Grammars in sequence modeling

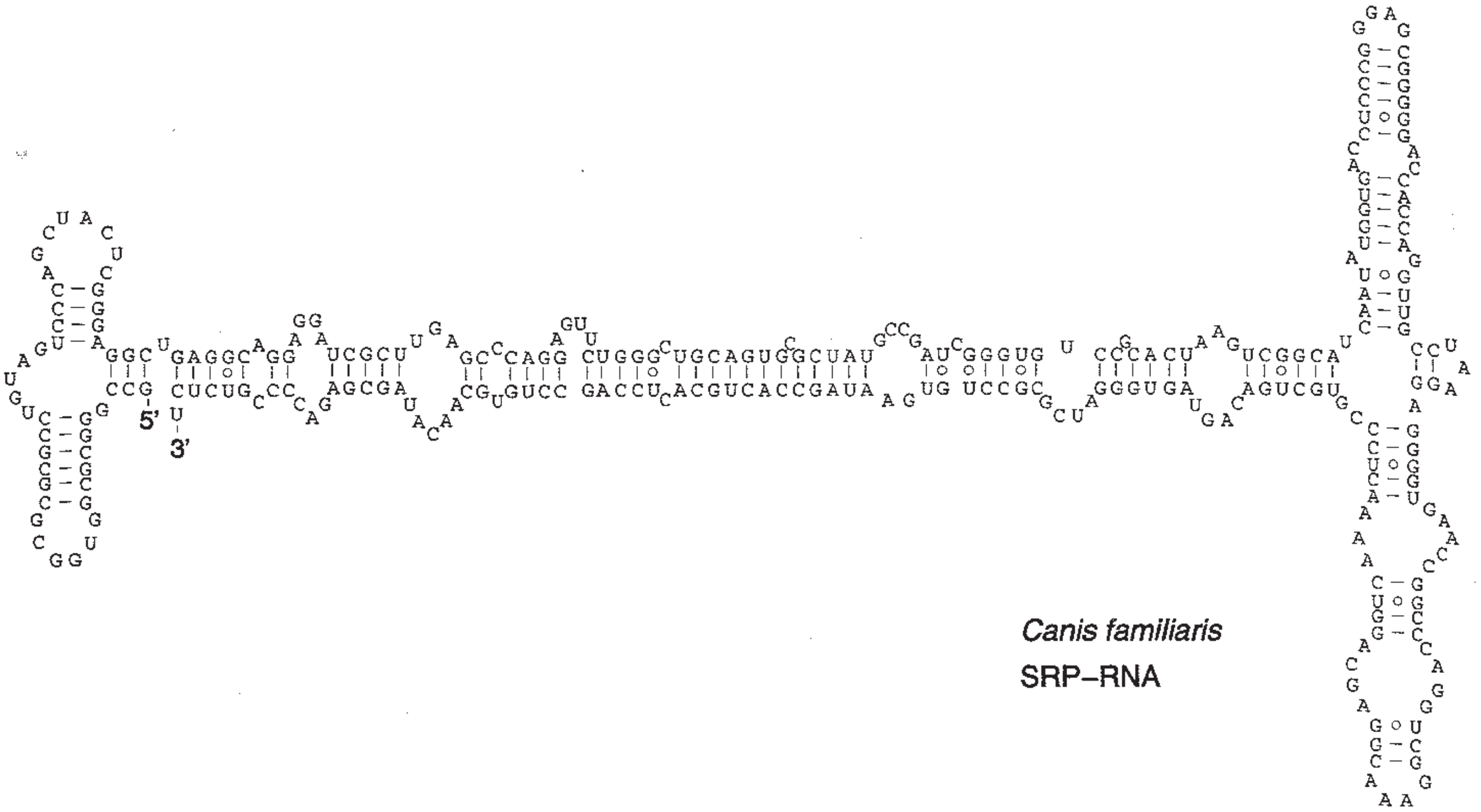
P.Antal

antal@mit.bme.hu

BME

RNA I

RNA is single stranded sequence of bases A, U, C, G, but base pairs arise such as A-U, G-C (canonical) or non-canonical pairs such as G-U, which are relatively stable as well. ⇒ An RNA strand has complex structure because of (linearly) distant, but paired bases. RNA is not just a ”messenger", but effector (autocatalytic RNAs) (⇒ ”RNA world" hypothesis).



