

Acceleration of Covariance Models for Non-coding RNA Search

Scott F. Smith

Department of Electrical and Computer Engineering
Boise State University
Boise, Idaho 83725-2075 USA
sfsmith@boisestate.edu

Abstract-Stochastic context-free grammar (SCFG) based models for non-coding RNA (ncRNA) gene searches are much more powerful than regular grammar based models due to the ability to model intermolecular base pairing. The SCFG models (also known as covariance models) can be scored exactly using dynamic programming techniques. However, the computational resources needed to compute optimal scores using dynamic programming is too great for most applications. Pre-filtering of the database using regular grammar based models can lead to significant improvements in performance at little or no cost in terms of specificity or sensitivity. While pre-filtering is a major improvement, the algorithm is still way to slow. The use of an alternative search strategy for high scoring subsequences in the sequence database is explored in this paper. Rather than sequentially computing the best score at each database position and subsequence length as is done in the dynamic programming method, good suboptimal scores are found throughout the position and length search space and the search is expanded about these trial solutions.

I. INTRODUCTION

A context-free grammar (CFG) as described by Chomsky [1] is powerful enough to describe base pairing of the nucleotides in a single-stranded RNA molecule. This is due to the CFG's ability to add symbols to both ends of a string simultaneously, whereas a regular grammar builds up a string only from left to right (or right to left). The regular grammar does not allow long range interactions between non-adjacent symbols. When searching for non-coding RNA (ncRNA) genes in DNA sequence databases, the usual homology search algorithms based on regular grammars are not powerful enough due to low conservation of individual sequence positions. However, ncRNA secondary structure (intermolecular base pairing patterns) are much more highly conserved. As a result, the stochastic context-free grammar (SCFG) based covariance model (CM) is much more effective for ncRNA gene search than regular grammar based methods.

The regular grammar based methods include BLAST [2], FASTA [3], and Smith-Waterman [4] for searching a single sequence against a database. They also include hidden Markov model (HMM) [5] searches when a family of sequences is known and used jointly for database search. The HMM is considerably more powerful than the single sequence methods since position-specific match scoring and insertion and deletion probabilities can be implemented. The covariance model is also built from a family of known member sequences and can be viewed as an extension of the HMM to include base pairing probabilities. Whereas the structure of the HMM is very regular, the covariance model structure varies with the secondary structure of the ncRNA family modeled. This makes analysis of the models much more difficult.

The covariance model does a very good job of database search in terms of sensitivity (the ability to identify true family members) and specificity (the ability to reject false family members), but the algorithm is extremely slow. It is so slow that it is generally not used by itself, but instead on the output of a database pre-filtering algorithm. A common pre-filter is a standard HMM constructed from the same training sequences as that of the covariance model. The threshold of the HMM is set much lower than it would normally be for protein or coding gene database search and the hope is that there is enough primary sequence conservation in the ncRNA family to find regions of higher hit probability in the database sequences. This is the method used by the "Rfam" database of ncRNA families and related covariance models [6]. Recent research has determined a way of constructing the HMM and setting the threshold such that it is guaranteed that the pre-filter output will not reject any subsequence that the covariance model will subsequently find [7]. While this is a major advance, the output of this lossless pre-filter is still too large for reasonable execution times of the covariance model for many ncRNA families.

In recent years the number of RNA families known to have catalytic and regulatory functions without being translated into proteins has increased dramatically [8]. The Rfam database currently contains about 500 of these families. The functions include those that have

been known for decades such as transfer RNA (tRNA) used to transport single amino acids and ribosomal RNA (rRNA) which performs an essential catalytic function in translating messenger RNA (mRNA) to protein in the ribosome. More recently discovered ncRNA families include microRNA [9], small nucleolar RNA (snoRNA) [10], and the RNA subunit of telomerase [11].

The structure of a covariance model forms a binary tree of nodes. Each node contains between one and six internal states and each of these states is evaluated for every possible starting position in the database sequence and for every possible subsequence length extending from that starting position. In order to make the problem tractable, the subsequence lengths searched are truncated at an upper limit D that is at least as long as the longest known member of the ncRNA family and may be as much as twice as long. The D value is chosen using expert opinion and represents a disadvantage of the usual dynamic programming scoring method since it causes rejection of true positives that might be longer than D . The slowness of the dynamic programming scoring method is largely a result of needing to evaluate the CM tree for all starting positions j in the database sequence and for all extensions of length d between 1 and D . The method proposed here does not enumerate all j and d and does not set a hard upper limit D .

