SCFGs as Predictors of RNA’s Secondary Structure

Antonio F. Martínez-Alcántara*

Bioc218 Computational Molecular Biology,
Final Paper
antonio@colpos.mx

Abstract

The use of bioinformatics’ tools for the study of non-coding RNA still remains a difficult task since RNA’s primary structure (the sequence itself) is not as informative as that of coding RNA and for this reason, research must rely on the study of secondary structure. One of the most promising tools in the field is the use of formal grammars. In this article a critical review of different methodologies based on the use of formal grammars for predicting the secondary structure of RNA sequences is presented. We conclude with a discussion of possible improvements on the described tools.

Key words: RNA Secondary Structure Prediction, Formal Grammar, Stochastic Context Free Grammar.

1. Introduction

RNA molecules play an important role in biological entities. Their most recognized function is as messengers in the process of protein translation; however, there are many other forms of RNA some of which have an influence on the way genes are expressed [Mount]. These functions are not associated with the region that encodes for proteins but are stored in the larger non-coding regions of the genome, once classified as ‘junk DNA’ [Pesole]. Non-coding RNA (ncRNA) is produced by special genes whose transcripts are used directly as RNA instead of producing proteins [Eddy99]. Unlike coding RNA, the main source of information in ncRNA resides in the interactions between self-complementary regions of the single stranded RNA molecule [Mount]. The interactions of the self complementary regions shapes the RNA molecules into different stable patterns (double stranded regions, stacked regions, hairpins, etc.) referred to as ‘RNA’s secondary structure’ (Figure 1), which in turn interact with each other to form the three dimensional (‘tertiary’) structure. Therefore, the study and prediction of the secondary structure of RNA, is crucial in understanding its function.

1.1. Non-coding RNA

It is widely recognized that ncRNA segments in the 5’ and 3’ untranslated regions (UTR) of genes are involved in the control of post-transcriptional pathways [Pesole], but many other ncRNA’s have been described and their number continues to grow.

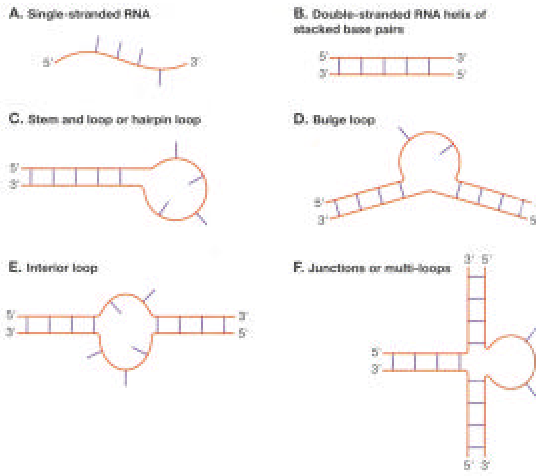
A. Single-stranded RNA
5'
3'
B. Double-stranded RNA helix of stacked base pairs
5'
3'
3'
5'
C. Stem and loop or hairpin loop
5'
3'
D. Bulge loop
5'
3'
5
E. Interior loop
5'
3'
3'
5'
F. Junctions or multi-loops
3' 5'
5'
3'
5' 3'

Figure 1. RNA’s Secondary Structure

Source: [Mount]

* ISEI, Colegio de Postgraduados, México

Their relevance has moved from the more traditionally recognized tasks of regulating the stability of mRNA, controlling mRNA localization and its translation efficiency [Pesole], to a newly recognized diverse set of functions; including the processing transfer RNAs, translocation of proteins, and regulation during meiosis. Interest in ncRNA is not only driven by its diversity and variety of functions, but there is also new interest in studying the evolution of ncRNAs: the hypothesis that RNA based life preceded both to proteins and DNA in evolution (the 'RNA World Hypothesis' for the origin of life) [Eddy99], [Eddy02].

A deeper study of ncRNA poses new challenges to bioinformaticians: current techniques available (for sequence alignment, database searches, gene finding, etc.) have been directed to the study of coding RNA and are not up to the challenge [Eddy02] of handling the less informative, more diverse ncRNA.

