Combinatorial RNA Design: Designability and Structure-Approximating Algorithm

Jozef Haleš[1]   Ján Maňuch[1,3]   Yann Ponty[1,2]   Ladislav Stacho[1]

[1]Simon Fraser University, Canada
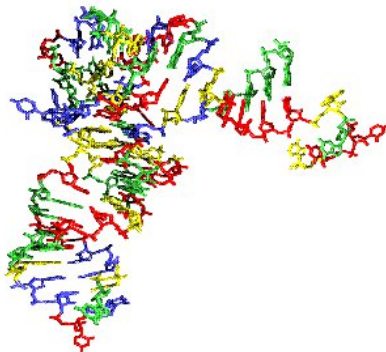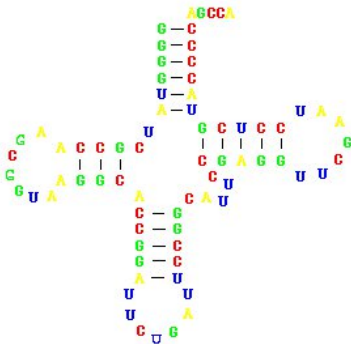[2]Pacific Institute for Mathematical Sciences, Canada
[3]University of British Columbia, Canada

CPM 2015

RNA Structures

Composed of four bases: **adenine** (A), **guanine** (G), **cytosine** (C) and **uracil** (U)
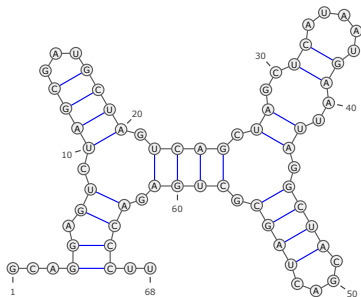
Representations of Secondary Structures

Structure is a pair (n, P), where n is the number of bases and P is a set of pairs (i, j) with $1 \leq i < j \leq n$ representing a base pair between the i-th base and the j-the base.

Representations of Secondary Structures

Structure is a pair (n, P), where n is the number of bases and P is a set of pairs (i, j) with $1 \leq i < j \leq n$ representing a base pair between the i-th base and the j-the base.
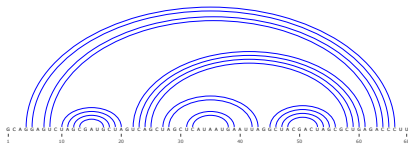

arc diagram

Representations of Secondary Structures

Structure is a pair (n, P), where n is the number of bases and P is a set of pairs (i, j) with $1 \leq i < j \leq n$ representing a base pair between the i-th base and the j-the base.
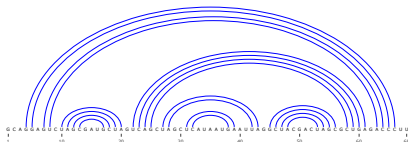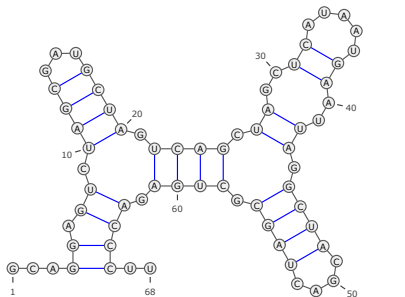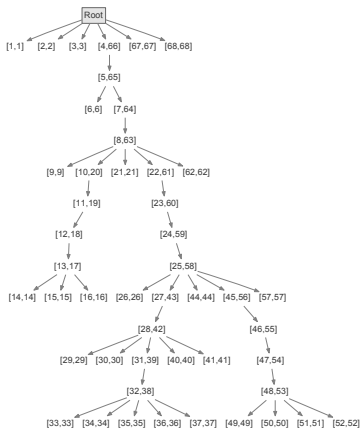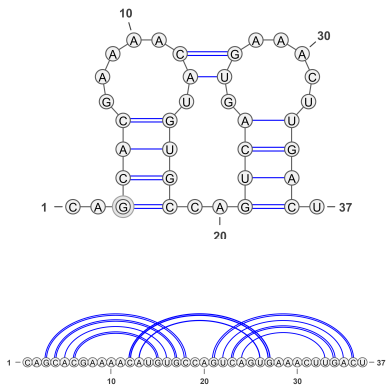
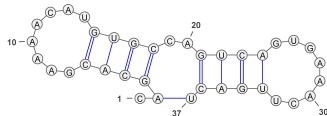