Canis familiaris

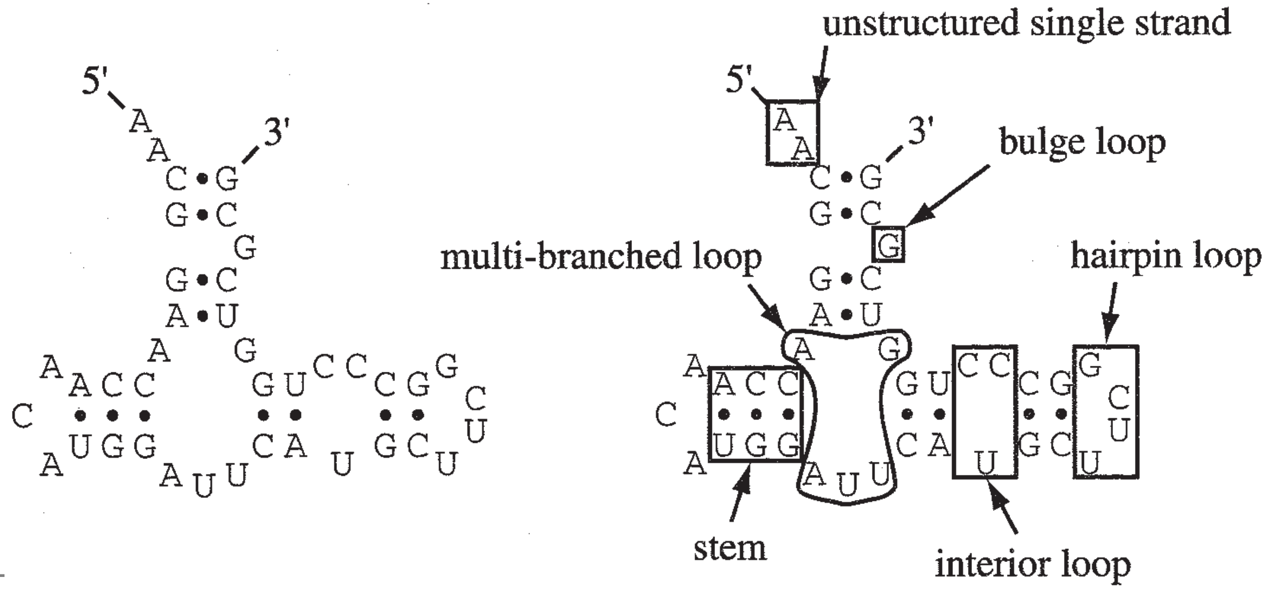
SRP–RNA

RNA II

Consecutive stacked base pairs called *stem* form A-form double helix (distorted by non-canonical pairs). A stem is surrounded by single stranded subsequences called *loops* (bulge/interior/hairpin and multibranch loops). These form the secondary structure of the RNA sequence.

Type of interactions:

1. **nested**: $(i, j), (i', j')$ pairs are nested-pairs if not related (e.g. $i < j < i' < j'$) or nested (e.g. $i < i' < j' < j$),
2. **non-nested**: base-pairs: copies, meta/reversed-copies (1%).



Grammars for sequence modeling

Profile HMMs allows (1) exploration, (2) decision on membership and (3) multiple alignment for proteins. (Semihidden) HMMs can be used for DNA. And for RNA with distant complementer regions?
Note that first-order time-homogeneous HMMs are equivalents to stochastic FSAs and regular grammars: a grammar scheme for profile HMMs.