By examination of families in the Rfam database, it has been found that large portions of the d values in the search space are extremely rare. We can use this fact to focus our search in the high probability regions of the search space while taking sparser samples in the improbable regions. The search then expands about those values of j and d (and insert/delete patterns within those search points) that yield good scores.

The paper is organized as follows. A short introduction to covariance models is given in Section II. Section III investigates the subsequence length usage of ncRNA families. The representation of local alignments of the CM to the database sequence and the method for searching the space of j , d , and local alignments is presented in Section IV. Concluding remarks appear in Section V.

II. BACKGROUND ON COVARIANCE MODELS

A covariance model may be estimated from a group of nucleotide sequences known to belong to an ncRNA family along with a structure-annotated multiple alignment of those sequences. The structural information may have been found from experimental evidence or by computational methods such as the Zucker algorithm [12]. If the structure contains pseudoknots (non-nested base pairing), then the CFG of the covariance model is not able to accurately describe the structure. However, ignoring the base pairing

information of part of the pseudoknots and forcing a nested structure seems to work adequately for most families. A much more detailed discussion of covariance models may be found in [13].

A. Consensus Sequence and Structure Lead to CM Nodes

The CM tree is composed of P, L, and R emitting nodes and S, B, and E non-emitting nodes. The emitting nodes are associated with consensus columns of the multiple alignment. Figure 1 shows 7 of the 15 sequences and the first 17 columns of the structure-annotated multiple alignment of the RyeB ncRNA family from Rfam. Fifteen of the columns are consensus columns (those where the majority of sequences show a nucleotide). The two non-consensus columns contain a "." as the consensus structure and consensus sequence entry. The fifteen consensus columns are associated with a symbol-emitting P, L, or R node in the CM and the two non-consensus columns are not associated with any node. Columns that are annotated as base paired have a "<" or ">" symbol as the consensus structure entry. The "<" indicates that it is the nucleotide of the pair closer to the 5' end of the RNA molecule and ">" that it is closer to the 3' end. Since pseudoknots are not allowed, there is no ambiguity of which column is associated with which. Column 4 pairs with column 13, 3 with 14, and 2 with 15. Each of these three pairs of columns is associated with a single P (pair-emitting) node. Columns with consensus structure marked "-" do not base pair and are associated with an L (left emitting) or R (right emitting) node. A P node emits one symbol on each end of the substructure of the tree nodes below it. An L node emits a single symbol to the left (5' end) of the substructure of the nodes below it and an R node emits to the right. There are often situations where either an L or an R node could be used and the L node is always used in these situations by convention.

AP002559	AGGC.AACUA.AGCCUG
AE016761	AGGC.AACUA.AGCCUG
BX950851	GGGC.UAGUACAGCUUG
AE008783	AGGC.GAUUU.AGCCUG
AL627272	AGGC.GAUUU.AGCCUG
AE013855	CGGCUGAAUA.AGCCUA

<i>Consensus:</i>	
<i>Structure</i>	--<<<.-----.->>>--
<i>Sequence</i>	aGGC.gAnUa.AGCcUg

Fig. 1. A multiple alignment annotated with structure.

The structure of Figure 1 only has a single stem, so the binary tree for a model of the multiple alignment shown in the figure does not branch. If there were

multiple stems, then they are separated by bifurcations. The non-emitting start (S), bifurcation (B), and end (E) nodes are used to give a tree structure to the emitting P, L, and R nodes. The node at the root of the tree is always an S node (called the root start node) and given a node index of 0. Branching is accomplished by the B nodes which all have exactly two children. The children of B nodes are always both S nodes. The S nodes have slightly different internal structure depending on whether they are the root S node, a left child S node, or a right child S node. The E nodes appear at the ends of the branches. Evaluation of the CM tree starts at the E nodes and proceeds up the tree to the root S node.

The CM "tree" associated with the multiple alignment in Figure 1 is shown in Figure 2. The emitting nodes are labeled with the symbol or symbol pair that is emitted with highest probability. Each L and R node has four probabilities associated with emitting A, C, G, or U respectively. P nodes have sixteen probabilities associated with each of the possible pairs of the four nucleotide symbols. Notice that evaluating the score of a query with respect to the model starts somewhere in the middle of the query sequence and works out towards both ends. The L node closest to the E node is associated with column 12 of the multiple alignment and the L and R nodes near the root S node are the first column and last two columns of the multiple alignment respectively.