arc diagram

tree representation

Pseudoknot-Free Secondary Structures

pseudoknotted structure   pseudoknot-free structure

Pseudoknot-Free Secondary Structures
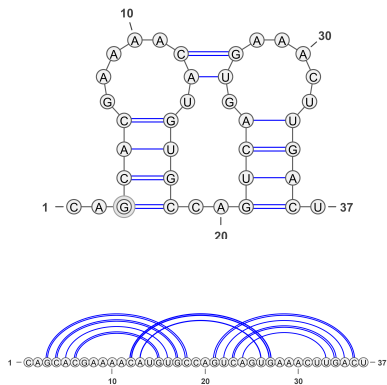
pseudoknotted structure
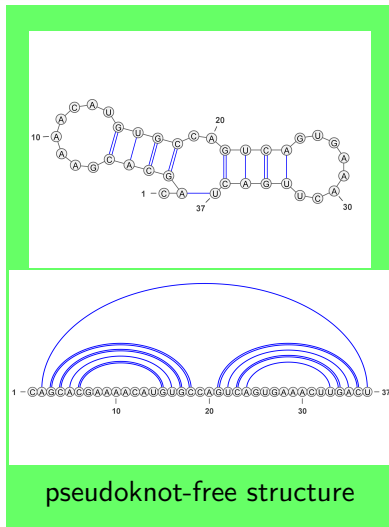
pseudoknot-free structure

Let S_n denote all pseudoknot-free structures with n bases.

RNA Folding

Let $\mathcal{M}$ be an energy model.
RNA Folding problem looks from the MFE structure(s).

Problem

RNA-FOLD$_{\mathcal{M}}$ *problem*
Input: *RNA sequence w*
Output: *set of PKF structures* $\arg\min_{S \in \mathcal{S}_{|w|}} E_{\mathcal{M}}(w, S)$.

Assuming an *additive energy model* which adds up local
contributions, finding one structure in RNA-FOLD$_{\mathcal{M}}(w)$ can be
done in time $O(n^3 / \log(n))$ using Dynamic Programming
[Nussinov, Jacobson (1980),Frid et al. (2010),etc.].

Energy Models

-

Turner model: free energy is the sum of loop energies
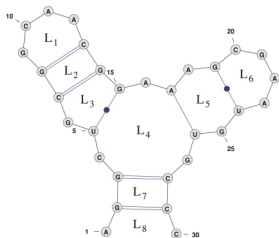
Source: [Lorenz, Clote (2011)]

Energy Models

●

Turner model: free energy is
the sum of loop energies

Source: [Lorenz, Clote (2011)]

- Simplified models:
  - **Base-pair maximization (Watson-Crick model)** $\mathcal{W}$: Count the number of Watson-Crick base pairs ($C \cdot G$ and $A \cdot U$)
  - **Base-pair sum**: Sum of energy contributions of base pairs ($\delta_B(x, x')$) — usually includes weak base pairs $G \cdot U$
  - **Stacked base-pairs**: Sum of energy contributions of consecutively nested pairs ($\delta_S(x, x', y, y')$)
  - **Nearest neighbor**

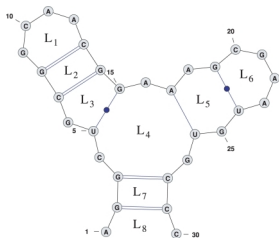# Energy Models

-

Turner model: free energy is
the sum of loop energies
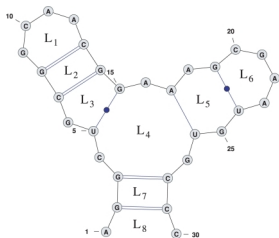
Source: [Lorenz, Clote (2011)]

- Simplified models:
  - **Base-pair maximization (Watson-Crick model)** $\mathcal{W}$: Count the number of Watson-Crick base pairs ($C \cdot G$ and $A \cdot U$)
  - **Base-pair sum**: Sum of energy contributions of base pairs ($\delta_B(x, x')$) — usually includes weak base pairs $G \cdot U$
  - **Stacked base-pairs**: Sum of energy contributions of consecutively nested pairs ($\delta_S(x, x', y, y')$)
  - **Nearest neighbor**

RNA Design Problem

Let $\mathcal{M}$ be an energy model.

Problem

RNA-DESIGN$_{\mathcal{M},\Sigma,\Delta}$ *problem*
Input: *Secondary structure S + Energy distance $\Delta > 0$*
Output: *RNA sequence $w \in \Sigma^\star$ — called a design for S — such that:*

$$\forall S' \in \mathcal{S}_{|w|} \setminus \{S\} : \ E_{\mathcal{M}}(w, S') \geq E_{\mathcal{M}}(w, S) + \Delta$$

or $\varnothing$ if no such sequence exists.