Several techniques have been tried in attempts to predict the secondary structure of RNAs based on the sequence of nucleotides of its molecule, all of them with limited success. To date, the most successful approaches are those that take into consideration thermodynamic information (the 'free energy' of a given predicted structure) as the guiding parameter to be minimized in the process of optimization [Do]. However, these methodologies are expensive and time consuming, considering that they are based on experimentally measured information. It is then desirable to have handy cheaper and faster techniques that are as accurate as or more so than those based on free energy minimization. From the several computational alternatives that have been tried, those based on probabilistic methods have proved to be the best candidates, namely, the Stochastic Context Free Grammars (SCFGs) [Durbin] and more recently Conditional Log-Linear Models (CLLMs) [Do].

2. RNA: The Linguistic Approach

There is a prevailing metaphor that considers the genetic information in DNA to be 'the language of life'. This analogy can be taken as far as comparing adjustments made during the evolution of genomes and languages [Zhang].

In the early eighties efforts began to represent nucleic acid sequences as "words over the alphabet of nucleotides" [Brendel]. The tools used were limited as was the success of the attempt, however an important step was taken: a key idea emerged that formal languages could be applied to the study of biological sequences.

2.1. Formal Languages

In order to have an understanding of the capabilities of formal languages as a tool for representing biological sequences, it is necessary to have a minimal understanding of the formal language theory itself. There are plenty of excellent texts that cover the subject in depth e.g. [Hopcroft]. For a quick and to the point introduction to the subject see [Durbin]. Here we will say that a formal language is a set of strings of symbols or 'words' (e.g. RNA and protein strings) over an alphabet (e.g. the set of nucleotides or the set of aminoacids) that can be generated by rewriting rules, a 'grammar'. Grammars can be ordered in a hierarchy, 'The Chomsky Hierarchy' after its creator Noam Chomsky, from simpler to more complex, based on their capability to represent features of words of a (formal) language. For each grammar of this form, there exists a corresponding 'processing machine', called 'automaton' (plural: 'automata'). Grammars and automata are two different but closely related things: grammars *generate* the strings of a language; an automaton for a given grammar is a (abstract) 'machine' that *parses* (a *parse* is a derivation of some sequence through continuous application of rewriting rules) a string and either accepts or rejects a word as belonging or not to the language with which the automaton is associated.

The simplest grammars are called 'Regular Grammars' (RGs) and they can be quickly parsed by the machines associated to them, known as 'Finite State Automata' (FSA). The structure of the rewriting rules is typically $S \longrightarrow cS$ (where the symbol S is called a 'non-terminal' and c is called a 'terminal' symbol) and represents the possibility of starting a word of the language by replacing the starting symbol S by the string cS, in which we again can replace the S by cS. We can keep going indefinitely, creating a sequence (for this case) *cccc*…. In order for the process to stop a special rule $S \longrightarrow e$ is required. It represents the possibility of replacing the non terminal S by the terminal e (the 'empty string'). As we can see from the example, the grammar generates words of a very simple language in which the words are strings of symbols c of any length (0 to infinite). The possibilities of RG are limited given their simplicity, however they were used in the earliest attempts to characterize RNA sequences [Brendel]. In that case, the authors found a regular language (and a corresponding FSA representation) of RNA sequences of group I RNA phages. Their study was limited in that they considered only the primary structure of RNA, leaving aside secondary features. Another example on how regular grammars can be used for biological sequences is found in [Durbin] where the authors observe how PROSITE patterns clearly correspond to regular grammars. Even in this case where the regularity of the patterns can be directly represented by these simple grammars, increasing amounts of sequences generate multiple exceptions that make it difficult to find good patterns. This requires the patterns to be narrowed to such an extent that they become too general and may match a large number of sequences, reducing their usefulness.

