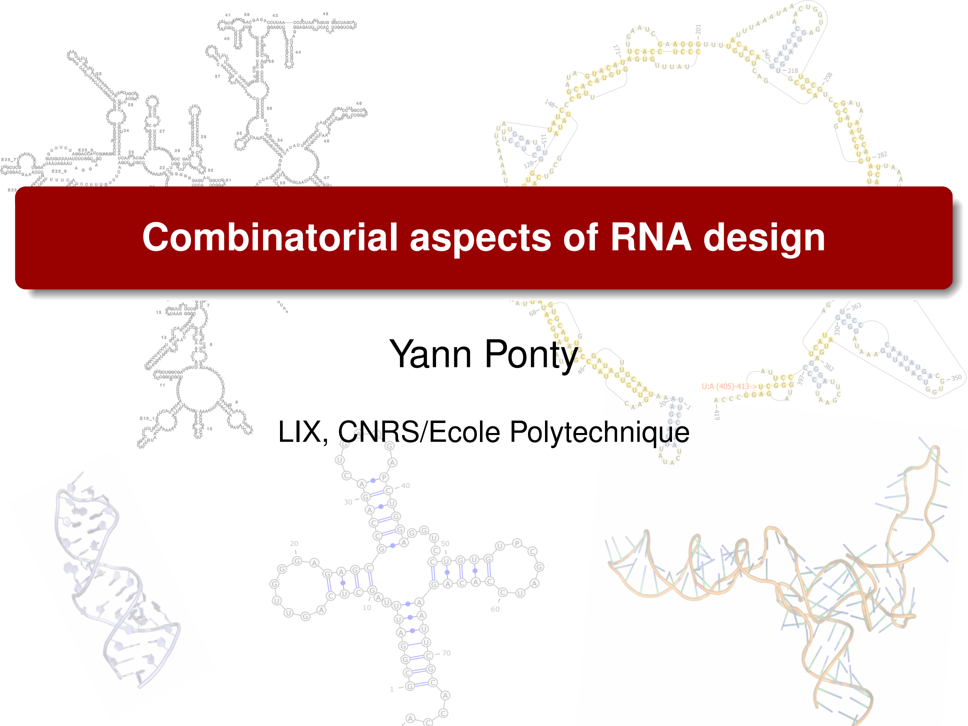
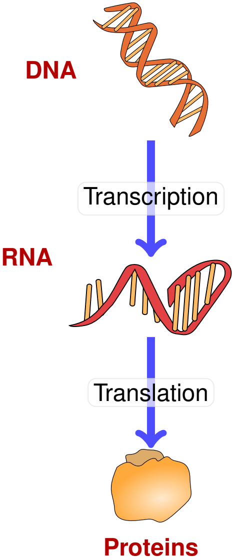


Combinatorial aspects of RNA design

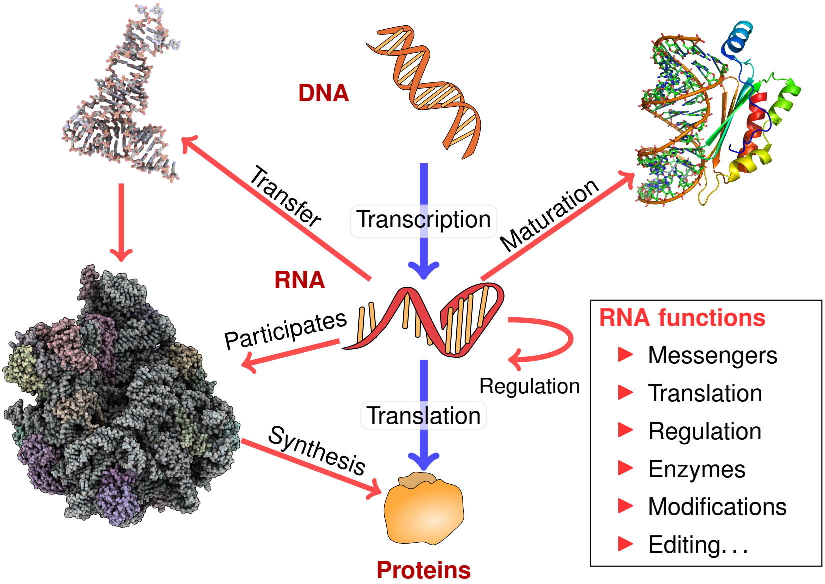
Yann Ponty

LIX, CNRS/Ecole Polytechnique

Fundamental dogma of molecular biology



Fundamental dogma of molecular biology (v2.0)



