ncRNA Structure including simple pseudo-knots
RNA structure (including pseudoknots)

- **Basics:**
  - Structure is described by a set of base-pairings $M$.
  - Normally, the base-pairs do not interleave.
Incorporating pseudoknots in structure prediction

- Pseudoknots are only loosely defined.
- If any interleaving is allowed, then simply select a structure in which a max number of nucleotides can be paired.
- Under some restrictive notions, the structure problem becomes NP-hard.
A region $[i_0, k_0]$ forms a simple pseudoknot if there exist positions $j'_0, j_0$ s.t.
- Each $(i, j) \in M$ satisfies either
  - $i_0 \leq i < j'_0 \leq j < j_0$ or $j'_0 \leq i < j_0 \leq j \leq k_0$
  - Within one of the two groups, there is no interleaving.
  - $i < i' < j'_0$ or $j'_0 \leq i < i' \leq j_0$ implies $j > j'$.
Simple pseudoknotted structure

- Collection of simple pseudoknots ($M_i$) and other basepairs $M'$.
- None of the simple pseudoknot regions overlap.
- $M'$ is a secondary structure without pseudoknots for the sequence obtained by excising all pseudoknotted regions.
Main idea

- Rotate the sequence so that it forms two loops.
- Each allowed base-pair is a horizontal line in exactly one of the two loops. The horizontal lines in the two loops are non-crossing.
- All base-pairs can be ordered.
Main idea

- The base-pairs have a total order that a d.p. can exploit.
- \((i,j) < (i',j')\) if one of the following holds:
  - \(i'<j'<i<j\)
  - \(i<i'<j<j'\)
  - \(i<i'<j<j<j_0\)
  - \(j'<i<i<j<j'\)
• We need to construct an increasing path of base-pairs.
• Consider a ‘frontier’ triple $(i,j,k)$.
• We have the following cases:
  - $(i,j)$ form a base-pair
  - $(j,k)$ form a base-pair
  - Neither $(i,j)$ nor $(j,k)$ form a base-pair
Recurrences

- $S_L(i,j,k)$ is the optimum score of a frontier $(i,j,k)$ assuming that
  - $i < j' < j < j_0 < k$
  - $(i,j)$ form a base-pair
- $S_R(i,j,k)$ is the optimum score of a frontier $(i,j,k)$ assuming that $i < j' < j < j_0 < k$, and
  - $(j,k)$ form a base-pair
- $S_M(i,j,k)$ is the optimum score of a frontier $(i,j,k)$ assuming that $i < j' < j < j_0 < k$, and
  - Neither $(i,j)$ nor $(j,k)$ form a base-pair
Computing $S_L$

\[
S_L(i, j, k) = v(a_i, a_j) + \max \begin{cases} 
S_L(i-1, j+1, k) \\
S_M(i-1, j+1, k) \\
S_R(i-1, j+1, k)
\end{cases}
\]
Putting it all together

\[ S_R(i, j, k) = v(a_j, a_k) + \max \left\{ S_L(i, j + 1, k - 1), S_M(i, j + 1, k - 1), S_R(i, j + 1, k - 1) \right\}, \]

\[ S_M(i, j, k) = \max \left\{ S_M(i - 1, j, k), S_M(i, j + 1, k), S_M(i, j, k - 1), S_L(i - 1, j, k), S_L(i, j + 1, k), S_R(i, j + 1, k), S_R(i, j, k - 1) \right\}, \]

\[ S_{\text{pseudo}}(i_0, k_0) = \max_{i_0 \leq i < j < k \leq k_0} \left\{ S_L(i, j, k), S_M(i, j, k), S_R(i, j, k) \right\}. \]
Computing optimal pseudo-knotted structures

- Let $S(i,j)$ be the opt score of a simple pseudoknotted structure.
  - Either $(I,j)$ is a simple pseudoknot
    - $S(i,j) = S_{\text{pseudo}}(i,j)$
  - Or, not
    - $S(i,j) = \max\{ v(a_i,a_j) + S(i+1,j-1), \max_{i<k<j} S(i,k-1)+S(k,j) \}$

\[
S(i,j) = \max \left\{ S_{\text{pseudo}}(i,j), \ S(i+1,j-1) + v(a_i,a_j), \ \max_{i<k<j} \{S(i,k-1)+S(k,j)\} \right\}
\]
Time Complexity

• We compute $S_{\text{pseudo}}(i,j)$ for all $(i,j)$. Each computation is $O(n^3)$. Total time?
• For each $i_0$, perform the following computation:

$$\begin{align*}
\text{for } i &= i_0 \text{ to } n - 2 \text{ do} \\
&\quad \text{for } j = n - 1 \text{ downto } i + 1 \text{ do} \\
&\quad\quad \text{for } k = j + 1 \text{ to } n \text{ do}
\end{align*}$$

