HMM WITH AUXILIARY MEMORY: A NEW TOOL FOR MODELING RNA SECONDARY STRUCTURES

Byung-Jun Yoon and P. P. Vaidyanathan

Dept. of Electrical Engineering
California Institute of Technology, Pasadena, CA 91125, USA
E-mail: bjyoon@caltech.edu, ppvnath@systems.caltech.edu

ABSTRACT

For a long time, proteins have been believed to perform most of the important functions in all cells. However, recent results in genomics have revealed that many RNAs that do not encode proteins play crucial roles in the cell machinery. The so-called ncRNA genes that are transcribed into RNAs but not translated into proteins, frequently conserve their secondary structures more than they conserve their primary sequences. Therefore, in order to identify ncRNA genes, we have to take the secondary structure of RNAs into consideration. Traditional approaches that are mainly based on base-composition statistics cannot be used for modeling and identifying such structures and models with more descriptive power are required. In this paper, we introduce the concept of context-sensitive HMMs, which is capable of describing pairwise interactions between distant symbols. It is demonstrated that the proposed model can efficiently model various RNA secondary structures that are frequently observed.

1. INTRODUCTION

It has been the central dogma of biology that genetic information flows from DNA to RNA to protein. RNA has been mainly viewed as a passive intermediary between DNA and protein, except for several infrastructural RNAs such as the tRNA (transfer RNA) and the rRNA (ribosomal RNA). Proteins have been believed to perform most of the crucial functions in all cells, and therefore, most of the research has been naturally focused on identifying protein-coding genes and their functions. The small portion of the genome that encodes proteins has been regarded as the only important part in the entire genome, and the vast majority that does not convey any information for encoding proteins has been thought to be useless remnants of genetic evolution.

However, during the last decade, a number of non-coding RNAs (ncRNAs) have been found that take part in various important processes in the cell machinery [1, 2, 3, 4]. For example, it has been found that ncRNAs affect transcription and the chromosome structure, participate in RNA processing and modification, regulate mRNA stability and translation, and also affect protein stability and transport [2]. In fact, the importance of ncRNAs has been underestimated for a long time, and it was only very recent that we realized that many crucial ncRNAs have evaded our detection for several decades. During the last few years, surprisingly many functions of ncRNAs have been discovered, but there are still countless ncRNAs whose functions are not known to us. Moreover, although numerous new ncRNAs have been found in laboratories, the ncRNAs that have been identified till now are still considered to be only a small fraction of the existing ncRNAs [4].

One interesting characteristic of many ncRNAs is that they conserve their secondary structures more than they conserve their primary sequences [5, 6]. Unlike protein-coding genes, ncRNA genes do not display strong base composition biases in most organisms [5]. Therefore, in order to identify ncRNA genes in a DNA sequence, we have to consider both the consensus secondary structure of the ncRNA gene and its primary sequence. Traditionally, hidden Markov models (HMMs) have been successfully used in computational identification of protein-coding genes [7, 8]. HMMs are well-known for their efficiency in modeling short-term dependencies between adjacent symbols. However, they mainly depend on sequence composition statistics, and they are not able to grasp longer-range interactions between symbols that are frequently observed in RNA secondary structures. Due to this reason, traditional models cannot be used for modeling RNA secondary structures and detecting ncRNA genes.

We need more complex models with greater descriptive power for this purpose. Until now, a number of methods have been proposed by several groups of researchers [9, 10] that are based on stochastic context-free grammars (SCFGs), a higher order relative of HMMs. In this paper, we introduce the concept of context-sensitive hidden Markov model that can effectively grasp pairwise interactions between distant symbols. The proposed model provides an efficient framework for modeling RNA secondary structures, and it has several advantages over SCFGs as will be demonstrated later.