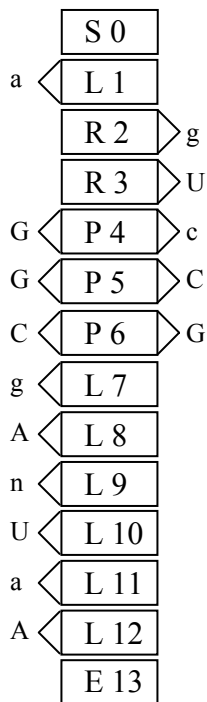


Fig. 2. Covariance model tree associated with structure-annotated multiple alignment.

B. Internal State Structure of Nodes

It is necessary to allow a database sequence to have symbols that are not part of the consensus (an insert) or not have symbols that are in the consensus (a delete). For example, sequence AE013855 in Figure 1 has an insert with respect to the consensus in the fifth column. If we tried to fit this sequence to the model so far, it would score poorly since either the columns to the right or to the left of this insert would be displaced.

The ability to handle inserts and deletes comes from the internal state structure of the nodes. The nodes are internally divided into two tiers, the upper or first tier is associated with matching or deleting a consensus symbol or pair of symbols and the lower or second tier is associated with inserting extra symbols between the emitted consensus symbols and the structure of the tree below the node. L nodes have three internal states, one to match the consensus symbol, one to delete the consensus symbol, and one to add extra symbols between the consensus symbol and the subsequence being build upon to the right. R nodes also have three internal states, one to match, one to delete, and one to add extra symbols between the consensus and the subsequence being build upon to the left. P nodes have six internal states, one to match both consensus symbols, one to match only the left consensus symbol, one to match only the right consensus symbol, one to delete both consensus symbols, and one each to insert symbols between the emitted consensus on either side. Some S nodes also have insert (but not delete) states to handle insertions on the two ends and between subsequences being joined by a bifurcation.

The internal structure of the most complicated type of node (the P node) is shown in Figure 3. The match pair (MP) state is visited if both symbols of the pair are matched. Match right (MR) and match left (ML) are visited if only the right or left half of the pair respectively are matched and the other is deleted. The D state is used if both of the consensus pair symbols are omitted. The IL state inserts additional symbols between the left symbol of the consensus pair and the subsequence from the tree below. The IR state inserts additional symbols between the right symbol of the consensus pair and the subsequence being built upon. The self loops on the insert states allow more than one insert.

C. Scoring a CM Using Dynamic Programming

The standard dynamic programming method for database search is to use an outer loop over database sequence start position j , and an inner loop over subsequence length d . Every state in every node is evaluated for the score of fitting the database subsequence given by j and d to the sub-model represented by the state and all the states in the tree below. The score of a state depends on the scores of all

it's child states, so the model is evaluated from the E states (in the E nodes) upward to the root S state (in the root S node). The score of the root S state is the score of the overall CM. This amounts to using dynamic programming to do a pair-wise alignment between the consensus sequence and the database subsequence indicated by j and d . This alignment is repeated for all locations j and length extensions d between 1 and D . The complexity of the computation is proportional to the database sequence length L , the subsequence upper bound D , and the number of states in the model.

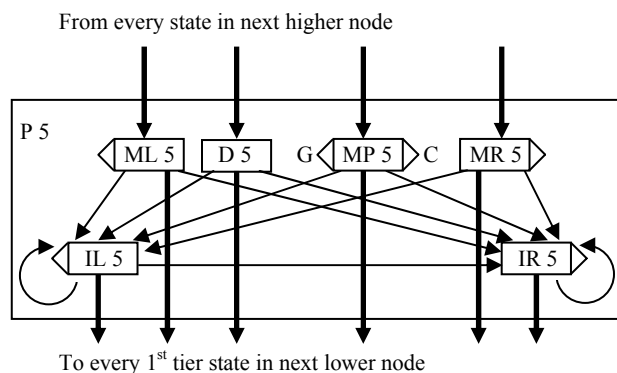


Fig. 3. Internal states of a P-type node.

