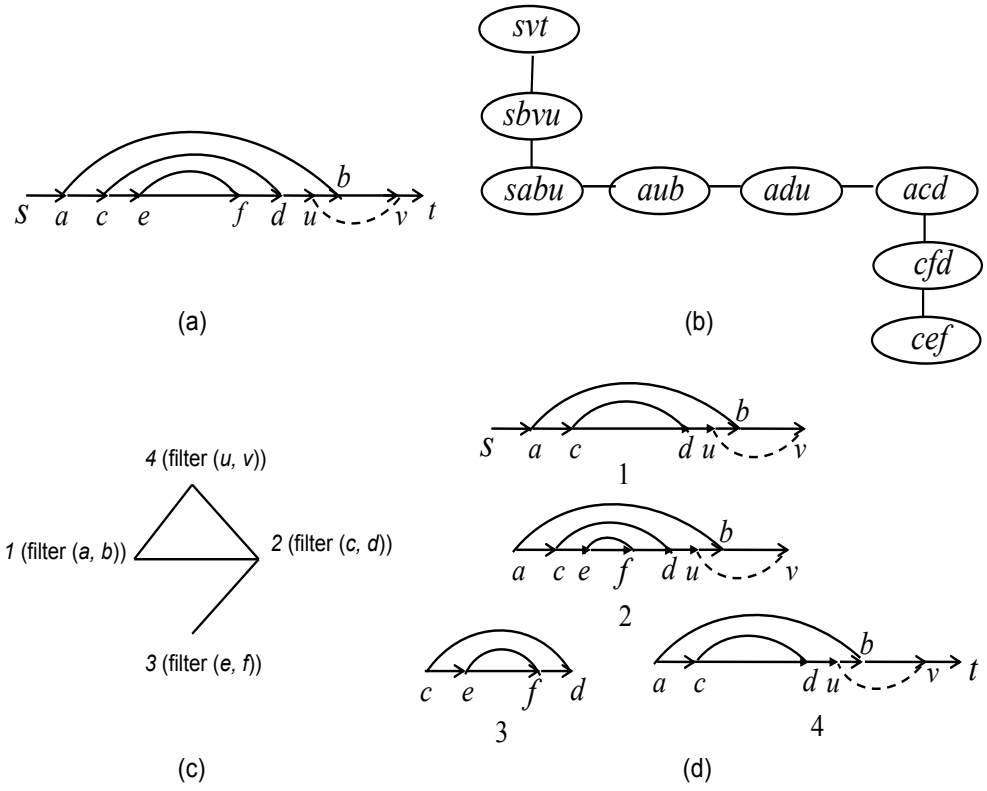

Fig. 3. (a) The conformational graph for a secondary structure that includes a pseudoknot. (b) A tree decomposition for the graph in (a). (c) A filter graph for the secondary structure in (a). (d) Substructures of the filters.