RNA Design Problem (simplified)

Simplified formulation for Watson-Crick model $\mathcal{W}$ and $\Delta = 1$:

Problem

RNA-DESIGN$_\Sigma$ *problem*
Input: *Secondary structure S*
Output: *RNA sequence $w \in \Sigma^\star$ — called a design for S — such that:*

$$\text{RNA-FOLD}_\mathcal{W}(w) = \{S\}$$

or $\varnothing$ if no such sequence exists.

RNA Design Problem (simplified)

Simplified formulation for Watson-Crick model $\mathcal{W}$ and $\Delta = 1$:

Problem

RNA-DESIGN$_\Sigma$ *problem*
Input: *Secondary structure S*
Output: *RNA sequence $w \in \Sigma^\star$ — called a design for S — such that:*

$$\text{RNA-FOLD}_{\mathcal{W}}(w) = \{S\}$$

or $\varnothing$ if no such sequence exists.

Example

a. Target sec. str. S **b.** Invalid sequence for S **c.** Design for S

((.) (. .))   G G A C A G G U C   A C A G G U U C U

RNA Design Problem (simplified)

Simplified formulation for Watson-Crick model $\mathcal{W}$ and $\Delta = 1$:

Problem

RNA-DESIGN$_\Sigma$ *problem*
Input: *Secondary structure S*
Output: *RNA sequence $w \in \Sigma^\star$ — called a design for S — such that:*
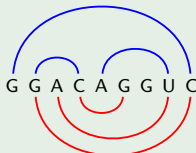
$$\text{RNA-FOLD}_{\mathcal{W}}(w) = \{S\}$$
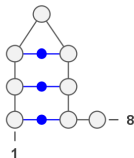
or $\varnothing$ if no such sequence exists.

Let Designable(Σ) be the set of all structures for there exists a design.

Our Results: Definitions and notations

Given a secondary structure S.

- Let Unpaired$_S$ be the set of all unpaired positions of S.

Example

Unpaired$_S = \{4, 8\}$

Our Results: Definitions and notations

Given a secondary structure S.

- Let Unpaired_S be the set of all unpaired positions of S.
- S is *saturated* if $\text{Unpaired}_S = \emptyset$.
  Let Saturated be the set of all saturated structures.
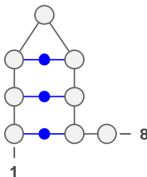
Example

not saturated

saturated

Our Results: Definitions and notations

Given a secondary structure S.
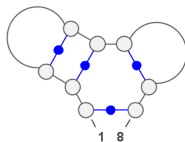
- Let Unpaired_S be the set of all unpaired positions of S.
- S is *saturated* if $\text{Unpaired}_S = \emptyset$.
  Let Saturated be the set of all saturated structures.
- Let $D(S)$ be the maximal *paired degree* of nodes in the tree representation of S. The *paired degree* is the number of nodes representing base pairs.

Example
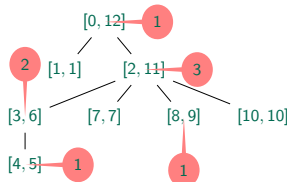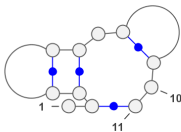
$D(S) = 3$

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable$(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

Example

$1 -\bigcirc\bigcirc\bigcirc\bigcirc\bigcirc\bigcirc- 6$

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable$(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 Designable$(\Sigma_{1,0}) = ($Saturated $\cap \{S \mid D(S) \leq 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

Example

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable$(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 Designable$(\Sigma_{1,0}) = (\text{Saturated} \cap \{S \mid D(S) \le 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 Designable$(\Sigma_{1,1}) = \{S \mid D(S) \le 2\}$.
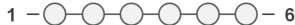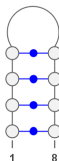
Example

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, $\text{Designable}(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 $\text{Designable}(\Sigma_{1,0}) = (\text{Saturated} \cap \{S \mid D(S) \leq 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 $\text{Designable}(\Sigma_{1,1}) = \{S \mid D(S) \leq 2\}$.

Question: Why not degree 3?

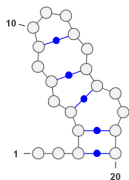## Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable($\Sigma_{0,u}$) = $\{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 Designable($\Sigma_{1,0}$) = (Saturated $\cap$ $\{S \mid D(S) \leq 2\}$) $\cup$ $\{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 Designable($\Sigma_{1,1}$) = $\{S \mid D(S) \leq 2\}$.

Question: Why not degree 3?

Proof.

In the root:

? ... ? ? ... ? ? ... ?   — we can only use $C \cdot G$ or $G \cdot C$

□

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, $\text{Designable}(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 $\text{Designable}(\Sigma_{1,0}) = (\text{Saturated} \cap \{S \mid D(S) \leq 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 $\text{Designable}(\Sigma_{1,1}) = \{S \mid D(S) \leq 2\}$.