The most important limitation of regular grammars come from their inability to represent slightly more complex features of a language that would be fundamental for enabling a formal grammar to represent the secondary structure of RNA molecules. As stated before, RNAs secondary structure is due to interactions between self complementary nucleotides present at distant positions of the same string (see Figure 1). An example of a self complementary sequence of nucleotides is *gauauc* which can be folded into a structure similar to C in Figure 1, since the first three nucleotides (*gau* in 5' to 3' order) are complementary to the last three nucleotides (*cua*, in order 3' to 5'). From the point of view of formal language theory this type of string can be generated by the so called "Context Free Grammars" (CFGs).

For CFGs the rewriting rules are typically of the form $S \longrightarrow aSu$. The associated processing machines are called 'Pushdown Automata' (PDA) and they are not as computationally efficient as their FSA counterpart. The name reminds us that this kind of automata require an auxiliary device (a 'pushdown' stack) that helps in 'remembering' symbols (for example the complementary portion of an RNA string). An example of a CFG which could generate the string *gauauc* above is $S \longrightarrow gSc$; $S \longrightarrow aSu$; $S \longrightarrow uSa$; $S \longrightarrow e$, applied in that order. Realistic secondary structures like stem loops, bulges and arbitrarily branched structures can be easily generated by CFGs.

In spite of the great possibilities of CFGs for representing secondary structure features of RNAs, there are some details that are worth pointing out. First of all, the fact that we want to use CFGs binds us to the computational properties of the abstract machines associated with them. The PDA are computationally time consuming machines. Another point worth noting is the inability of CFGs to represent other 'non-orthodox' secondary structures [Searls] such as pseudoknots (Figure 2).

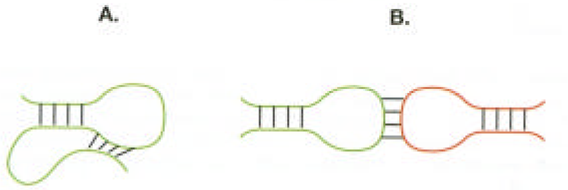

Figure 2. Pseudoknots (A); Kissing hairpins (B)

Source: Mount, 2004

In the literature we can find other formalisms (e.g. tree adjoining and other "mildly context-free" grammars [Searls]), able to deal with a larger spectrum of RNA secondary structures.

One is tempted to ask about other grammars which are higher in the hierarchy and therefore are more powerful and more capable of representing features for which CFGs are limited. Again, there are theoretical limits that bind more powerful grammars to slower processing machines and, at times, to machines incapable of giving an answer (for details see [Hopcroft]).

Up to this point we have stated that CFGs (and other grammars) are capable of representing RNA's secondary structure, and we as well benefit from the theoretical existence of their corresponding processing machines. Although from a biological point of view it sounds promising to have machines that recognize different strings of nucleotides as belonging or not to a certain class, the reality is that the capabilities of such machines are limited to, for example, looking up certain patterns in databases but they will not help to predict the secondary structure of an RNA molecule.

3. Secondary Structure Prediction

In order to be able to predict the secondary structure of RNA, CFGs elicit additional help from probability theory, giving rise to the use of Stochastic Context Free Grammars (SCFGs) that have been used in the area of speech recognition [Sakakibara]. In general, all of the grammars in the hierarchy of Chomsky can be used in a stochastic fashion. Non stochastic versions of the grammars either do or do not generate a terminal symbol. A stochastic version of the same grammar would apply a rule (generating some symbol(s)) with a probability *p*, generating the different words of the language with a certain probability [Durbin]. In this way given a sequence and a grammar, we can score (and then rank) the possible parses and then infer which is the most optimal.

Probability theory has long been used as an aid to bioinformatics: algorithms for sequence alignment and the fashionable Hidden Markov Models (HMMs) themselves can be seen as equivalent to variations of stochastic grammars [Durbin]. It is not surprising that the algorithms used for training and scoring stochastic grammars resemble those used for HMMs.