2. TRANSFORMATIONAL GRAMMARS

According to the Chomsky hierarchy of transformational grammars [11], there are four classes of grammars as shown in Fig. 1. These include *regular grammars*, *context-free grammars*, *context-sensitive grammars* and *unrestricted grammars*, in the order of decreasing restrictions in the production rules. With less restrictions on the production rules, higher order grammars have more descriptive power, and therefore they are capable of describing complex interactions between symbols. On the other hand, although lower order grammars such as the regular grammars are more restricted and less powerful, they have advantages with respect to computational complexity, since they are easier to parse. According to the Chomsky hierarchy of transformational grammars, HMMs can be viewed as the stochastic version of regular grammars.

Two interesting examples of languages that cannot be described using regular grammars - or equivalently, by HMMs - are the palin-

Work supported in parts by the ONR grant N00014-99-1-1002 USA and the Microsoft Research fellowship.

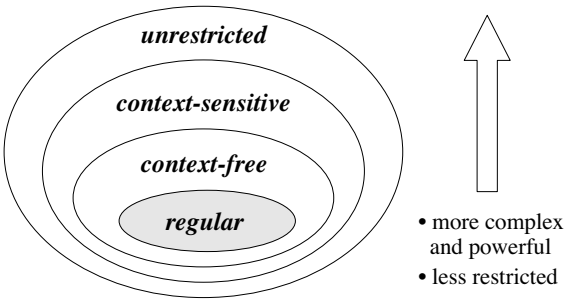

Fig. 1. The four classes of grammars in the Chomsky hierarchy of transformational grammars.

drome language and the copy language. The palindrome language is a language that contains all strings that read the same forwards and backwards. For example, if we consider a palindrome language that uses an alphabet of two letters $\{a, b\}$ for terminal symbols, it contains all symbol strings of the form aa, bb, $abba$, $aabbaa$, $abaaba$ and so forth. The copy language includes all sequences that consist of the concatenation of two identical sequences. For example, it contains all symbol strings that have the form aa, bb, $abab$, $abbabb$ and so on. Figure 2 shows examples of symbol strings that are included in these languages. The lines in the figure indicate the pairwise interactions between symbols that are distant from each other. This kind of longer range correlations cannot be described using regular grammars. It is of course possible that a regular grammar generates palindromes as part of its language. However, a regular grammar is not capable of generating *only such palindromes*, hence it cannot effectively discriminate palindromes from non-palindromic sequences. In fact, in order to describe a palindrome language, we have to use a higher-order grammar such as the context-free grammars. Context-free grammars are capable of modeling nested dependencies between symbols, as shown in Fig. 2 (a). Although the copy language does not appear any more complex than the palindrome language, we need context-sensitive grammars to represent such a language. This is due to the crossing interactions as shown in Fig. 2 (b), which cannot be modeled using context-free grammars.

As mentioned earlier, one important application of stochastic context-free grammars (SCFGs) [12] is the RNA secondary structure analysis. Many interesting ncRNAs conserve the secondary structure, which makes their primary sequences look like palindromes or concatenations of several palindromes. Since HMMs cannot be used for modeling palindromic sequences, we have to

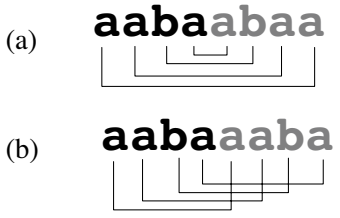

Fig. 2. Examples of symbols strings in (a) the palindrome language and (b) the copy language. The lines show the pairwise correlations between distant symbols.

resort to a more complex model such as the SCFG. Until now, several SCFG-based methods have been introduced for computational analysis of RNA sequences [9, 10]. It has been shown that the SCFG-based approach can identify ncRNAs with high accuracy [5, 13]. One major disadvantage of the SCFG-based methods is their high computational complexity, which makes the prediction of large ncRNA genes in long DNA sequences infeasible.

