U S I G RNA S P W P

Uwe Schöning
Universität Ulm
uwe.schoening@uni-ulm.de

Monika von Knop
von-Knop@t-online.de

Abstract

We suggest to use stochastic indexed grammars to predict RNA structure which includes pseudoknots.

A *stochastic context-free grammar* is a context-free grammar $G = (V, \Sigma, P, S)$, augmented with an assignment of probability values to each production rule,

$$p : P \to [0, 1]$$

Let $p[A \to \alpha]$ denote the probability assigned to the context-free production rule $A \to \alpha$. These probabilities have to satisfy for each non-terminal (or variable) $A \in V$ the condition

$$\sum_{(A\to\alpha)\in P} p[A \to \alpha] = 1$$

Let $D = (S \Rightarrow w_1 \Rightarrow \cdots \Rightarrow w_n), w_n = x$, be a left-derivation of terminal string $x \in L(G) \subseteq \Sigma^*$, i.e. in each derivation step, the leftmost variable has been replaced according to some grammar rule. Let $p_1, p_2, \ldots, p_n$ be the sequence of probability values associated with these rules. Then $\prod_{i=1}^{n} p_i$ is the probability associated with this particular derivation D, denoted by $p[D]$.

Finally, the probability of a string $x \in \Sigma^*$ is defined as

$$p[x] = \sum_{D \text{ is a left-derivation of } x} p[D]$$

Obviously, $p[x] = 0$ if $x \notin L(G)$.

Bulletin of the EATCS no 101, pp. 185–188, June 2010

It is possible to design a dynamic programming style algorithm of complexity $O(n^3)$ like the Cocke-Younger-Kasami algorithm, which, given $x \in \Sigma^*$, $|x| = n$, decides whether $x \in L(G)$, and furthermore finds the derivation of x which has the highest associated probability (a "Viterbi-like" algorithm). Such an algorithm can be used to predict the most likely RNA secondary folding structure. In this application, $x \in \{a, u, g, c\}^*$ is a given RNA sequence where a, u, g, c stand for the 4 nucleotids, *adenine, uracil, guanine,* and *cytosine*. We sketch a possible context-free grammar which is able to derive RNA strings, together with a suggested folding structure [7, 8, 9].

$$\begin{array}{lcl} S & \rightarrow & L \mid SS \mid cSg \mid gSc \mid aSu \mid uSa \\ L & \rightarrow & SL \mid a \mid u \mid c \mid g \mid aL \mid uL \mid cL \mid gL \end{array}$$

This is somewhat oversimplified, but the idea is that an RNA string can be formed, either by a stem (S) which means that matching Watson-Crick pairs *a-u*, *u-a*, *c-g*, *g-c* are aligned, or a sequence of several stems (SS). A stem can end in a loop (L) of arbitrary length where no particular order or alignment of the nucleotids is necessary, and within a loop, a new stem can start. Such a grammar can be easily augmented with additional features, for example, allowing also "wobble pairs" (*g-u*, *u-g*). Further, it is possible to distinguish between loops, hairpins, bulges, and multi-loops, by introducing new variables, and enforce that stems and hairpins should have a certain minimum length.

Such a context-free grammar can only model the secondary RNA structure, it is not possible to model "pseudoknots" which are non-contextfree. This is the reason why the cloverleaf structure of t-RNA is usually not predicted by RNA folding algorithms which are based on such context-free modeling concepts only. The t-RNA structure comes about because of the 3-dimensional winding of the RNA string against itself, and additional bindings of nucleotids which do not obey the context-freeness property.

There have been several suggestions in the literature how to augment context-free grammars by non-contextfree concepts to be able to predict pseudoknots, too (cf. [2]). Here, it seems reasonable to restrict oneself to some subset of pseudoknots, not allowing arbitrary nestings, because otherwise the pseudoknot recognition (and prediction) problem becomes NP-complete [6]. We use indexed grammars, originally introduced by Aho [1], cf. [10], by complexity reasons restricted to be linear (cf. [4]), and augmented with rule probabilities, i.e. stochastic linear indexed grammars.