Start	1	...
$S_{M_0} \to \hat{I}_0 \vert \hat{M}_1 \vert \hat{D}_1$	$M_1 \to \hat{I}_1 \vert \hat{M}_2 \vert \hat{D}_2$	$\dots$
$\hat{I}_0 \to a\hat{I}_0 \vert aI_0 \vert \dots$	$\hat{M}_1 \to aM_1 \vert \dots$	$\dots$
	$\hat{I}_1 \to aI_1 \vert a\hat{I}_1 \vert \dots$	$\dots$
	$\dots$	$\dots$

Goal: a semantic, static, probabilistic model for a given set of homologous sequences. That is we would like a non-dynamic model for sequences deriving from a common ancestral sequence through a hidden (stochastic but already fixed) evolutionary process, i.e. to avoid the problems of modeling the underlying phylogeny process, but to model the effects of evolutionary constraints over distant regions (more advanced basic models for mutations and indels, which are used in a static framework).
Goal': a stochastic canonical context-free grammar schemata, which can be specialized and parameterized for a given set of homologous RNA sequences.

Grammars

Goal: definition of a given set of words (language $\mathcal{L}$) over a finite alphabet Σ.

Generative/transformational grammars: Members of the language can be derived using *rewrite rules* containing *terminal* and *nonterminal* symbols (denoted with small and capital letters).

Parsing consists of the reconstruction of a derivation/parse tree (alignment).

Questions:

1. parsing: find parse T resulting in terminal sequence x
2. membership: $x \in \mathcal{L}^G$ or is there any parse T resulting in terminal sequence x

Chomsky hierarchy of grammars (*:right/left, with/without ϵ;**:nondecreasing):

Grammar	Rule	Automaton	Parsing	Language
regular*	$W \rightarrow aW$	FSA	linear	a reg.expressio
context-free	$W \rightarrow \beta$	push-down	polynomial	palindromes
context-sensitive**	$\alpha_1 W \alpha_2 \rightarrow \alpha_1 \beta \alpha_2$	linear bounded	exponential	copies
unrestricted		Turing machine (TM)	semidecidable	$KB - FOL$
-	-			halting TMs

Complexity of parsing=>CFGs

Stochastic grammars

Rewrite rules in grammar G have application probabilities (θ denotes their vector).
Questions (T_x denotes parse tree with terminal sequence x):

1. parsing: $T_x^* = \arg\max_{T_x} p(T_x|\theta, G)$
2. membership: $p(x|\theta, G) = \sum_{T_x} p(T_x|\theta, G)$
3. parameter learning: $\theta^* = \arg\max_\theta p(x^{(1)}, \ldots, x^{(n)}|\theta, G)$
4. posterior decoding:
   $p(W \dashrightarrow x_{i:j}|x, \theta, G) = \sum_{T_x} p(T_x|\theta, G)$ 1($x_{i:j}$ is generated from W in parse tree T_x")

SCFG algorithms

Assume: M nonterminals ($W = W_1, \ldots, W_M$), Chomsky normal form ($W_v \rightarrow W_y W_z$ or ($W_v \rightarrow a$) with transition and emission probabilities $t_v(y,z)$ and $e_v(a)$

The **inside** algorithm computes the probability of sequence x $p(x)$ summing over all possible derivation (parse tree).

Idea: calculate recursively the probability $\alpha(i,j,v)$ of a parse subtree rooted at nonterminal W_v for subsequence $x_{i:j}$ for all i, j, v.

```
Require: SCFG,x
Ensure: $p(x|SCFG)$
  Ini: i=1 to L, v=1 to M: $\alpha(i,i,v) = e_v(x_i)$
  for i=1 to L-1 do {length}
    for j=1 to L-i do {starting positions}
      for v=1 to M do {states}
        $\alpha(j,j+i,v) = \sum_{y=1}^{M} \sum_{z=1}^{M} \sum_{k=j}^{j+i} \alpha(j,k,y)\alpha(k+1,j+i,z)t_v(y,z)$
  End: $p(x|SCFG) = \alpha(1,L,1)$
```

The **outside** algorithm computes a probability called $\beta(i,j,v)$ of a complete parse tree for sequence x, excluding subtrees with W_v nonterminal and $x_{i:j}$ leaves.