Question: Why not degree 3?

Proof.

In the root:

C ... G G ... C C ... G — one of them has to repeat

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable($\Sigma_{0,u}$) = $\{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 Designable($\Sigma_{1,0}$) = (Saturated $\cap \{S \mid D(S) \leq 2\}$) $\cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 Designable($\Sigma_{1,1}$) = $\{S \mid D(S) \leq 2\}$.

Question: Why not degree 3?

Proof.

In the root:

C ... G G ... C C ... G — there is an alternative fold □

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable($\Sigma_{0,u}$) = $\{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 Designable($\Sigma_{1,0}$) = (Saturated $\cap$ $\{S \mid D(S) \leq 2\}$) $\cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 Designable($\Sigma_{1,1}$) = $\{S \mid D(S) \leq 2\}$.

Question: Why not degree 3?

Proof.

In an internal node:

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, Designable$(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 Designable$(\Sigma_{1,0}) = (\text{Saturated} \cap \{S \mid D(S) \le 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 Designable$(\Sigma_{1,1}) = \{S \mid D(S) \le 2\}$.

Question: Why not degree 3?

Proof.

In an internal node:

$\dots$? C $\dots$ G C $\dots$ G ? $\dots$ — either we get a repeat, or$\dots$ $\square$

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, $\text{Designable}(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;
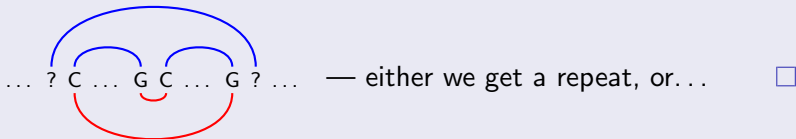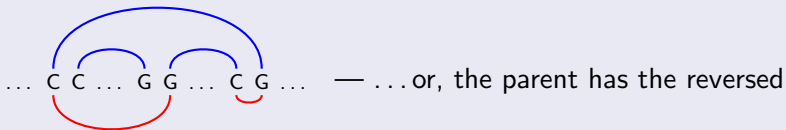
R2 $\text{Designable}(\Sigma_{1,0}) = (\text{Saturated} \cap \{S \mid D(S) \leq 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 $\text{Designable}(\Sigma_{1,1}) = \{S \mid D(S) \leq 2\}$.

Question: Why not degree 3?

Proof.

In an internal node:

... C C ... G G ... C G ... — ... or, the parent has the reversed

base pair of a child □

Our Results: Designability over Restricted Alphabets

Let $\Sigma_{c,u}$ be an alphabet with c pairs of complementary bases and u bases without a complementary base.

R1 For every $u \in \mathbb{N}^+$, $\text{Designable}(\Sigma_{0,u}) = \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R2 $\text{Designable}(\Sigma_{1,0}) = (\text{Saturated} \cap \{S \mid D(S) \leq 2\}) \cup \{(n, \emptyset) \mid \forall n \in \mathbb{N}\}$;

R3 $\text{Designable}(\Sigma_{1,1}) = \{S \mid D(S) \leq 2\}$.

This can be easily generalized to:

Lemma
For any structure S in $\text{Designable}(\Sigma_{c,u})$, $D(S) \leq 2c$.

Our Results: Designability over the Complete Alphabet
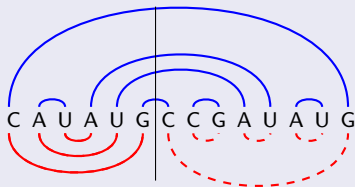
Let $\Sigma_{2,0} = \{A, U, C, G\}$.

Our Results: Designability over the Complete Alphabet

Let $\Sigma_{2,0} = \{A, U, C, G\}$.

R4 Designable$(\Sigma_{2,0}) \cap$ Saturated $= \{S \mid D(S) \leq 4\} \cap$ Saturated.

Idea.

Lemma: C A U A U G C C G A U A U G

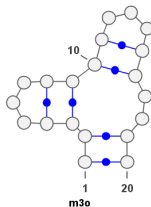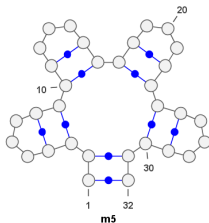Use this lemma to prove that the structure is unique by a bottom-up tree induction.

Our Results: Designability over the Complete Alphabet

Let $\Sigma_{2,0} = \{A, U, C, G\}$.

R4 Designable$(\Sigma_{2,0}) \cap$ Saturated $= \{S \mid D(S) \leq 4\} \cap$ Saturated.