3.1. Stochastic Context Free Grammars

Stochastic Context Free Grammars can be thought of as a “joint probability distribution over RNA sequences and their secondary structures” [Do]. A SCFG G defines: a) A set R of rewriting rules; b) A probability distribution over the rewriting rules; c) A mapping from the different derivations to secondary structures. The following example from Do describes a SCFG that works for some RNA secondary structures:

a) Rewriting rules: $S \longrightarrow aSu$; $S \longrightarrow uSa$; $S \longrightarrow cSg$; $S \longrightarrow gSc$; $S \longrightarrow gSu$; $S \longrightarrow uSg$; $S \longrightarrow aS$; $S \longrightarrow cS$; $S \longrightarrow gS$; $S \longrightarrow uS$; $S \longrightarrow e$;

b) Probabilities: The rules have an associated set P of probabilities in such way that each of the above r_i rules ($i=1,..,11$) would have an associated probability p_{ri} of being applied.

c) Mapping derivations to structures: a secondary structure for a given derivation pairs two letters *iif* the two letters were generated at the same step during the derivation.

Given this grammar, the sequence $x=agucu$ (which has an associated secondary structure s), the parse p that generates it is:

$$S \longrightarrow aSu \longrightarrow agScu \longrightarrow aguScu \longrightarrow agucu,$$

and so the probability of the sequence AND parse[1] is:

$$P(\, x,p\,) = p_{r1} x\, p_{r4} x\, p_{r10} x\, p_{r11}.$$

It is not difficult to combine information from an evolutionary model with that from a biophysical model in order to express them in the form of probabilities that affect the application of a given rule in a grammar G, hence obtaining a more powerful model [Dowell].

Once we have written a grammar that can model the strings of some specific type of RNA molecules, we are faced with three problems [Durbin]:

a) Finding the *optimal derivation* of a sequence x given G

b) Finding the *probability* of a sequence x given G

c) Estimating an optimal set P of probabilities associated with each rule, given a set of pairs $\{x_i,s_i\}$ of sequences x_i and their respective secondary structure s_i (preferably validated by experimental means).

For SCFGs there also exist Dynamic Programming (DP) algorithms that will find optimal solutions. Each of these problems are solved in a manner analogous to the way they are solved in the case of HMMs). The DP algorithms for SCFGs and the problems they solve are presented in Table 1 (for details about the algorithms see [Durbin]).

Table 1. DP Algorithms for SCFGs

Problem	Algorithm
a) Optimal derivation given G	CYK[2]
b) Probability of x given set G	Inside
c) Estimation of set P	Inside-Outside

Source: [Durbin]

[1] There could be several ways of deriving the same string using the same grammar and applying the rewriting rules in a different order (this is called ‘ambiguity’ of a set R of rewriting rules), however for this particular grammar there is only one way of obtaining the example string (R is unambiguous).

[2] The CYK algorithm will not efficiently calculate an optimal derivation if the set R is ambiguous. However there is not an easy way of deciding whether or not a given grammar is ambiguous [Dowell].

3.1.1. How many parameters?

Given that SCFGs can incorporate information from diverse sources in the form of parameters of the model, one might be tempted to think that by adding more parameters the model gets better. In general this is never the case and it also holds untrue for SCFGs.

Thus a relevant question that deserves attention is: What is the best tradeoff between complexity and accuracy of predictions? To answer this question [Dowell] evaluated a set of small grammars by testing the accuracy of their predictions using a set of known RNA secondary structures and comparing the results against the energy minimization methods. Their conclusion was that the accuracies reached by compact grammars are not distant from the energy minimization methods, but the methods based on physics are still better.

From the grammars they evaluated, the best performer was the grammar that is used by Pfold [Knudsen03]. This grammar is surprisingly simple: $S \longrightarrow L$; $S \longrightarrow LS$; $F \longrightarrow dFd$; $F \longrightarrow LS$; $L \longrightarrow s$; $L \longrightarrow dFd$; "*s* symbolizes a base in a single string and *ds* symbolizes bases that pair up in a stem. The nonterminal *S* produces loops and *F* produces stems, while *L* decides whether a specific loop position should be a single base or the start of a new stem" [Knudsen99]. Figure 3 shows examples of the use of this grammar. Pfold uses a SCFG for producing a prior probability distribution of RNA structures, but its novelty is the additional use of phylogenetic information: Pfold starts by taking an alignment of RNA sequences believed to share a common secondary structure as input. Those sequences will serve to obtain a consensus sequence *CS* and a tree *T* relating the sequences[3]. Using *CS* and *T*, the grammar is used to estimate the plausible secondary structure common to the sequences in the alignment.