Instead of using SCFGs, we can alternatively use the context-sensitive HMM that is proposed in this paper. Unlike traditional HMMs, context-sensitive HMMs are capable of modeling pairwise interactions between symbols that are distant from each other. The proposed model has the advantage that the states in the model directly correspond to the base locations in an RNA sequence[1]. In addition to this, since the proposed model is an extension of the HMM, it can be easily incorporated into existing gene-finders that are built on HMMs. Moreover, the context-sensitive HMM has efficient algorithms for finding the optimal state sequence (the *alignment* problem) [15] and computing the probability (the *scoring* problem) [16] of a given observation sequence. For sequences with a single nested structure, the computational complexity of these algorithms is $O(L^2M^3)$, where L denotes the length of the sequence and M denotes the number of different states (or non-terminals in SCFGs). This is smaller than the complexity $O(L^3M^3)$ of the alignment and scoring algorithms for general SCFGs [12].

3. CONTEXT-SENSITIVE HIDDEN MARKOV MODELS

The context-sensitive HMM can be viewed as an extension of the traditional HMM, where some of the states are equipped with auxiliary memory. Symbols that are emitted at certain states are stored in the memory, and the stored data serves as the context which affects the emission probabilities and the transition probabilities of the model. There are three different kinds of states, namely, the *single-emission state* S_n, the *pairwise-emission state* P_n, and the *context-sensitive state* C_n. The states P_n and C_n always exist in pairs. For example, consider the case when there are two pairwise-emission states P_1 and P_2 in the model. Then the HMM is required to have also two single-emission states C_1 and C_2. Each pair (P_n, C_n) is associated with a separate memory element, such as a stack or a queue. The triplet of P_n, C_n and the n-th memory element is shown in Fig. 3.

The differences between the three classes of states are as follows. The single-emission state S_n is identical to the regular state in traditional HMMs. It emits a symbol according to the associated emission probabilities, as we enter the state. After emitting a symbol, it makes a transition to the next state, according to the specified transition probabilities. The pairwise-emission state P_n is almost identical to the single-emission state S_n, except that the symbol that is emitted at P_n is stored in the auxiliary memory dedicated for P_n and C_n. The data stored in the memory affects the emission probabilities and the transition probabilities of C_n in the future. After storing the emitted symbol in the memory, a transition is made to the next state by following the transition probabilities that are associated with P_n. The context-sensitive state C_n is considerably different from the other states, in the sense that its emission probabilities and the transition probabilities are not fixed. In fact, these probabilities depend on the *context*, or the data

[1] In SCFGs, some of the non-terminals, which are the equivalent of states in HMMs, may not emit any symbol. These *abstract* non-terminals do not correspond to the base locations.

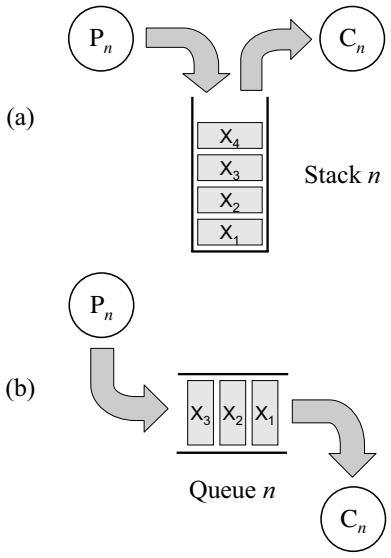

Fig. 3. Examples of the triplet P_n, C_n and the n-th memory element that is associated with these states. (a) When using a stack. (b) When using a queue.

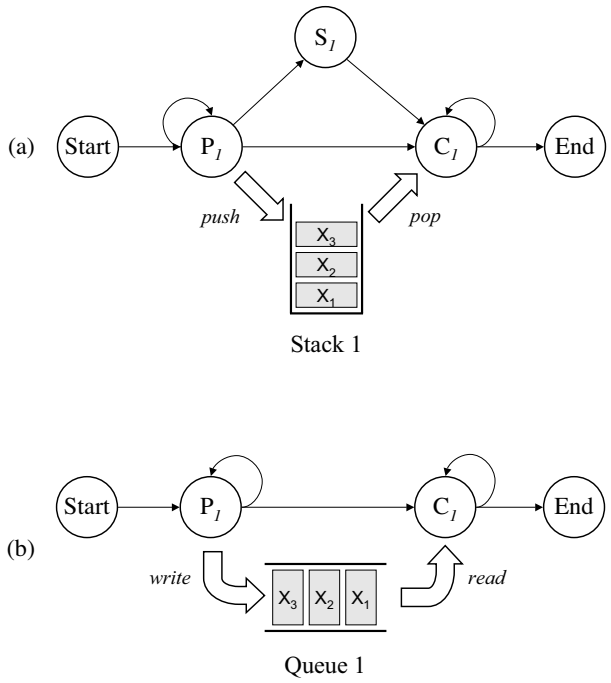

Fig. 4. (a) An example of a context-sensitive HMM that generates only palindromes. (b) An example that simulates a copy language.