When unpaired positions are allowed in the target structure, our characterization is only partial:

R5 (Necessary) If $S \in$ Designable$(\Sigma_{2,0})$, then S does not contain "*a node having degree more than four*" (motif m_5) and "*a node having one or more unpaired children, and degree greater than two*" (motif $m_{3\circ}$).

Our Results: Designability over the Complete Alphabet

Let $\Sigma_{2,0} = \{A, U, C, G\}$.

R4 $\text{Designable}(\Sigma_{2,0}) \cap \text{Saturated} = \{S \mid D(S) \leq 4\} \cap \text{Saturated}$.

When unpaired positions are allowed in the target structure, our characterization is only partial:

R5 (Necessary) If $S \in \text{Designable}(\Sigma_{2,0})$, then S does not contain "*a node having degree more than four*" (motif m_5) and "*a node having one or more unpaired children, and degree greater than two*" (motif $m_{3\circ}$).

R6 (Sufficient) Let Separated be the set of structures for which there exists a separated (proper) coloring of the tree representation, then $\text{Separated} \subset \text{Designable}(\Sigma_{2,0})$

Our Results: Separated Coloring

Consider the tree representation T_S of structure S. Color every paired node of T_S different from the root by black $(G \cdot C)$, white $(C \cdot G)$ or grey color $(A \cdot U$ or $U \cdot A)$. This coloring is called *proper* if:

1. every node has at most one black, at most one white and at most two grey children;
2. a grey node has at most one grey child;
3. a black node does not have a white child; and
4. a white node does not have a black child.

Our Results: Separated Coloring

Consider the tree representation T_S of structure S. Color every paired node of T_S different from the root by black ($G \cdot C$), white ($C \cdot G$) or grey color ($A \cdot U$ or $U \cdot A$). This coloring is called *proper* if:

1. every node has at most one black, at most one white and at most two grey children;
2. a grey node has at most one grey child;
3. a black node does not have a white child; and
4. a white node does not have a black child.

Given a proper coloring of T_S, let the *level* of each node be the number of black nodes minus the number of white nodes on the path from this node to the root.
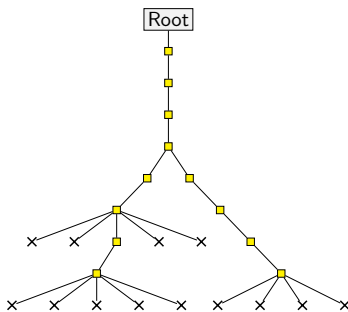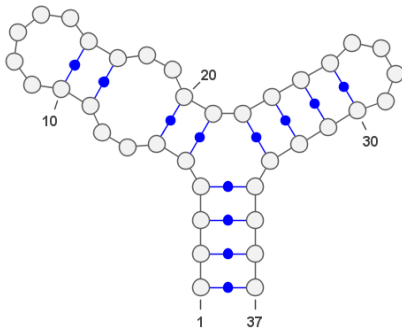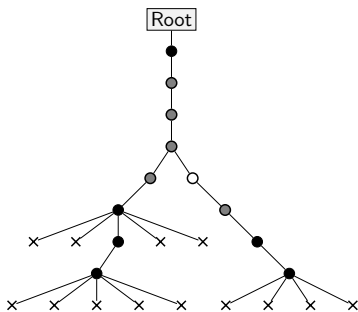
Our Results: Separated Coloring

Consider the tree representation T_S of structure S. Color every
paired node of T_S different from the root by black ($G \cdot C$), white
($C \cdot G$) or grey color ($A \cdot U$ or $U \cdot A$). This coloring is called *proper*
if:

1. every node has at most one black, at most one white and at
   most two grey children;
2. a grey node has at most one grey child;
3. a black node does not have a white child; and
4. a white node does not have a black child.

Given a proper coloring of T_S, let the *level* of each node be the
number of black nodes minus the number of white nodes on the
path from this node to the root.

A proper coloring is called *separated* if the two sets of levels,
associated with grey and unpaired nodes respectively, do not
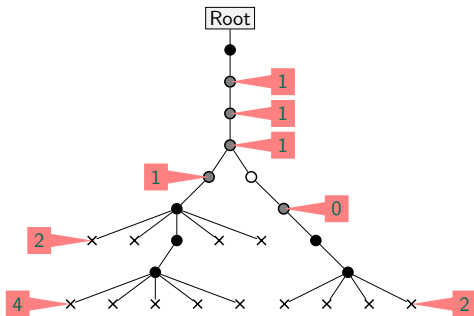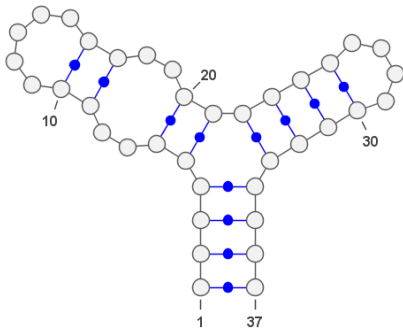intersect.

