Hardness of predicting Pseudoknots in RNA 2nd Structure
Predicting pseudo knots is hard

Score function: Number of stacked base pairs

\[ \text{BPS}(S) = | \{ i \cdot j \in S \mid (i + 1) \cdot (j - 1) \in S \} | \]

Theorem: Given an RNA sequence \( s \) and target \( K \), it is NP-hard to determine whether there is a legal structure \( S \) of \( s \) containing only Watson-Crick and Wooble base pair) with \( \text{BPS}(S) \geq K \)
Reduction from bin-packing

Reduction from bin-packing, e.g. show that if we can solve the folding problem, then we can also solve the bin packing problem.

**Bin-packing:** Given $k$ items of sizes $a_1, a_2, \ldots, a_k$ and $B$ bins each with capacity $C$, determine if the items fit into the bins.

**Note:** Bin-packing is strongly NP-hard, i.e. it remains NP-hard even when all numbers in the input are bounded by some polynomial (www.nist.gov/dads). Also, we can assume that $a_i > 1$ for all $i$
The reduction

Given an instance of bin-packing \((a_1, \ldots, a_k, B \text{ and } C)\), we construct the following RNA sequence \(s\) and target \(K\):

\[
\begin{align*}
S &= C \ldots C AC \ldots CA \ldots AC \ldots AAA G \ldots G A G \ldots G A \ldots A C \ldots G \\
\underline{a_1} & \quad \underline{a_2} \quad \underline{a_k} \\
\text{le items of} & \quad \text{B substrings of } GC
\end{align*}
\]

Target: \(K = (a_1 + \ldots + a_k) - k\)

**Intuition:** Since \(s\) contains no U's, we can only form GC base pairs. Each such base pairs conceptually corresponds to putting a piece of an item in a bin.
The reduction

Given an instance of bin-packing \((a_1, ..., a_k, B\text{ and } C)\), we construct the following RNA sequence \(s\) and target \(K\):

\[
\begin{align*}
S &= C ... CAC ... CA ... AC ... CAAAA G ... G A G ... GA ... AC ... G \\
a_1 &\quad a_2 \quad a_k \\
\underbrace{\text{le items of}} & \quad \underbrace{\text{B substrings of } Gc} \\
\text{sizes } a_1, ..., a_k & \\
\end{align*}
\]

Since bin-packing is **strongly NP-hard**, we can assume that the numbers \(a_1, ..., a_k, B\text{ and } C\), are all polynomially bounded in the number of items \((k)\), i.e. \(|s|\) is polynomially in \(k\), and \(s\) can be constructed in polynomial time.
Proof

We will argue that we can find a structure $S$ of $s$ with $\text{BPS}(S)=K$ if and only if there is a solution to the bin-packing problem.
Proof

Part 1: If there is solution to the bin packing problem, then construct the folding where all C's in an item are paired with consecutive G's in the bin containing the item.

\[
\text{BPS}(S) = \#BP - \#\text{Helices} = (a_1 + \ldots + a_k) - k = K
\]
Proof

Part 2: If there is folding $S$ with score

$$\text{BPS}(S) = \#\text{BP} - \#\text{Helices} = K$$

Then consider two cases:

Case 1: If all C's are paired, then we must have that all C's from one item paired to G's from the same bin. Otherwise there would be too many helices. This implies a solution to the bin-packing problem.

Case 2: If not all C's are paired, then $\#\text{Helices}$ must be reduced similarly, otherwise $\text{BPS}(S)$ cannot be $K$. To reduce $\#\text{Helices}$ by 1, we must unpair an entire item. Since an item contains at least 2 C's, we cannot reduce $\#\text{Helices}$ by the number of unpaired C's. i.e. all C's must be paired.