stored in the associated memory element, which is the reason why C_n is called a context-sensitive state. When we enter C_n, it first accesses the n-th memory element and retrieves a symbol x. Once the symbol is retrieved, the emission probabilities of C_n are adjusted according to the value of x. For example, we may adjust the emission probabilities of C_n such that it emits the same symbol x with high probability (possibly, with probability one). When modeling a RNA secondary structure, we may adjust the emission probabilities so that C_n generates the complementary base of x. Another distinctive character of the context-sensitive state C_n is that the transition to the state C_n is not always allowed. For example, let us consider the case when some state attempts to make a transition to C_n. Before making the transition, the auxiliary memory that is associated with C_n is examined first. If the memory is empty, the transition to C_n is not allowed, and it is forced to make a transition to another state. This is done by setting the transition probability to C_n to zero and adjusting the remaining probabilities correspondingly, so that the probabilities add up to unity. This restriction is necessary to maintain the same number of P_n and C_n in a state sequence. Let $\mathbf{s} = s_1 s_2 \dots s_L$ be a feasible state sequence of an observed symbol string $\mathbf{x} = x_1 x_2 \dots x_L$. Then the number of occurrence of P_n in $\mathbf{s}$ is kept the same as the number of occurrence of C_n in $\mathbf{s}$ by the previous restriction. This is a reasonable restriction for the following reason. In the first place, if there are more C_n states than there are P_n states, the emission probabilities of the context-sensitive state C_n cannot be properly determined. On the other hand, if there are more P_n states than C_n states, the symbols that were emitted by surplus P_n states do not affect the probabilities in the model at all, hence they may be simply replaced by single-emission states.

By using the proposed context-sensitive HMM, we can easily construct a simple model that generates *only* palindromes. For example, we may use the structure shown in Fig. 4 (a). In this model, the n-th memory element that is associated with the pair (P_1, C_1) is a stack. Initially, the model begins at the pairwise-emission state P_1. It makes several self-transitions to generate a number of symbols, which are pushed onto the stack. At some point, it makes a transition to the context-sensitive state C_1. Once we enter the context-sensitive state C_1, the emission probabilities and the transition probabilities of C_1 are adjusted, such that the state always emits the symbol on the top of the stack and makes self-transitions until the stack becomes empty. In this way, C_1 emits the same symbols as were emitted by P_1, but in the reverse order. If we denote the number of symbols that were emitted by P_1 as N, the generated string will always be a palindrome of the form $x_1 \cdots x_N x_N \cdots x_1$ or $x_1 \cdots x_N x_{N+1} x_N \cdots x_1$. Similarly, we can also simulate a copy language by replacing the stack by a queue as illustrated in Fig. 4 (b). In this case, C_1 emits the same symbols as those emitted by P_1, but this time, in the same order since the queue is a first-in-first-out (FIFO) system. Consequently, the resulting string will always be a concatenation of two identical sequences which is of the form $x_1 \cdots x_N x_1 \cdots x_N$.

As the emission probabilities and the transition probabilities in the context-sensitive HMM are not fixed, and as they depend on the context of the system, algorithms that were used in traditional HMMs cannot be directly applied. Therefore, the Viterbi's algorithm [14] for finding the most probable state sequence and the forward algorithm [14] for computing the probability of an observation sequence cannot be used in this case. However, there exist efficient algorithms for finding the optimal alignment [15] and computing the probability of an observed symbol string [16] for context-sensitive HMMs. These algorithms can be used for models with a single nested structure, in which case they are faster than the algorithms[2] for general SCFGs [12].