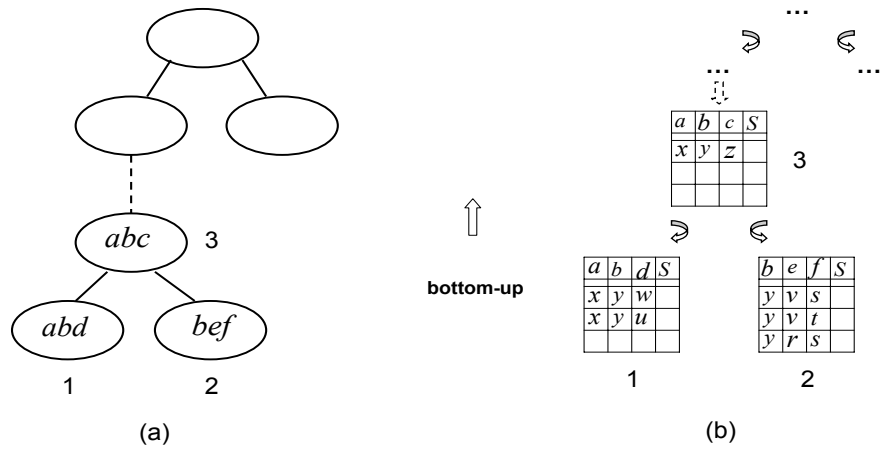

Fig. 4. A sketch of the dynamic programming approach for optimal alignments. The algorithm maintains a dynamic programming table in each tree node. Starting with leaves of the tree, the algorithm follows a bottom-up fashion. In computing the table for a parent node, only combinations of the images of the vertices in the node are considered. In every such combination, only one locally best combination (computed in the children tables) is used for vertices that occur in the children nodes but not in the parent node.

valid if one of them is not satisfied. Otherwise, the algorithm queries the tables for X_j and X_k. e_i is set to be valid if and only if there exist valid entries e_j and e_k from the tables of X_j and X_k such that e_j and e_k have the same assignments of images as that of e_i for vertices in $X_i \cap X_j$ and $X_i \cap X_k$ re-

spectively. The partial optimal alignment score for a valid entry e_i includes the alignment scores of stems and loops induced by images of vertices only in X_i and the maximum partial alignment scores over all valid entries e_j's and e_k's with the same assignments of images for vertices in $X_i \cap X_j$ and $X_i \cap X_k$ as that of e_i in tables for X_j and X_k respectively. Figure 4 provides an example for the overall algorithm.

The alignment score is the sum of the scores for aligning individual stems and loops in the structure profile. The alignment score for a stem is calculated between the stem profile and a chosen image in the target of the stem. Since any loop in the structure is between some two stems, the alignment score for a loop is calculated between its profile and the sequence segment in the target within the two chosen images for the two stems. The time complexity for this dynamic programming approach is $O(k^t N^2)$, where k is the number of images for each vertex in the conformational graph, t is the tree width of its tree decomposition and N is its number of vertices.

3. EXPERIMENTAL RESULTS

We performed experiments to test the accuracy and efficiency of this filtration based approach and compared it with that of the original tree decomposition based program. The training data was obtained from the Rfam database [12]. For each family, we choose up to 60 sequences with pair-wise identities lower than 80% from the structural alignment of seed sequences.

In practice, to obtain a reasonably small value for the parameter k, the upper bound on the number of images that a stem can map to, we constrain the images of a stem within certain region, called the *constrained image region* of the stem, in the target sequence. We assume that for homologous sequences, the distances from the pairing region of a given stem to the 3' end follow a Gaussian distribution. For a stem, we compute the mean and standard deviation of distances from its two pairing regions to the 3' end of the sequence respectively, evaluated over all training sequences. For training data representing distant homologs of an RNA family with structural variability, we can effectively divide data into groups so that a different but related profile can be built for each group and used for searches. This ensures a small value for the parameter k in the models.

As a first profiling and searching experiment, we inserted several RNA sequences from the same family into a random background generated with the same base composition as the sequences in the family. We then used this filtration based approach and the original tree decomposition based program to search for the inserted sequences. We compared the sensitivity and specificity of both approaches on several different RNA families. Finally, we tested the performance of our approach by searching for non-coding RNA genes in real biological genomes.

3.1. On Pseudoknot-Free Structures

We implemented this filter selection algorithm and combined it with our tree decomposition based searching tool to improve searching efficiency. To test its accuracy and computational efficiency, we used this program to search for about 30 pseudoknot-free RNA structures inserted in a random background of 10^5 nucleotides generated with the same base composition as the RNA structure. In particular, we computed the filtration ratio of each selected filter with a random sequence of 10000 nucleotides, which is generated with the same base composition as that of the sequence to be searched. The statistical distribution of alignment scores for each filter and the overall structural profile is determined on the same sequence using a method similar to that used by RSEARCH [13]. To improve the computational efficiency, we determine the maximum size of the substructure for each filter; a window with a size that is about 1.2 times of this value is used for searching while this filter is used.