Our Results: Separated Coloring (example)
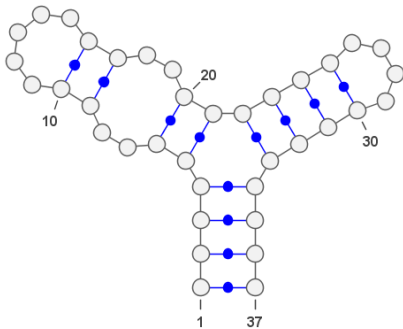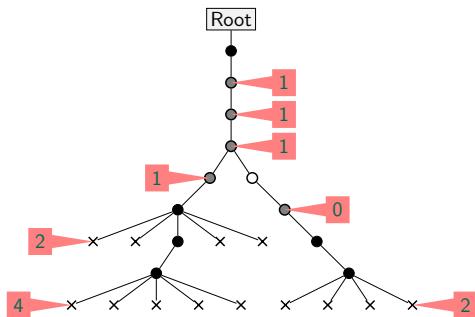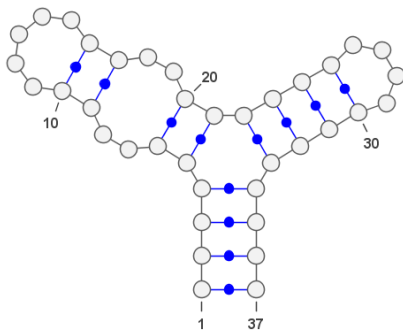
Our Results: Separated Coloring (example)

Our Results: Separated Coloring (example)
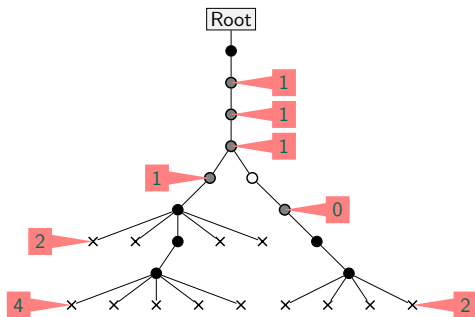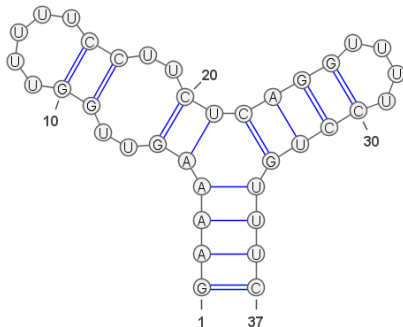
Our Results: Separated Coloring (example)

Levels of grey nodes: 0,1
Levels of leaves: 2,4
This is a separated coloring

Our Results: Separated Coloring (example)

Levels of grey nodes: 0,1

Levels of leaves: 2,4

This is a separated coloring

Design: ● → GC ○ → CG ◕ → AU|UA ✕ → U

GAAAAGUUGGUUUUUCCUUCUCAGGUUUUUCCUGUUUC

Our Results: Separated Coloring (sketch of the proof)

Let w be the sequence obtained from the separated coloring.
Let S' be an MFE fold for w.

Our Results: Separated Coloring (sketch of the proof)

Let w be the sequence obtained from the separated coloring.

Let S' be an MFE fold for w.

- In S, every C, G and A is paired.
- Hence, in S', every C, G and A must be paired.

Our Results: Separated Coloring (sketch of the proof)

Let w be the sequence obtained from the separated coloring.
Let S' be an MFE fold for w.

- In S, every C, G and A is paired.
- Hence, in S', every C, G and A must be paired.
-

Lemma

Any A · U base pair must be between positions on the same level.

Our Results: Separated Coloring (sketch of the proof)

Let w be the sequence obtained from the separated coloring.
Let S' be an MFE fold for w.

- In S, every C, G and A is paired.
- Hence, in S', every C, G and A must be paired.
-

Lemma

Any A $\cdot$ U *base pair must be between positions on the same level.*

Proof.

If not that the portion enclosed by this base pair has an imbalance in the number of C and G, hence, not all of them are base-paired, a contradiction. $\qquad\square$

Our Results: Separated Coloring (sketch of the proof)

Let w be the sequence obtained from the separated coloring.
Let S' be an MFE fold for w.