3.2. Other probabilistic approaches

As noted by [Dowell], none of the SCFGs in their study performed as good as the free energy based methods. That is one reason why many groups are still pursuing alternatives to the physics based models which rely on "thousands of experimentally-based thermodynamic parameters" [Do]. In this section we will describe a promising alternative, not strictly based on SCFGs but deeply related to them: Conditional Log-Linear Models (CLLMs).

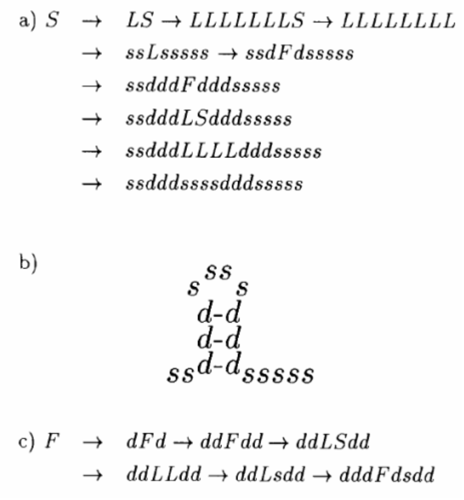
a) $S \rightarrow LS \rightarrow LLLLLLLS \rightarrow LLLLLLLL$
$\rightarrow ssLsssss \rightarrow ssdFdsssss$
$\rightarrow ssdddFdddsssss$
$\rightarrow ssdddLSdddsssss$
$\rightarrow ssdddLLLLdddsssss$
$\rightarrow ssdddssssdddsssss$
b)
s ss s
d-d
d-d
ss d-d sssss
c) $F \rightarrow dFd \rightarrow ddFdd \rightarrow ddLSdd$
$\rightarrow ddLLdd \rightarrow ddLsdd \rightarrow dddFdsdd$

Figure 3. Example of use of the grammar used by Pfold

Source: [Knudsen99]

3.2.1 Conditional Log-Linear Models

This is a novel approach to RNA's secondary structure prediction due to Do et al. Conditional Log-Linear Models (CLLMs) are a generalization of SCFGs (i.e. for each SCFG there is an equivalent CLLM) that, according to the evaluation done by its authors, have accuracies that are better than those of the current probabilistic and physics based models. CLLMs are beneficial in that they hold the possibility of representing complex scoring systems (like the ones used by physics models). An additional bonus is the possibility of having a way of controlling the sensitivity and specificity of the algorithm.

A key observation that hints at the possibility of using CLLMs is that SCFGs can be rewritten as log-linear models (LLM) in which the parameters (called 'weights') are constrained to take values from a restricted set. Another constraint is that other parameters of the LLM (called the 'features of the model') are restricted by the complexity of the grammar. For a LLM those restrictions are unnecessary and removing them opens new possibilities for the model. An immediate change due to this removal of restrictions is that the estimation of parameters is done in a different way, but it still closely follows the traditional inside and outside algorithms used for SCFGs.

[3] If *T* is not given it has to be estimated from the data.

The scene is completed by the possibility of straightly transferring the complex scoring terms of physics based models to CLLMs.

CONTRAfold, the CLLM implementation in [Do], takes into account 13 different features (amongst others: base pairs, hairpin lengths, helix lengths, bulge and internal loop lengths and free bases). The authors studied the set of grammars used by [Dowell] and generated their corresponding CLLM. In all but two cases CLLMs performed better than the corresponding SCFGs, and in the two cases in which SCFGs did better the differences were actually small. When CLLMs were compared against other commonly used probabilistic and free energy methods, the difference again favored CLLMs, even when compared with the current best method, Mfold [Zuker].