III. SUBSEQUENCE LENGTH USAGE

Investigation of the members of Rfam families shows that it is highly unusual for the best fit of a true family member to the CM to deviate in length very much from the consensus length represented at any of the model states. This indicates that the range of highly probable d values at a given state is quite narrow. The standard solution method searches all possible d values up to D (and no d values over D) in every state at every possible starting point j . If one does not have the computing resources to do this, it would be best to expend computing resources where the likelihood of success is highest. One way to do this is to choose initial test points for d with highest probability in the vicinity of d values observed most often in practice.

As an example, the RyeB ncRNA family from the Rfam database (accession number RF00111) [14-16] is presented. Examination of many other families in Rfam indicates that the conclusions drawn from this example are applicable generally. This family has fifteen known members of average length 100. A condensed version of the CM tree is shown in Figure 4. In the figure all S, B, and E nodes have been omitted and adjacent nodes of the same type have been condensed into a single node. The numbers inside the condensed nodes refer to the

original node indices from the CM file obtained from Rfam.

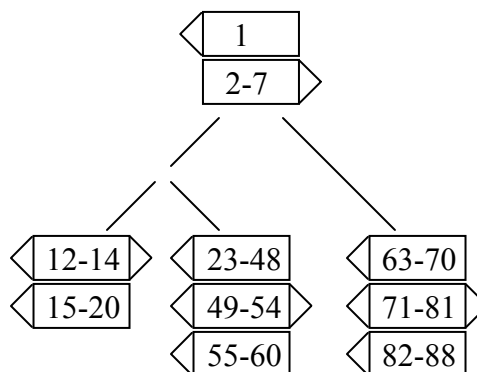


Fig. 4. Organization of the RyeB (RF00111) model.

TABLE I
SUBSEQUENCE LENGTH DEVIATIONS FOR RYEB FAMILY

Nodes	-2	-1	0	+1	+2
88-81			15		
80-63			12	3	
60-58			15		
57		1	14		
56-24	1		14		
23	1	1	13		
20			15		
19-15			14	1	
14-12			11	14	
7-1		2	10		3

All fifteen true family members were fit to the states of the CM and the length of the subsequence fit at each state and the length implied by the consensus sequence of the sub-tree compared. The actual minus the implied length is recorded as the length deviation in Table I. The deviations are listed by the index of the node that contains the state and groups of nodes with the same pattern listed together. The first row of the table shows that for the eight L nodes closest to the E node of the right tree branch, no insertions or deletions were needed to fit any of the fifteen sequences optimally. Since the value of D used by Rfam for this family is 150, the standard method of score evaluation would search over deviations from -1 to +149 for node L88, -2 to +148 for node L87, -3 to +147 for node L86, etc. Of the 150 search points, 149 never happened in practice. Over the entire model, no state ever had a deviation of more than two in either direction. This tendency for deviations to remain within a very small fraction of D is in fact quite general.

U51991: AGGCAACUAAGCCGCAUUAUGCC
 BX950851: GGGCUAGUACAGCUUGUAUAAAUGCC
 RyeB seq: AGGCGACUAAGCCUGCAUUAUGCC

AACUUUUAGCGCACGGCUCUCUCCCAAGAGCCAUUUCC
 . . AACUUUUAGCGCACGGCUCUCUCCCAAGAGCCAUUUCC
 AACUUUUAGCGCACGGCUCUCUCCCAAGAGCCAUUUCC

CUGGACCGAAUACAGGAAUCGUGUUCGGUCUCUUUUU
CUAGACUGAAUACAGGAAUCGUAUUCAGUCUUUUUUU
 CUGGACCGAAUACAGGAAUCGUAUUCGGUCUCUUUUUU

Fig. 7. Best single-position gap alignments of U51991 and BX950851 against RyeB consensus sequence.

To try the opposite of pushing portions of the consensus apart, runs of 0 can be introduced to pull portions together. These runs of 0 might also be inserted around the one quarter, one half, and three quarters points in the alignment vector up to a maximum run length R . Figure 7 shows the best alignments possible for the same two sequences as in Figure 6, but with a single run of inserts or deletes allowed at the position one fourth of the way from the start of the consensus sequence. The insertion/deletion point after the 25th consensus symbol is just to the right of the end of the first row of data in Figure 7. The BX950851 sequence score is improved by placing two deletes in the consensus sequence starting at this position. This mirrors the two deletes that occur in the optimal alignment of Figure 5 after the 46th consensus sequence position. The U51991 score can not be improved with either inserts or deletes after the 25th consensus sequence position.