[2]CYK algorithm can be used for finding the optimal alignment and the inside algorithm can be used for computing the probability of an observed sequence (the scoring problem).

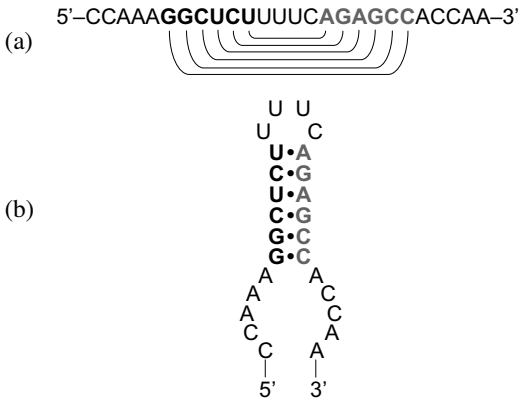

Fig. 5. The 3' end of a histone mRNA. (a) Primary sequence before folding. The lines indicate interactions between base-pairs. (b) Consensus secondary structure.

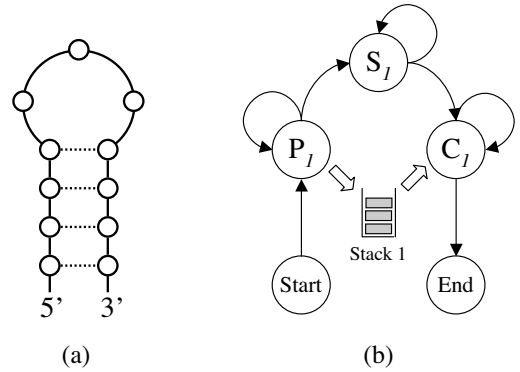

Fig. 6. (a) A typical stem-loop. The dotted lines indicate the interactions between bases that form complementary base-pairs. (b) An example of a context-sensitive HMM that generates stem-loops.

4. MODELING RNA SECONDARY STRUCTURES USING CONTEXT-SENSITIVE HMMS

Now that we have introduced the basic concept of context-sensitive HMMs, let us consider how they can be used in modeling RNA secondary structures. RNA is a nucleic acid that consists of a sequence of nucleotides A, C, G and U, where U is chemically similar to T in DNA. The nucleotide A forms a hydrogen bonded pair with U, and C forms a pair with G, which are called *complementary base-pairs* or *Watson-Crick base pairs*. RNA is generally a single-stranded molecule, and it typically folds onto itself to form consecutive base-pairs that are stacked onto each other, which is called a *stem*. The structure that results from these base-pairs is called the *RNA secondary structure*. An example of a simple RNA secondary structure is shown in Fig. 5, which illustrates the consensus secondary structure of the 3' end of the histone mRNA [17]. This kind of structure is called a *stem-loop* or a *hairpin*, and it is frequently observed in various RNAs. As can be seen in this example, there are pairwise interactions between bases that are distant from each other. Most of the pairwise interactions in RNAs occur in a nested fashion, where the interactions do not cross each other. However, some RNAs have also non-nested base pairs, which are called *pseudoknots*.

Context-sensitive HMMs are capable of modeling various kinds of RNA secondary structures. Given a consensus secondary structure, designing a model that generates sequences with the specified structure is relatively easy. For example, a typical stem-loop, or hairpin structure, that is illustrated in Fig. 6 (a) can be represented using the model in Fig. 6 (b). In this model, the pairwise-emission state P_1 and the context-sensitive state C_1 are associated with a stack, and they generate the stem part of the structure. The single-emission state S_1 is used for generating the loop, since the bases in the loop do not form pairs. We can also model a bulge, which is defined as non-paired bases inside a stem, by adding additional states to the model. More complex structures with multiple stem-loops can be represented using multiple state pairs (P_n, C_n) with separate stacks. Figure 7 (a) shows the typical secondary structure of a tRNA (transfer RNA). The tRNA is a short RNA molecule that usually consists of 74∼93 nucleotides. It transfers a specific amino acid to a growing polypeptide chain during the *translation* procedure of mRNA into protein [18]. The tRNAs have a highly con-