An *indexed grammar* (originally defined in [1], but simplified here for our purposes) is given by pairwise disjoint alphabets V (variables), T (terminals), and F (indices or flags), and grammar rules which have one of the following forms:

$$A \rightarrow \alpha \qquad A \rightarrow B^f \qquad A^f \rightarrow \alpha$$

where $A, B \in V$ are variables; $f \in F$ is a flag, and $\alpha \in (V \cup T)^*$. The flags are always associated with a variable and denoted as a superscript of the variable. During a derivation the set of flags associated with a variable can increase or decrease like a stack. The productions are applied as follows. A rule of the form $A \to \alpha$ substitutes the variable A by α and all flags associated with A (if any) are copied to the variables occuring in α. A rule of the form $A \to B^f$ adds a flag to variable A while replacing A with B. A rule of the form $A^f \to \alpha$ deletes a flag from A and then proceeds as in the first type of rule.

As an example, an indexed grammar can derive the language consisting of words of the form $w\overline{w}$ which is non-contextfree. Here for $w \in \{g, c, a, u\}^*$ the string $\overline{w}$ denotes the string of complementary bases (i.e. $\overline{g} = c$, $\overline{c} = g$, $\overline{a} = u$, $\overline{u} = a$), *in the same order* as in w:

$$S \to bS^b \mid A \qquad A^b \to A\overline{b} \qquad A \to \varepsilon$$

More complicated pseudoknot structures are of the form $u\,w\,\overline{u}^R\,\overline{w}$ or $u\,w\,\overline{u}^R\,\overline{w}^R$ where R indicates the reverse order. Both can be easily described by indexed grammars. For example, the latter one can be derived by (correcting a mistake in [10], formula (64)):

$$S \to bS^b \mid A \qquad A \to bA\overline{b} \mid B^b \qquad B \to \overline{b}B \qquad B \to \varepsilon$$

Now several things can be observed. First the above examples which can model all "naturally occurring" pseudoknots are actually *linear* indexed grammars (there is only one variable on the right-hand side of rules), and for linear indexed grammars the parsing problem can be solved in polynomial time [5]. (Notice, for arbitrary index languages parsing needs exponential time [6].)

Additionally, since index grammars "look like" context-free grammars, stochasticity can be added as described in the beginning. Therefore, we can implement a Viterby-like dynamic programming algorithm which parses a given RNA sequence and selects the derivation with the highest probability. This derivation should (at least up to same realistic scale) reflect the most likely folding structure of the RNA including pseudoknots.

References

[1] A.V. Aho. Indexed grammars - An extension of context-free grammars. *Journal of the ACM*, Vol. 15, No. 4, Oct. 1968, pp. 647–671.

[2] M. Brown, C. Wilson: RNA pseudoknot modeling using intersections of stochastic context-free grammars with applications to database search. Pacific Symposium on Biocomputing, 1996.

[3] R. Durbin, S. Eddy, A. Krogh, G. Mitchison. *Biological Sequence Analysis.* Cambridge University Press, 1998.

[4] G. Gazdar: Applicability of indexed grammars to natural languages. in: Natural Language Parsing and Linguistic Theories. Reidel, 1988, 69–94.

[5] L. Kallmeyer, W. Maier: Parsing beyond context-free grammar - linear indexed grammars. Manuscript, Uni Tübingen, 2006.

[6] R.B. Lyngo, C.N.S. Pedersen: RNA pseudoknot prediction in energy based models. *J of Comp. Biology* 7, 3/4 (2000) 409–427.

[7] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjölander, R. C. Underwood, D. Haussler: Recent Methods for RNA Modeling Using Stochastic Context-Free Grammars. *CPM* 1994: 289-306

[8] Y. Sakakibara, M. Brown, R. C. Underwood, I. S. Mian, D. Haussler: Stochastic Context-Free Grammars for Modeling RNA. *HICSS* (5) 1994: 284-294

[9] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjölander R.C. Underwood, D. Haussler: Stochastic Context-Free Grammars for tRNA Modeling. *Nucleic Acids Research*, 1994, Vol. 22, No. 23, pp. 5112–5120.

[10] D.B. Searls. The Computational Linguistics of Biological Sequences. In: L. Hunter (ed.). *Artificial Intelligence and Molecular Biology*. AAAI Press. 1993. pp. 47–121.