The order that the selected filters are applied is critical to the performance of searching. However, the number of possible orders for l selected filters is up to $l!$ and we thus are unable to exhaustively search through all possible orders and find the best one. In practice, we develop a heuristic method to determine the order of filters. In particular, we consider both the filtration ratio and the computation time of a filter. For each selected filter, we associate it with the value $\frac{\ln f}{T}$, where f is its measured filtration ratio and T is the computation time needed for the filter to scan the testing sequence. We then apply the structural profiles of filters to scan the target sequence with an increasing order of this value.

A sequence segment passes the screening of a filter if its corresponding alignment Z-score is larger than 2.0. For final processing, we use the original tree decomposition based algorithm to process the remaining sequence segments. An alignment Z-score larger than 5.0 is reported as a hit. In our experiments, for each stem, the algorithm selects k images with the maximum alignment scores within the constrained image region of the stem. In order to evaluate the impact of the parameter k on the accuracy of the algorithm, we carried out the same searching experiments for each given k. Table 1 shows the number of filters selected for each tested structure and the filtration ratio for the one that is first applied to scan the genome.

Table 2 shows that on the tested RNA families, the filtration based approach achieves the same or better searching accuracy than that of the original approach. In particular, a significant improvement on specificity is observed on a few tested families. From Table 3, compared to the original approach, the filtration based approach consumes a significantly reduced amount of computation time. On most of the tested families, the filtration based searching is more than 30.0 times faster than our original approach.

3.2. On Pseudoknot Structures

We also performed searching experiments on several RNA families that contain pseudoknot structures. For each family, we inserted about 30 structures that contain pseudoknots into a background randomly generated with the same base composition as that of the inserted sequences. The training data was also obtained from the Rfam database [12] where we selected up to 40 sequences with pair wise identity lower than 80% from the seed alignment for each family.

For each tested pseudoknot structure, the filtration ratio for the first filter that is applied to scan the genome is shown in Table 4. Tables 5 and 6 compare the searching accuracy and efficiency between the filtration based approach and the original one. It is evident that on families with pseudoknots, the filtration based algorithm achieves the same accuracy as that of the CM based algorithm when parameter k reaches a value of 7. In addition, the filtration based approach is more than 20 times faster than the original approach on most of the tested pseudoknot structures.

3.3. On Biological Genomes

We used the program to search biological genomes for structural patterns that contain pseudoknots: corona virus genomes, tmRNA, and telomerase RNAs. For example, the secondary structure formed by nucleotides in the 3' untranslated region in the genomes of the corona virus family contains a pseudoknot structure. This pseudoknot was recently shown to play important roles in the replication of the viruses in the family [11]. We selected four genomes from the corona virus family and used the algorithm to search for this pseudoknot. For bacteria, the tmRNA is essential for the trans-translation process and is responsible for adding a new C-terminal peptide tag to the incomplete protein product of a broken mRNA [18]. The secondary structure of tmRNA contains four pseudoknots; Figure 5 provides a sketch of the stems that constitute the secondary structure of a tmRNA. The tree decomposition based algorithm was also used to search for tmRNA genes on the genomes of two bacteria organisms, *Haemophilus influenzae* and *Neisseria meningitidis*. Both of the genomes contain more than 10^6 nucleotides. Among the bacteria containing tmRNAs, these two are relatively distant from each other evolutionarily. To test the accuracy and efficiency of the algorithm on genomes with a significantly larger size, we used the algorithm to search for the telomerase RNA gene in the genomes of two yeast organisms, *Saccharomyces cerevisiae* and *Saccharomyces bayanus*, both of which contain more than 10^7 nucleotides. Telomerase RNA is responsible for the addition of some specific simple sequences onto the chromosome ends [5].

The parameter k used in the tree decomposition based algorithm for searching all genomes is 7. Table 4 also shows the filtration ratio of the first applied filter obtained on different values of k for each pseudoknot structure. Table 7 provides the real locations of the searched patterns and the identified location offsets deviating from the real locations annotated by the filtration based and the original approaches respectively. The table clearly shows that compared with the original approach, the filtration based approach is able to achieve the same accuracy with a

Table 1. The number of filters selected on tested pseudoknot free structures. For each structure, the filtration ratio for the first filter used to scan the genome is also shown.