The optimal parse tree can be found by the **Cocke-Younger-Kasami (CYK)** algorithm: same as inside with $\max_{y,z,k}$ instead of $\sum_{y,z,k}$ and with pointers for backtracking.

HMMs/SFSAs/SRGs versus SCFGs

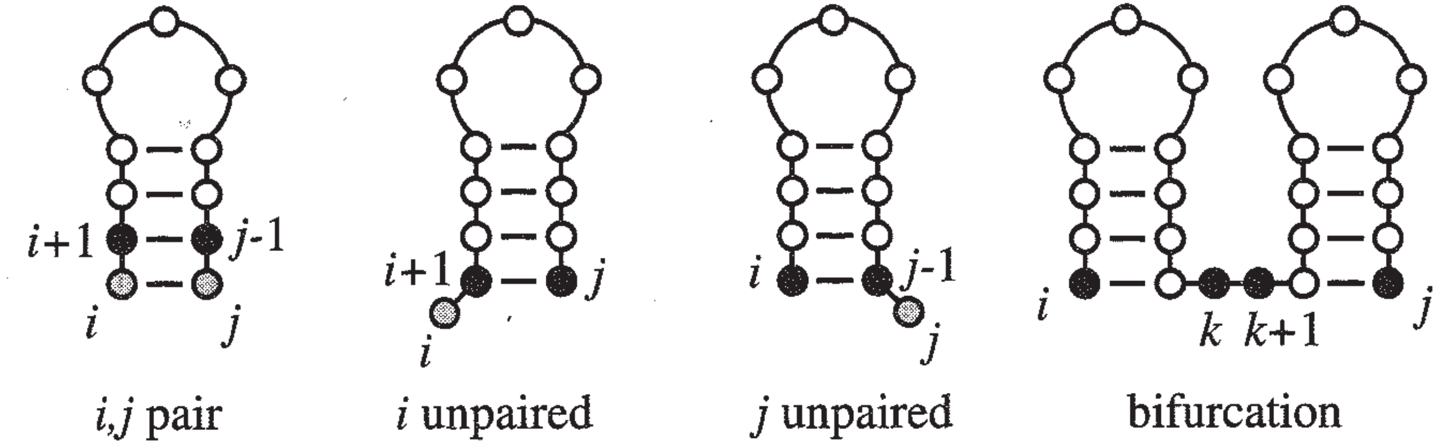
The same questions for stochastic context free grammars (SCFGs) modeling RNA: (X^h/X^o hidden/observed variables)

Goal	stochastic regular grammars	stochastic context-free grammars
Explanation:$p(X^h\|X^o, \theta^M, M)$	alignment: Viterbi	parse tree: CYK
Matching:$p(X^o\|\theta^M, M)$	p(sequence): forward alg.	p(seq.): inside alg.
Canonical model class:$M \in \mathcal{M}$	profile HMMs (length)	covariance models
Imputation-based parameter learning:θ^M	Viterbi-based	CYK-based
EM-based parameter learning:θ^M	forward-backward	inside-outside
Time complexity	$\mathcal{O}(LM^2)$	$\mathcal{O}(L^3M^3)$
Space complexity	$\mathcal{O}(LM)$	$\mathcal{O}(L^2M)$

Note that SCFG models allows a more powerful representation of a distribution of homologous sequences than HMMs (e.g. allowing palindrome constraints) or phylogenetic tree with i.i. substitution stochastic process assumption.