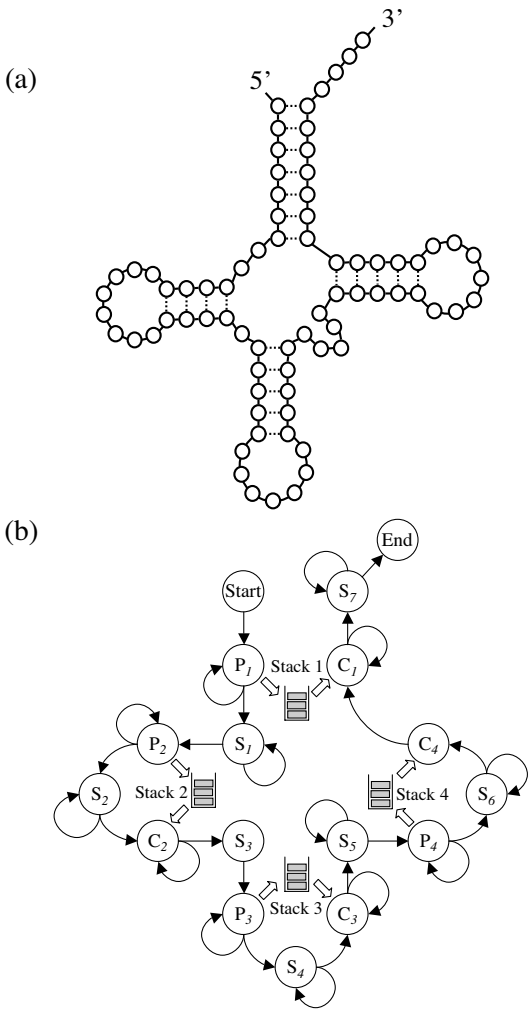

Fig. 7. (a) A typical tRNA cloverleaf structure. (b) An example of a context-sensitive HMM that can generate the cloverleaf structure.

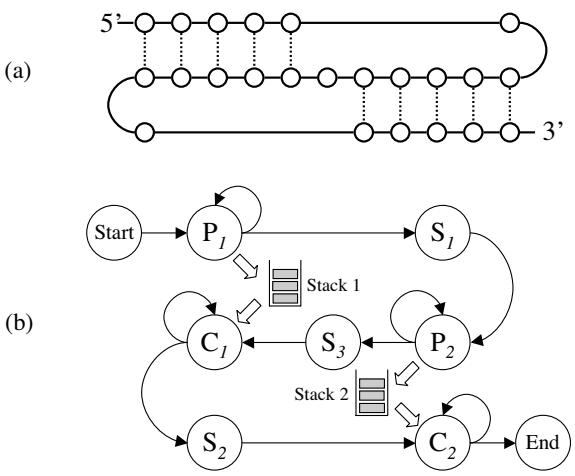

Fig. 8. (a) An example of a pseudoknot. (b) A context-sensitive HMM that can generate the pseudoknot.

served secondary structure with three stem-loops, which is called the *cloverleaf* structure due to its shape. As shown in Fig. 7, the cloverleaf structure can be modeled using four pairs of (P_n, C_n), where a separate stack is dedicated to each pair. Note the similarity between the original consensus RNA structure and the constructed context-sensitive HMM. As every state in the HMM corresponds to one or more base locations in the RNA sequence, the design procedure of context-sensitive HMMs is very intuitive. Figure 8 (a) illustrates an example of a pseudoknot structure. Note that there are several pairwise interactions that cross each other. As mentioned earlier, crossing interactions cannot be generated by SCFGs. However, as we can see in Fig. 8 (b), context-sensitive HMMs are capable of representing such dependencies, hence they can also be used for modeling pseudoknots. This example clearly shows that the context-sensitive HMMs have greater descriptive power than the stochastic context-free grammars.