RNA	Number of Selected Filters	Filtration Ratios		
		$k = 6$	$k = 7$	$k = 8$
EC	1	0.147	0.084	0.084
EO	1	0.082	0.049	0.049
Let_7	2	0.110	0.074	0.055
Lin_4	3	0.045	0.030	0.030
Purine	1	0.042	0.042	0.021
SECIS	1	0.089	0.036	0.036
S_box	3	0.189	0.189	0.189
TTL	2	0.093	0.056	0.056

EC, EO and TTL represent Entero_CRE, Entero_OriR, and Tymo_tRNA-like respectively.

Table 2. A comparison of the searching accuracy of filtration based approach and the original tree decomposition based program in terms of sensitivity and specificity.

RNA	Without Filtration						With Filtration					
	$k = 6$		$k = 7$		$k = 8$		$k = 6$		$k = 7$		$k = 8$	
	SE	SP	SE	SP	SE	SP	SE	SP	SE	SP	SE	SP
EC	100	80.65	100	80.65	100	80.65	100	91.18	100	93.93	100	96.87
EO	100	100	100	100	100	100	100	100	100	100	100	100
Let_7	95.8	100	100	100	100	100	95.8	100	100	100	100	100
Lin_4	100	94.11	100	94.11	100	94.11	100	100	100	100	100	100
Purine	93.10	96.43	93.10	96.43	93.10	96.43	93.10	96.43	93.10	100	93.10	100
SECIS	100	97.30	100	97.30	100	97.30	100	97.30	100	97.30	100	97.30
S_box	100	92.86	100	96.30	100	96.30	100	96.30	100	100	100	100
TTL	100	96.67	100	96.67	100	96.67	100	96.67	100	96.67	100	96.67

SE and SP are sensitivity and specificity in percentage respectively.

Table 3. The computation time for both approaches on all pseudoknot free RNA families.

RNA	Without Filtration			With Filtration					
	$k = 6$	$k = 7$	$k = 8$	$k = 6$		$k = 7$		$k = 8$	
	RT	RT	RT	RT	SU	RT	SU	RT	SU
EC	2.85	3.21	3.38	0.07	40.71×	0.08	40.13×	0.11	30.73×
EO	4.91	5.26	5.42	0.17	28.88×	0.23	22.87×	0.27	20.07×
Let_7	14.97	16.38	16.92	0.24	62.38×	0.31	52.84×	0.34	49.76×
Lin_4	3.22	4.25	5.10	0.11	29.27×	0.14	30.36×	0.16	31.87×
Purine	7.09	8.49	9.61	0.25	28.36×	0.33	25.72×	0.38	25.29×
SECIS	9.14	10.23	10.89	0.15	60.94×	0.20	51.15×	0.23	39.73×
S_box	29.76	34.76	41.01	1.22	24.39×	1.71	20.33×	1.81	22.65×
TTL	5.01	6.10	7.07	0.20	25.05×	0.24	25.42×	0.30	23.57×

RT is the computation time in minutes; SU is the amount of speed up compared to the original approach.

significantly reduced amount of computation time. Both programs achieve 100% sensitivity and specificity for searches in genomes. The table also shows that on real biological genomes, the selected filter sets can effectively screen out the parts of the genome that do not contain the desired structures and thus improve the searching efficiency.

4. CONCLUSIONS

In this paper, we develop a new approach to improve the computational efficiency for annotating non-coding RNAs in biological genomes. Based on the graph theoretical profiling model proposed in our previous work, we develop a new filtration model that uses subtrees in a tree decomposition of the conformational graph as filters. This new filtering approach can be used to search genomes for structures

Table 4. The number of filters selected on tested pseudoknot structures. For each structure, the filtration ratio for the first filter used to scan the genome is also shown.