PCFG:Covariance model I

An SCFG model of RNA folding based on four types of recursive extension (paired, left-unpaired, right-unpaired, bifurcation) (Nussinov)



$$S \rightarrow aSu|cSg|gSc|uSa \ \ \text{(paired)} \quad (1)$$

$$S \rightarrow aS|cS|gS|uS \ \ \text{(left} - \text{unpaired)} \quad (2)$$

$$S \rightarrow Su|Sg|Sc|Sa \ \ \text{(right} - \text{unpaired)} \quad (3)$$

$$S \rightarrow SS \ \ \text{(bifurcation)} \quad (4)$$

PCFG:Covariance model I

A generic stem model with six states (W denotes any states):

$$P \rightarrow aWa|\ldots \quad (\text{pairwise}, 16) \tag{5}$$

$$L \rightarrow aW|\ldots \quad (\text{leftwise}, 4) \tag{6}$$

$$R \rightarrow Wa|\ldots \quad (\text{rightwise}, 4) \tag{7}$$

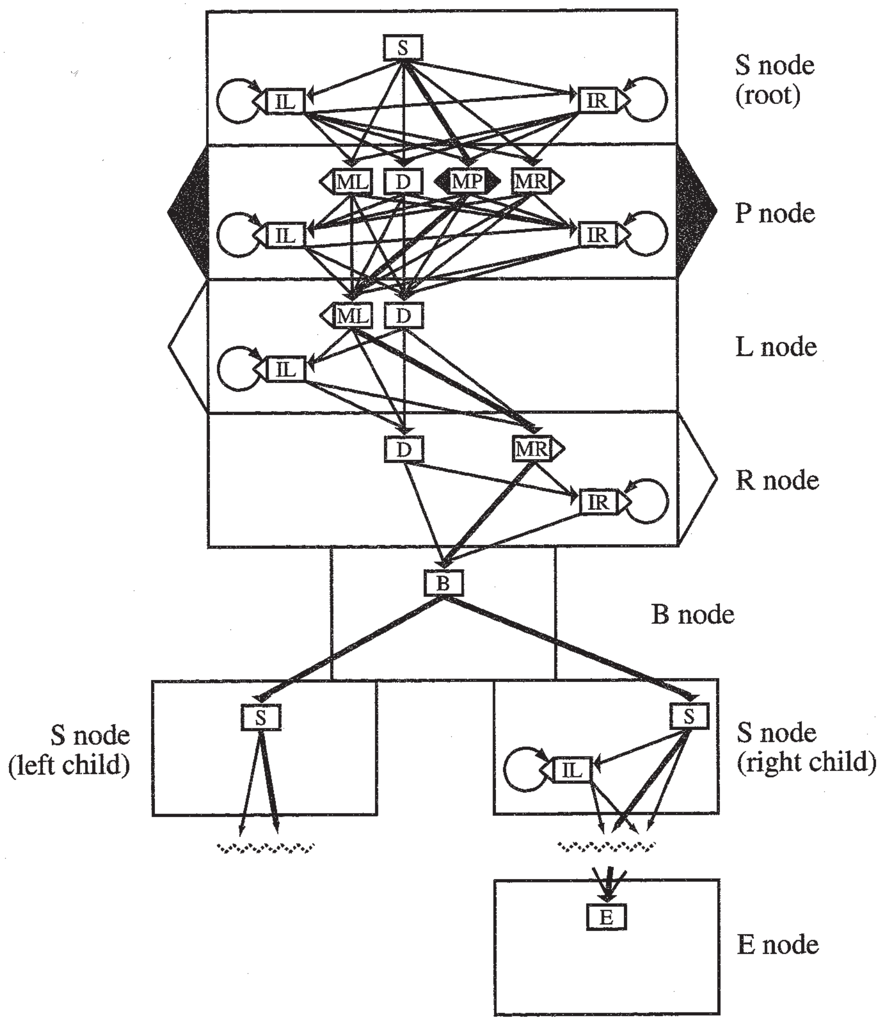
$$B \rightarrow SS \quad (\text{bifurcation}) \tag{8}$$

$$S \rightarrow W \quad (\text{start}) \tag{9}$$

$$E \rightarrow \epsilon \quad (\text{end}) \tag{10}$$

PCFG:Covariance model III

A special CFG called Covariance Model (CM) has the following building block:


S
IL
IR
S node
(root)
ML
D
MP
MR
IL
IR
P node
ML
D
IL
L node
D
MR
IR
R node
B
B node
S
S node
(left child)
S
IL
S node
(right child)
E
E node