• $S_{\text{pseudo}}(i_0,k_0) = \max_{i_0 \leq i < j \leq k_0} \{S_L(i,j,k_0), \ldots\}$
• Total time $O(n^4)$
Recursive pseudoknots

• Each loop of the RNA structure is a recursive pseudoknotted RNA structure.
• Optimal recursive pseudoknotted RNA structure problem can be solved in $O(n^5)$ time.
Open Questions

- There should be a direct generalization of simple pseudoknots to the following structure.
- Rivas and Eddy do consider such a generalization, but a systematic treatment is missing.
- Q: Given a pseudo-knotted structure, is it an Akutsu simple pseudoknotted structure?
  - Linear time algorithm was devised recently for this problem.
Structure as a proxy for ncRNA

- Any genomic region with an energetically favorable fold is a candidate ncRNA?
- Rivas and Eddy show otherwise.
- They use
  - LOD score \( L = \frac{\Pr(s|RNA)}{\Pr(s|Null)} \)
  - \( Z \)-score \( Z = \frac{G-\mu}{\sigma} \)
• A region of *C. elegans* is containing two tRNAs is chosen. The position of tRNA is indicated by stars.
• The true positions and the LOD-score correspond.
• Stronger results in *M. janaschii* (AT% = 70)
• Weak results in *E. coli* (GC% = 50)
Correlation with GC content

- GC content detector has results similar to structure prediction!
Significance of detected regions

- Test for significance was using Z-scores on shuffled sequences.
- Earlier tests shuffled sequences using a large window.
- Shuffling high scoring windows led to lower Z-scores.
Testing chimeric sequence

- Test on chimeric sequence. A real tRNA was embedded in 2000bp random sequence with similar GC-composition.
Increasing window size

- tRNA Z-score computation with a larger window (85nt).
• For very strong signals, a Z-score > 4 may still be significant.
Z-scores of 415 tRNA genes. 98% have Z-score lower than 4.
ncRNA gene detection using comparative sequence analysis

• Idea: It is likely for a random sequence to fold into an energetically favorable structure.

• However, it is unlikely for a random sequence to fold into a structure that is similar to a query structure.
Questions

1. Given a query ncRNA, compute an alignment that conserves both sequence, and structure.

2. Given two orthologous sequences, compute an alignment that simultaneously preserves sequence and structure in both.
Scoring RNA alignments

Score = \[ \sum_{j} \mathcal{L}(A[1,j], A[2,j]) + \sum_{(i,j),(k,l)} \mathcal{L}(i,j,k,l) \]
Binary tree representation of RNA structure
procedure alignRNA
(*S is the set of base-pairs in RNA structure of s. \(S'\) is the augmented set. *)
for all intervals \( (i, j) \), \( 1 \leq i < j \leq n \), all nodes \( v \in S' \)
if \( v \in S \)
\[
A[i, j, v] = \max \begin{cases} 
A[i + 1, j - 1, \text{child}(v)] + \delta(t[i], t[j], s[l_v], s[r_v]), \\
A[i, j - 1, v] + \gamma(\varepsilon, t[j]), \\
A[i + 1, j, v] + \gamma(\varepsilon, t[i]), \\
A[i + 1, j, \text{child}[v]] + \gamma(s[l_v], t[i]) + \gamma(s[r_v], \varepsilon), \\
A[i, j - 1, \text{child}[v]] + \gamma(s[l_v], \varepsilon) + \gamma(s[r_v], t[j]), \\
A[i, j, \text{child}[v]] + \gamma(s[l_v], \varepsilon) + \gamma(s[r_v], \varepsilon)
\end{cases}
\]
else if \( v \in S' - S \), and \( v \) has one child
\[
A[i, j, v] = \max \begin{cases} 
A[i, j - 1, \text{child}[v]] + \gamma(s[r_v], t[j]), \\
A[i, j, \text{child}[v]] + \gamma(s[r_v], \varepsilon), \\
A[i, j - 1, v] + \gamma(\varepsilon, t[j]), \\
A[i + 1, j, v] + \gamma(\varepsilon, t[i])
\end{cases}
\]
else if \( v \in S' - S \), and \( v \) has two children
\[
A[i, j, v] = \max_{i \leq k \leq j} \{A[i, k - 1, \text{left_child}[v]] + A[k, j, \text{right_child}[v]]\}
\]
end if
end for
Filtering