- In S, every C, G and A is paired.
- Hence, in S', every C, G and A must be paired.
-

Lemma

Any A $\cdot$ U *base pair must be between positions on the same level.*

Proof.

If not that the portion enclosed by this base pair has an imbalance in the number of C and G, hence, not all of them are base-paired, a contradiction. $\qquad\square$

- All U's unpaired in S, must be also unpaired in S'.

Our Results: Separated Coloring (sketch of the proof)

Let w be the sequence obtained from the separated coloring.
Let S' be an MFE fold for w.

- In S, every C, G and A is paired.
- Hence, in S', every C, G and A must be paired.
-

Lemma

Any A · U *base pair must be between positions on the same level.*

Proof.

If not that the portion enclosed by this base pair has an imbalance in the number of C and G, hence, not all of them are base-paired, a contradiction. □

- All U's unpaired in S, must be also unpaired in S'.
- The claim follows by the result **R4** (for saturated structures).

Our Results: Designability over the complete alphabet

Let $\Sigma_{2,0} = \{A, U, C, G\}$.

R4 Designable$(\Sigma_{2,0}) \cap$ Saturated $= \{S \mid D(S) \leq 4\} \cap$ Saturated.

When unpaired positions are allowed in the target structure, our characterization is only partial:

R5 (Necessary) If $S \in$ Designable$(\Sigma_{2,0})$, then S does not contain "*a node having degree more than four*" (motif m_5) and "*a node having one or more unpaired children, and degree greater than two*" (motif $m_{3\circ}$).

R6 (Sufficient) Let Separated be the set of structures for which there exists a separated (proper) coloring of the tree representation, then Separated $\subset$ Designable$(\Sigma_{2,0})$

Our Results: Designability over the complete alphabet

Let $\Sigma_{2,0} = \{A, U, C, G\}$.

**R4** $\text{Designable}(\Sigma_{2,0}) \cap \text{Saturated} = \{S \mid D(S) \leq 4\} \cap \text{Saturated}$.

R5 (Necessary) If $S \in \text{Designable}(\Sigma_{2,0})$, then S does not contain "*a node having degree more than four*" (motif m_5) and "*a node having one or more unpaired children, and degree greater than two*" (motif $m_{3\circ}$).

R6 (Sufficient) Let Separated be the set of structures for which there exists a separated (proper) coloring of the tree representation, then $\text{Separated} \subset \text{Designable}(\Sigma_{2,0})$
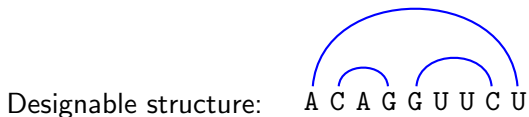
R7 If $S \in \text{Designable}(\Sigma_{2,0})$, then k-stutter $S^{[k]} \in \text{Designable}(\Sigma_{2,0})$.

Our Results: *k*-Stutter (example)
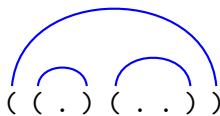
Designable structure: ((.) (. .))

Our Results: k-Stutter (example)

Designable structure:

A C A G G U U C U
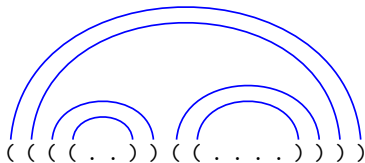
Then 2-stutter is designable as well:

Our Results: k-Stutter (example)

Designable structure: A C A G G U U C U

Then 2-stutter is designable as well:

((((. .)) ((. . . .))))

Our Results: k-Stutter (example)

Designable structure:

Then 2-stutter is designable as well:

Our Results: k-Stutter (example)
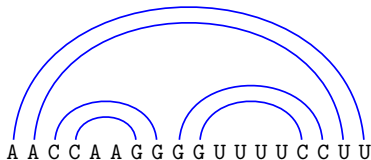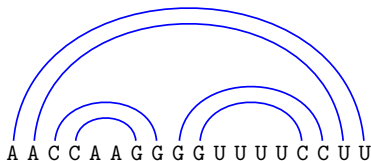
Designable structure:

A C A G G U U C U

Then 2-stutter is designable as well:

A A C C A A G G G G U U U U C C U U

Proof idea: Use König's Theorem (size of max. matching = size of min. vertex cover) to show that an MFE structure of the stutter sequence can't connect a region to two different regions.
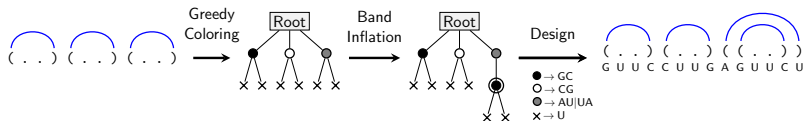
Our Results: Structure-Approximating Algorithm

R8 Any structure S without m_5 and $m_{3\circ}$ can be transformed in $\Theta(n)$ time into a $\Sigma_{2,0}$-designable structure S', by inflating a subset of its base pairs (at most one per band) so that the greedy coloring of the resulting structure is separated.