5. CONCLUDING REMARKS

In this paper, we have introduced the concept of context-sensitive HMM. The context-sensitive HMM can be viewed as an extension of the traditional HMM, where some of the states are equipped with auxiliary memory elements. Symbols that are emitted at certain states are stored in the memory, and they serve as the context of the system which affects the probabilities of the model. In this way, we can represent longer range interactions between symbols that are distant from each other, which is not possible for traditional HMMs. The proposed model has an important application in computational analysis of RNA sequences. They can efficiently model RNA secondary structures, and they can be used in building gene-finders that predict ncRNA genes in unannotated DNA sequences. In fact, the proposed model has been used in predicting the secondary structure of the 3' end of histone mRNAs, where it has achieved a high sensitivity and a high positive predictive value, both of which were over 0.99 (the theoretical maximum is unity) [19]. Future research includes application of the context-sensitive HMMs for predicting ncRNA genes with more complex secondary structures, and developing algorithms for the alignment and scoring of sequences with pseudoknots.

6. REFERENCES

[1] J. S. Mattick, "Challenging the dogma: the hidden layer of non-protein-coding RNAs in complex organisms", *BioEssays*, vol. 25, pp. 930-939, 2003.

[2] S. Gisela, "An expanding universe of noncoding RNAs", *Science*, vol. 296, pp. 1260-1263, 2002.

[3] S. R. Eddy, "Non-coding RNA genes and the modern RNA world", *Nature Reviews Genetics*, vol. 2, pp. 919-929, 2001.

[4] G. Ruvkun, "Glimpses of a tiny RNA world", *Science*, vol. 294, pp. 797-799, 2001.

[5] S. R. Eddy, "Computational genomics of noncoding RNA genes", *Cell*, vol. 109, pp. 137-40, 2002.

[6] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological sequence analysis*, Cambridge Univ. Press, Cambridge, UK, 1998.

[7] A. Krogh, I. Saira Mian, D. Haussler, "A hidden Markov model that finds genes in E. coli DNA", *Nucleic Acids Res.*, vol. 22, pp. 4768-4778, 1994.

[8] S. L. Salzberg, A. L. Delcher, S. Kasif, O. White, "Microbial gene identification using interpolated Markov models", *Nucleic Acids Res.*, vol. 26, pp. 544-548, 1998.

[9] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjölander, R. C. Underwood and D. Haussler, "Stochastic context-free grammars for tRNA modeling", *Nucleic Acids Res.*, vol. 22, pp. 5112-5120, 1994.

[10] S. R. Eddy and R. Durbin, "RNA sequence analysis using covariance models", *Nucleic Acids Research*, vol. 22, pp. 2079-2088, 1994.

[11] N. Chomsky, "On certain formal properties of grammars", *Information and Control*, vol. 2, pp. 137-167, 1959.

[12] K. Lari and S. J. Young, "The estimation of stochastic context-free grammars using the inside-outside algorithm", *Computer Speech and Language*, vol. 4, pp. 35-56, 1990.

[13] T. M. Lowe and S. R. Eddy, "tRNAscan-SE: a program for improved detection of transfer RNA genes in genomic sequence", *Nucleic Acids Res.*, vol. 25, pp. 955-964, 1997.

[14] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", Proceedings of the IEEE 77 (1989) 257-286.

[15] Byung-Jun Yoon and P. P. Vaidyanathan, "Optimal alignment algorithm for context-sensitive hidden Markov models", in submission.

[16] Byung-Jun Yoon and P. P. Vaidyanathan, "Dynamic programming algorithm for scoring context-sensitive HMMs", in submission.

[17] Z. Dominski and W. F. Marzluff, "Formation of the 3' end of histone mRNA", *Gene*, vol. 239, pp. 1-14, 1999.

[18] B. Alberts, D. Bray, A. Johnson, J. Lewis, M. Raff, K. Roberts and P. Walter, *Essential cell biology*, Garland Publishing Inc., New York, 1998.

[19] Byung-Jun Yoon and P. P. Vaidyanathan, "RNA secondary structure prediction using context-sensitive hidden Markov models", to appear in Proc. International Workshop on Biomedical Circuits and Systems (BioCAS), Singapore, Dec. 2004.