RNA	Number of Selected Filters	Filtration Ratios		
		$k = 6$	$k = 7$	$k = 8$
Alpha_RBS	3	0.095	0.071	0.071
Antizyme_FSE	1	0.078	0.066	0.042
HDV_ribozyme	3	0.030	0.030	0.010
IFN_gamma	5	0.069	0.035	0.035
Tombus_3_IV	3	0.067	0.048	0.048
corona_pk3	1	0.028	0.014	0.014
PK3	1	0.027	0.013	0.013
tmRNA	11	0.220	0.220	0.070
Telomerase	2	0.130	0.130	0.130

Table 5. The search sensitivity (SE) and specificity (SP) for both filtration based and original approaches on RNA sequences containing pseudoknots.

RNA	Without Filtration						With Filtration					
	$k = 6$		$k = 7$		$k = 8$		$k = 6$		$k = 7$		$k = 8$	
	SE	SP	SE	SP	SE	SP	SE	SP	SE	SP	SE	SP
Alpha_RBS	95.80	92.00	100	96.00	100	96.00	95.80	96.0	100	96.0	100	96.0
Antizyme_FSE	96.43	100	100	100	100	100	92.86	100	100	100	100	100
HDV_robozyme	100	97.37	100	97.37	100	97.37	100	97.37	100	97.37	100	97.37
IFN_gamma	100	100	100	100	100	100	90	100	100	100	100	100
Tombus_3_IV	100	100	100	100	100	100	100	100	100	100	100	100
corona_pk3	100	97.37	100	97.37	100	97.37	97.30	100	100	100	100	100

Table 6. The computation performance for both searching algorithms on some RNA families that contain pseudoknots.

RNA	Without Filtration			With Filtration					
	$k = 6$	$k = 7$	$k = 8$	$k = 6$		$k = 7$		$k = 8$	
	RT	RT	RT	RT	SU	RT	SU	RT	SU
Alpha_RBS	0.31	0.42	0.55	0.02	15.50×	0.03	14.00×	0.05	11.00×
Antizyme_FSE	0.13	0.18	0.23	0.003	43.33×	0.004	45.00×	0.006	38.33×
HDV_ribozyme	0.34	0.52	0.79	0.01	34.00×	0.02	26.00×	0.03	26.33×
IFN_gamma	0.72	1.07	1.52	0.04	18.00×	0.05	21.40×	0.06	25.33×
Tombus_3_IV	0.27	0.40	0.57	0.01	27.00×	0.03	13.33×	0.05	11.40×
corona_pk3	0.15	0.20	0.26	0.005	30.00×	0.007	28.57×	0.01	26.00×

The amount of RT is in hours; SU is the amount of speed up compared to the original approach.

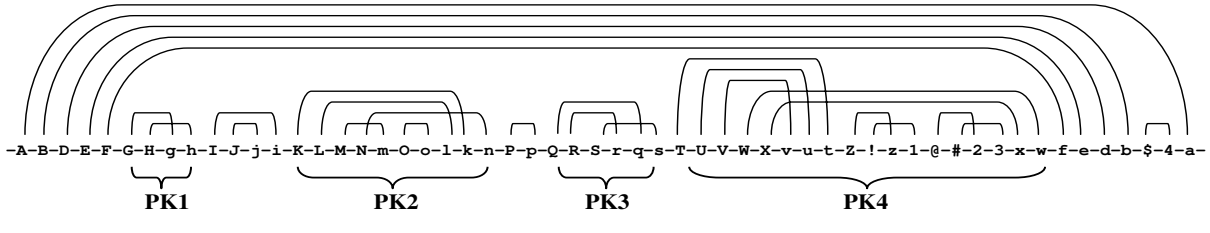

Fig. 5. Diagram of stems in the secondary structure of a tmRNA. Upper case letters indicate base regions that pair with the corresponding lower case letters. The four pseudoknots constitute the central part of the tmRNA gene and are labeled as Pk1, Pk2, Pk3, Pk4 respectively.

containing pseudoknots with high accuracy. Compared to the original method, a significant amount of speed up is also achieved. More importantly, this filtering method allows us to apply more sophisticated sequence-structure alignment algorithm on the remaining portions of the genome. For example, we

Table 7. A comparison of the accuracy and efficiency for both algorithms on searching biological genomes.