Our Results: Structure-Approximating Algorithm

R8 Any structure S without m_5 and $m_{3\,\circ}$ can be transformed in $\Theta(n)$ time into a $\Sigma_{2,0}$-designable structure S', by inflating a subset of its base pairs (at most one per band) so that the greedy coloring of the resulting structure is separated.
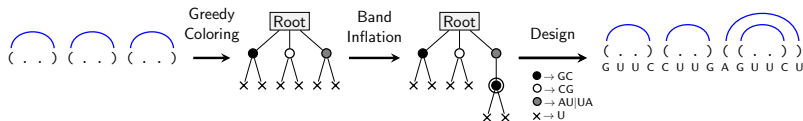
Our Results: Structure-Approximating Algorithm

R8 Any structure S without m_5 and $m_{3\circ}$ can be transformed in $\Theta(n)$ time into a $\Sigma_{2,0}$-designable structure S', by inflating a subset of its base pairs (at most one per band) so that the greedy coloring of the resulting structure is separated.

The main idea: Use inflating to separate grey vertices and leaves to odd/even levels.

Our Results: Structure-Approximating Algorithm

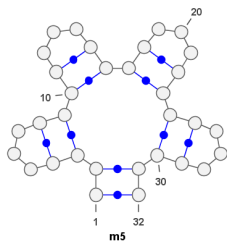**R8** Any structure S without m_5 and $m_{3\circ}$ can be transformed in
$\Theta(n)$ time into a $\Sigma_{2,0}$-designable structure S', by inflating a
subset of its base pairs (at most one per band) so that the
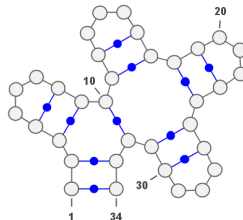greedy coloring of the resulting structure is separated.
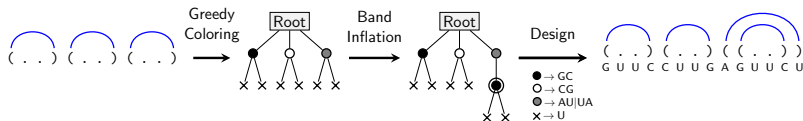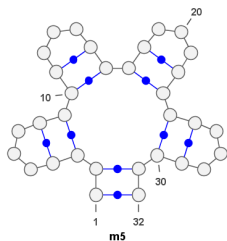

The main idea: Use inflating to separate grey vertices and leaves
to odd/even levels.

Remark: Arcs could be added to remove motifs m_5 and $m_{3\circ}$
(after which the algorithm could be applied).

Remark: Breaking motifs

m5

Remark: Breaking motifs

Open Problems and Future Work

1. What's the complexity of RNA-DESIGN problem? Could it be polynomial?

Open Problems and Future Work

1. What's the complexity of RNA-DESIGN problem? Could it be polynomial?

2. What's the complexity of RNA-DESIGN problem restricted to designs that use only one base for all unpaired position?

   (.) (.) (.)
   G U C C A G A G U

Open Problems and Future Work

1. What's the complexity of RNA-DESIGN problem? Could it be polynomial?

2. What's the complexity of RNA-DESIGN problem restricted to designs that use only one base for all unpaired position?

3. What's the complexity of determining if a structure has a separated coloring?

Open Problems and Future Work

1. What's the complexity of RNA-DESIGN problem? Could it be polynomial?

2. What's the complexity of RNA-DESIGN problem restricted to designs that use only one base for all unpaired position?

3. What's the complexity of determining if a structure has a separated coloring?

4. Extend the results to more complex energy models.
   Our results hold for the **Base-pair sum** model, as long as $-\delta_B(G, U)$ is smaller than $-\delta_B(C, G)$ and $-\delta_B(A, U)$.

Open Problems and Future Work

1. What's the complexity of RNA-DESIGN problem? Could it be polynomial?

2. What's the complexity of RNA-DESIGN problem restricted to designs that use only one base for all unpaired position?

3. What's the complexity of determining if a structure has a separated coloring?

4. Extend the results to more complex energy models.
   Our results hold for the **Base-pair sum** model, as long as $-\delta_B(G, U)$ is smaller than $-\delta_B(C, G)$ and $-\delta_B(A, U)$.

5. Find a better bound on the number of arcs that need to be inflated in our approximation algorithm.