# Stochastic Grammars

## Stochastic Context-Free Grammars

I. Holmes

Department of Bioengineering
University of California, Berkeley

Spring semester

# Outline

1. Overview of transformational grammars
2. RNA structure
3. Dynamic programming algorithms for SCFGs
4. Beyond SCFGs

- Overview: HMM profiles, HMM genefinders, SCFGs for RNA, repeats, beta-sheets; Natural Language Processing
- What is a transformational grammar? Formal definition: terminals $\Omega$, nonterminals $\Phi$, transformation rules
    - "Language" = set of strings generated by the grammar
    - "Parser" = computer program to **decide** if a given input string is in the language (returns "true" or "false")
    - More generally, we're interested in parsers that compute scores (energies, probabilities) for a given input string
    - These scores are associated with the transformation rules. The grammar is said to be score-*attributed* (Knuth)
- The Chomsky hierarchy of grammars and their associated parsers
    - Regular grammars: finite-state machines (HMMs)
    - Context-free grammars: pushdown automata (SCFGs)
        - The **parse tree**; the **inside sequence** and **outside sequence**
        - Chomsky Normal Form; Eddy *et al*'s "RNA Normal Form"

- Why RNA is important in evolution and cell biology
    - RNA world; pre- and post-transcriptional regulation; Crick's idea of studying simple examples; ribotechnology

- RNA structure terminology: basepairs, stems, loops, pseudoknots, kissing loops

Terms contributing to the free energy of a folded RNA structure (`scor.berkeley.edu`)

- Hydrogen bonding between bases. Canonical and noncanonical pairs.
- Stacking energies due to overlap of $\pi$-orbitals of adjacent planar basepairs
  - H-bonding and stacking terms can be combined and measured by direct experiment.
- Unusual configurations: tetraloops, triloops, triple-A platforms
  - Finite number of cases, so also amenable to experimental measurement.
- Entropic cost of closing loops
  - Theory: rods and Gaussian springs. Integrate out displacements, get likelihood ratio (Doi-Edwards, Isambert)
  - Statistics of random walk, $\langle|\Delta\mathbf{x}|^2\rangle \propto t$, and self-avoiding walk, $\langle|\Delta\mathbf{x}|^2\rangle \propto t^{1+\epsilon}$
    - Renormalisation (Edwards, de Gennes)

- The Nussinov algorithm: Finds *strictly nested* foldback structures, i.e. excluding pseudoknots.
  - RNA sequence $X$, length $L$, nucleotides $x_1 \ldots x_L$
  - Let $H(x, x') = 1$ if $xx'$ is a canonical Watson-Crick basepair, and 0 otherwise
  - Nussinov recursion finds the structure for $x_i \ldots x_j$ that has the most strictly nested canonical basepairs

$$S(i,j) = \max \Big( S(i+1,j), S(i,j-1), S(i+1,j-1) + H(x_i,x_j), \max_{i \le k \le}$$

  Best structure for $X$ is found by traceback from $S(1, L)$
  - Equivalent to the following **score-attributed grammar**

| Rule | | | Score |
|---|---|---|---|
| $S$ | $\rightarrow$ | $S\ x$ | 0 |
| | $\vert$ | $x\ S$ | 0 |
| | $\vert$ | $x\ S\ x'$ | H(x,x') |
| | $\vert$ | $S\ S$ | 0 |
| | $\vert$ | $\epsilon$ | 0 |

- Chomsky normal form. (RNA normal form is more useful in practise, but CNF is easier to present.)
  For nonterminals $A, B, C \in \Phi$ and terminals $a \in \Omega$:

| Rule | | | Name |
|---|---|---|---|
| $A$ | $\rightarrow$ | $B\ C$ | Bifurcation |
| | $\mid$ | $a$ | Emission |
| | $\mid$ | $\epsilon$ | Termination |

  Probabilities denoted by $P(\text{rule})$, e.g. $P(A \rightarrow BC)$
- Inside algorithm.
  Let $I_A(i,k) = P(x_i \dots x_{i+k}|A)$ be sum of probabilities for parse trees rooted in $A$ generating sequence $x_i \dots x_{i+k}$.

$$I_A(i,k) = \left( \sum_B \sum_C \sum_{j=0}^{k} P(A \rightarrow BC) I_B(i,j) I_C(i+j,k-j) \right) + \left\{ \begin{array}{l} 0 \\ P(A \\ P(A \end{array} \right.$$

NB loopy dependencies, e.g. if $P(A \rightarrow AA) \neq 0$ and

# Pair SCFGs, evolutionary SCFGs and tree transducers

- Evolutionary SCFGs: PFOLD (xfold, evofold, etc.)
  - As with Evolutionary HMMs, we can let the terminals be alignment columns
  - Again, terminal emission likelihood $P(A \rightarrow a)$ is implemented as Felsenstein pruning
- Pair SCFGs: Evoldoer. Version of TKF that describes evolution of RNA secondary structure (Holmes 2005).
  - Two kinds of TKF91 links model, recursively nested in a tree
  - Stem sequences rooted in $S$ nonterminals; basepair alphabet $\Omega^2$; ends in an $L$
  - Loop sequences rooted in $L$ nonterminals; nucleotide alphabet $\Omega$; $S$'s also allowed in sequence
    - Really need to adapt TKF91 model to allow deletion prob to depend on symbol, otherwise $S$ substructures get deleted at same rate as nucleotides
- Pair SCFGs (e.g. Stemloc, QRNA). Heuristic, but with lots

# Graph grammars

- Tree-adjoining grammars
  - Aravind Joshi (1975, 1985)
- Rivas-Eddy papers
  - A dynamic programming algorithm for RNA structure prediction including pseudoknots. JMB 1999.
  - The language of RNA: a formal grammar that includes pseudoknots. Bioinformatics 2000.
- Graph grammars: easy to describe and simulate, attractive for biology; but how does their DP work?
- Other grammars whose marginals are easy to compute by sum-product DP, e.g.
  - stochastic tree grammars (Abe and Mamitsuka, ISMB 1994)
  - context-sensitive HMMs with finite memory of last $N$ emitted characters (Yoon and Vaidyanathan, 2004)

# Discriminative grammars

Discriminative grammars: conditional log-linear models

- Recall HMMs and linear CRFs form a "generative-discriminative pair"
- The analogous discriminative model for SCFGs is a Conditional Log-Linear Model
- This allows a great deal of physics-like parameterization (loop entropies, stacking free energies, terminal mismatch...) without having to introduce many new nonterminals
  - Do, Woods and Batzoglou. "CONTRAfold: RNA secondary structure prediction without physics-based models." Bioinformatics 22:14, pp e90-e98

# Summary

- SCFGs