After these expansions about the test points the test point population can be trimmed once again. Perhaps take the best ten of the altered plus original alignment vectors at each position j . Find the best score among the ten retained solutions at each position and use that as the current position score. Then keep only the best fifth of the positions j among the current position scores. Further rounds of expansion about the test points and culling out the best can then be performed.

V. CONCLUSIONS

A method has been presented to search for ncRNA genes in DNA sequence databases when fewer computing resources are available than those required to do a traditional dynamic programming search with a covariance model. The method takes into account the observation that the majority of the search space traversed by the dynamic programming algorithm is not even close to almost all ncRNA genes observed in practice.

Much work still remains to be done in fine tuning a number of free parameters in the search method. It is not known what the best choices are for number of points along the alignment vector to use for insertion and deletion tests and how to optimally choose the number of inserts and deletes at those points. It is also not known what the optimal fraction of test points to retain on each cycle is.

ACKNOWLEDGMENT

The project described was supported by NIH Grant Number P20 RR016454 from the INBRE Program of the National Center for Research Resources.

REFERENCES

- [1] N. Chomsky, "On Certain Formal Properties of Grammars," *Information and Control*, 2, pp. 137-167, 1959.
- [2] S. Altschul, W. Gish, W. Miller, E. Myers, and D. Lipman, "Basic Local Alignment Search Tool," *Journal of Molecular Biology*, 215, pp. 403-410, 1990.
- [3] W. Pearson and D. Lipman, "Improved Tools for Biological Sequence Comparison," *Proceedings of the National Academy of Sciences*, 4, pp. 2444-2448, 1988.
- [4] T. Smith and M. Waterman, "Identification of Common Molecular Subsequences," *Journal of Molecular Biology*, 147, pp. 195-197, 1981.
- [5] S. Eddy, "Hidden Markov Models," *Current Opinion in Structural Biology*, 6, pp. 361-365, 1996.
- [6] S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, and S. Eddy, "Rfam: An RNA Family Database," *Nucleic Acids Research*, 31, pp. 439-441, 2003.
- [7] Z. Weinberg and W. Ruzzo, "Faster Genome Annotation of Non-coding RNA Families Without Loss of Accuracy," *Int. Conf. on Research in Computational Molecular Biology*, pp. 243-251, 2004.
- [8] R. Gesteland, T. Cech, and J. Atkins, *The RNA World*, 3rd Ed., Cold Spring Harbor Laboratory Press, 2005.
- [9] M. Lagos-Quintana, R. Rauhut, W. Lendeckel, and T. Tuschl, "Identification of Novel Genes Coding for Small Expressed RNAs," *Science*, 294, pp. 853-858, 2001.
- [10] T. Kiss, "Small Nucleolar RNAs: an Abundant Group of Noncoding RNAs with Diverse Cellular Functions," *Cell*, 109, pp. 145-148, 2002.
- [11] T. De Lange, V. Lundblad, and E. Blackburn, *Telomeres*, 2nd Ed., Cold Spring Harbor Laboratory Press, 2005.
- [12] M. Zucker, "Computer Prediction of RNA Structure," *Methods in Enzymology*, 180, pp. 262-288, 1989.
- [13] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis*, Cambridge University Press, 1998.
- [14] N. Lau, L. Lim, E. Weinstein, and D. Bartel, "An Abundant Class of Tiny RNAs with Probable Regulatory Roles in *Caenorhabditis elegans*," *Science*, 294, pp. 858-862, 2001.
- [15] L. Sempere, N. Sokol, E. Dubrovsky, E. Berger, and V. Ambros, "Temporal Regulation of microRNA Expression in *Drosophila melanogaster* Mediated by Hormonal Signals and Broad-complex Gene Activity," *Developmental Biology*, 259, pp. 9-18, 2003.
- [16] K. Wassarman, F. Repoila, C. Rosenow, G. Storz, and S. Gottesman, "Identification of Novel Small RNAs Using Comparative Genomics and Microarrays," *Genes and Development*, 15, pp. 1637-1651, 2001.
- [17] J. Yadgari, A. Amir, and R. Unger, "Genetic Threading," *Constraints* 6, pp. 271-292, 2001.