4. Possible Improvements

Along this research several approaches and opinions have been reviewed. It is not difficult to notice that none of the proposals can be considered final. The current best methods (energy based) are criticized for being expensive and time consuming, since large numbers of parameters have to be derived by experimental means. In addition, several different foldings lay around the minimum energy point causing difficulty in deciding which one is better [Mount].

As for the linguistics based methods material of this work, there are several possibilities that could be worth exploring. The proposals are not presented in any order of relevance.

In statistics some of the research has been oriented towards new methods for estimating parameters, whereas in the papers referenced, the method traditionally used is Expectation Maximization. This could be an avenue of research: the study of the behavior of the different methodologies under different approaches to parameter estimation.

In the study of performance of grammars by [Dowell], the best performer was Pfold. This tool uses a simple grammar AND aid from phylogenetic considerations. It is worth taking into account this type of ‘aid’ for the algorithms used in secondary structure prediction. For example, in the case of CLLMs: Could there be an improvement in the predictions by using this kind of additional information?

Finally, given that one of the difficulties in studying the secondary structure of ncRNAs is due to its diversity [Eddy99], another idea could be to have an assessment similar to the one by Dowell. In this case the performance of various grammars would be evaluated for different groups of ncRNA in order to obtain both good and bad performers depending on the type of RNA under study.

References

[Brendel] Brendel B, Busse HG: *Genome Structure Described By Formal Languages.* Nucl Acids Res 1984, 12:2561-2568.

[Do] Do CB, Woods DA, Batzoglou S: *CONTRAfold: RNA Secondary Structure Prediction Without Physics-Based Models.* Bioinformatics 2006 22, 14, pages e90-e98.

[Dowell] Dowell Robin D, Eddy Sean R: *Evaluation of Several Lightweight Stochastic Context Free Grammars for RNA Secondary Structure Prediction.* BMC Bioinformatics 2004, 5:71

[Durbin] Durbin R, Eddy SR, Krogh A, Mitchinson GJ: *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids* Cambridge UK: Cambridge University Press; 1998.

[Eddy99] Eddy SR: *Noncoding RNA Genes.* Current Opinion in Genetics & Development 1999, 9:695-699.

[Eddy02] Eddy SR: *Computational Genomics of Noncoding RNA Genes.* Cell 2002, 109:137-140.

[Hopcroft] Hopcroft JE, Ullman JD: *Introduction to Automata Theory, Languages and Computation.* Addison Wesley 1979.

[Knudsen99] Knudsen B, Hein J: *RNA Secondary Structure Prediction Using Stochastic Context-Free Grammars and Evolutionary Story.* Bioinformatics 1999, 15(6):446-454.

[Knudsen03] Knudsen B, Hein J: *Pfold: RNA Secondary Structure Prediction Using Stochastic Context-Free Grammars.* Nucl Acids Res 2003, 31:3423-3428.

[Mount] Mount David W: *Bioinformatics: Sequence and Genome Análisis* 2nd Edition, Cold Spring Harbor Laboratory Press, Cold Spring Harbor, New York USA; 2004.

[Pesole] Pesole G, Grillo G, Larizza A, Liuni S: *The Untranslated Regions of Eukaryotic mRNAs: Structure, Function, Evolution and Bioinformatic Tools for Their Analysis.* Briefings in Bioinformatics 2000, 1(3):236-249.

[Sakakibara] Sakakibara Y, Brown M, Hughey R, Mian IS, Sjölander K, Underwood RC, Haussler D: *Stochastic Context-Free Grammars for tRNA Modeling*. Nucl Acid Res 1994, 22(23):5112-5120.

[Searls] Searls DB: *The Language of Genes.* Nature 2002, 420 (November) 211-217.

[Zhang] Zhang HY: *The Evolution of Genomes and Language*. EMBO Reports 2006, 7(8):748-749.

[Zuker] Zuker M: *Mfold WebServer for Nucleic Acid Folding and Hibridization Prediction.* Nucl Acids Res 2003, 31(13):3406-3415.