OR	ncRNA	Without Filtration			With Filtration				Real location		GL
		L	R	RT	L	R	RT	SU	Left	Right	
BCV	3'PK	0	0	0.053	0	0	0.008	6.63×	30798	30859	0.31
MHV	3'PK	0	0	0.053	0	0	0.007	7.57×	31092	31153	0.31
PDV	3'PK	0	0	0.048	0	0	0.004	12.00×	27802	27882	0.28
HCV	3'PK	0	0	0.047	0	0	0.006	7.83×	27063	27125	0.27
HI	tmRNA	−1	−1	44.0	−1	−1	0.32	137.50×	472210	472575	18.3
NM	tmRNA	0	0	52.9	0	0	0.37	142.97×	1241197	1241559	22.0
SC	TLRNA	−3	−1	492.3	−3	−1	8.74	56.33×	307691	308430	103.3
SB	TLRNA	−3	2	550.2	−3	2	9.28	59.29×	7121532	7122282	114.8

OR is the name of the organism; GL is the length of the genome in multiples of 10^5 nucleotides. BCV is Bovine corona virus; MHV is Murine hepatitus virus; PDV is Porcine diarrhea virus; HCV is Human corona virus; HI and NM represent *Haemophilus influenzae* and *Neisseria meningitidis* respectively, and SC and SB represent *Saccharomyces cerevisiae* and *Saccharomyces bayanus* respectively. L and R are the left and right offsets of the resulting locations respectively compared to the real locations. RT is the single CPU time needed to identify the ncRNA in hours. For tmRNA and telomerase RNA searches, RT was estimated from the time needed by a parallel search with 16 processors. SU is the amount of speed up compared to the original approach.

are able to search remote homologs of a sequence family using a few alternative profiling models for each stem or loop. This approach can be used to find remote homologs with unknown secondary structure.

References

1. S. Arnborg and A. Proskurowski, "Linear time algorithms for NP-hard problems restricted to partial k-trees.", *Discrete Applied Mathematics*, 23: 11-24, 1989.
2. V. Bafna and S. Zhang, "FastR: Fast database search tool for non-coding RNA.", *Proceedings of the 3rd IEEE Computational Systems Bioinformatics Conference*, 52-61, 2004.
3. M. Brown and C. Wilson, "RNA Pseudoknot Modeling Using Intersections of Stochastic Context Free Grammars with Applications to Database Search.", *Pacific Symposium on Biocomputing*, 109-125, 1995.
4. L. Cai, R. Malmberg, and Y. Wu, "Stochastic Modeling of Pseudoknot Structures: A Grammatical Approach.", *Bioinformatics*, 19, $i66 - i73$, 2003.
5. A. T. Dandjinou, N. Lévesque, S. Larose, J. Lucier, S. A. Elela, and R. J. Wellinger, "A Phylogenetically Based Secondary Structure for the Yeast Telomerase RNA.", *Current Biology*, 14: 1148-1158, 2004.
6. S. Eddy and R. Durbin, "RNA sequence analysis using covariance models.", *Nucleic Acids Research*, 22: 2079-2088, 1994.
7. D. N. Frank and N. R. Pace, "Ribonuclease P: unity and diversity in a tRNA processing ribozyme.", *Annu Rev Biochem.*, 67: 153-180, 1998.
8. D. Gautheret and A. Lambert, "Direct RNA motif definition and identification from multiple sequence alignments using secondary structure profiles.", *Journal of Molecular Biology*, 313: 1003-1011, 2001.
9. F. Gavril, "Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph", *SIAM Journal on Computing*, 1:180-187, 1972.
10. F. Gavril, "The intersection graphs of subtrees in trees are exactly the chordal graphs", *Journal of Combinatorial Theory Series B*, 16: 47-56, 1974.
11. S. J. Geobel, B. Hsue, T. F. Dombrowski, and P. S. Masters, "Characterization of the RNA components of a Putative Molecular Switch in the 3' Untranslated Region of the Murine Coronavirus Genome.", *Journal of Virology*, 78: 669-682, 2004.
12. S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, and S. R. Eddy, "Rfam: an RNA family database.", *Nucleic Acids Research*, 31: 439-441, 2003.
13. R. J. Klein and S. R. Eddy, "RSEARCH: Finding Homologs of Single Structured RNA Sequences.", *BMC Bioinformatics*, 4:44, 2003.
14. A. Krogh, M. Brown, I.S. Mian, K. Sjolander, and D. Haussler, "Hidden Markov models in computational biology: Applications to protein modeling", *J. Molecular Biology*, 235: 1501-1531, 1994.
15. C. Liu, Y. Song, R. Malmberg, and L. Cai, "Profiling and Searching for RNA Pseudoknot Structures in Genomes.", *Lecture Notes in Computer Science*, 3515: 968-975.
16. T. M. Lowe and S. R. Eddy, "tRNAscan-SE: A Program for Improved Detection of Transfer RNA genes in Genomic Sequence.", *Nucleic Acids Research*, 25: 955-964, 1997.
17. J. Matousek and R. Thomas, "On the complexity of finding iso- and other morphisms for partial k-trees.", *Discrete Mathematics*, 108: 343-364, 1992.
18. N. Nameki, B. Felden, J. F. Atkins, R. F. Gesteland, H. Himeno, and A. Muto, "Functional and struc-

tural analysis of a pseudoknot upstream of the tag-encoded sequence in E. coli tmRNA.", *Journal of Molecular Biology*, 286(3): 733-744, 1999.

19. V. T. Nguyen, T. Kiss, A. A. Michels, and O. Bensaude, "7SK small nuclear RNA binds to and inhibits the activity of CDK9/cyclin T complexes.", *Nature* 414: 322-325, 2001.
20. E. Rivas and S. Eddy, "The language of RNA: a formal grammar that includes pseudoknots.", *Bioinformatics*, 16: 334-340, 2000.
21. E. Rivas and S. R. Eddy, "Noncoding RNA gene detection using comparative sequence analysis.", *BMC Bioinformatics*, 2:8, 2001.
22. E. Rivas, R. J. Klein, T. A. Jones, and S. R. Eddy, "Computational identification of noncoding RNAs in E. coli by comparative genomics.", *Current Biology*, 11: 1369-1373, 2001.
23. E. Rivas and S. R. Eddy, "A dynamic programming algorithm for RNA structure prediction including pseudoknots.", *Journal of Molecular Biology*, 285: 2053-2068, 1999.
24. N. Robertson and P. D. Seymour, "Graph Minors II. Algorithmic aspects of tree-width.", *Journal of Algorithms*, 7: 309-322, 1986.
25. Y. Song, C. Liu, R. L. Malmberg, F. Pan, and L. Cai, "Tree decomposition based fast search of RNA structures including pseudoknots in genomes", *Proceedings of the 2005 IEEE Computational Systems Bioinformatics Conference*, 223-224, 2005.
26. Y. Uemura, A. Hasegawa, Y. Kobayashi, and T. Yokomori, "Tree adjoining grammars for RNA structure prediction.", *Theoretical Computer Science*, 210: 277-303, 1999.
27. Z. Weinberg and W. L. Ruzzo, "Faster genome annotation of non-coding RNA families without loss of accuracy.", *Proceedings of the Eighth Annual International Conference on Computational Molecular Biology*, 243-251, 2004.
28. Z. Yang, Q. Zhu, K. Luo, and Q. Zhou, "The 7SK small nuclear RNA inhibits the Cdk9/cyclin T1 kinase to control transcription.", *Nature* 414: 317-322, 2001.