DNA

Transfer

Transcription

Maturation

RNA

Participates

Regulation

Translation

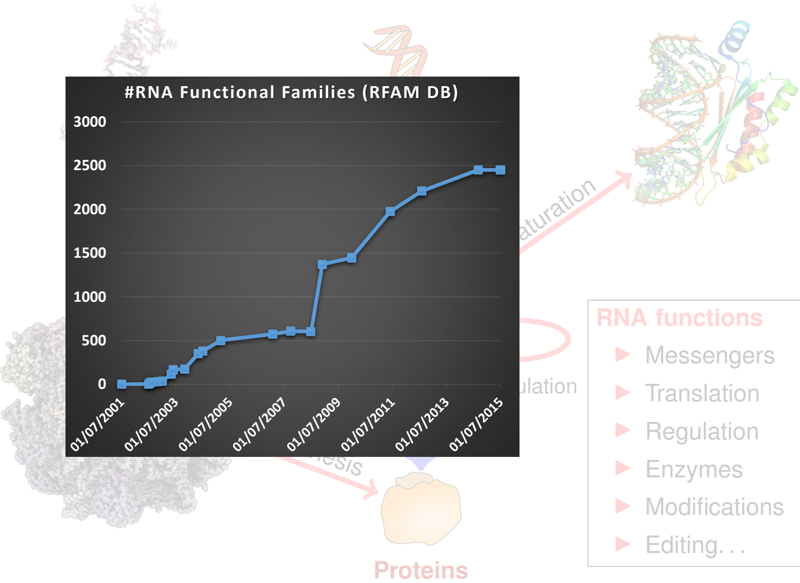
Synthesis

Proteins

RNA functions

- Messengers
- Translation
- Regulation
- Enzymes
- Modifications
- Editing. . .

Fundamental dogma of molecular biology (v2.0)



RNA functions

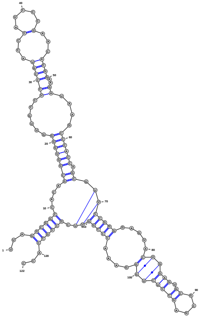
- Messengers
- Translation
- Regulation
- Enzymes
- Modifications
- Editing. . .

RNA structure(s)

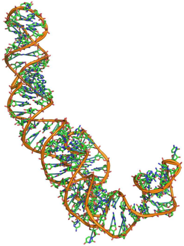
RNA = Linear Polymer = Nucleotides sequence $w \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{U}\}^\star$

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA
CC

Primary struct.



Secondary (2D) struct.



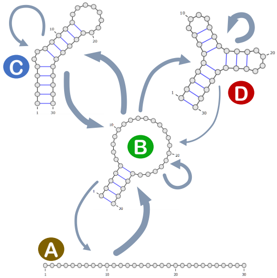
Tertiary (≈ 3D) struct.

Source: 5s rRNA (PDBID: 1K73:B)

Secondary structure S = Set of **base-pairs** $(i, j) \in [1, n]^2$ such that:

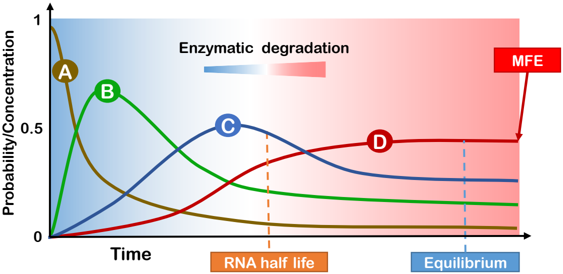
- **Monogamy:** Each position $x \in [1, n]$ involved in **at most** one base-pair
- **No crossing base-pairs:** $\forall (i, j) \in S, \nexists (k, l) \in S$ such that $i < k < j < l$
- **Steric constraints:** $\forall (i, j) \in S, |i - j| > \theta$ $(\theta = 1$ or $3)$
- **Valid base pairs:** $\forall (i, j) \in S, \{w_i, w_j\}$ is either $\{\mathrm{G}, \mathrm{C}\}$,$\{\mathrm{A}, \mathrm{U}\}$, or $\{\mathrm{G}, \mathrm{U}\}$

Paradigms in RNA structural bioinformatics



A – Kinetic Landscape

Continuous-time Markov chain



B – Evolution of concentrations

Given **free-energy** $E : \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{U}\}^\star \times \mathcal{S} \to \mathbb{R}$, at the Boltzmann equilibrium one has:

$$\mathbb{P}(S \mid w) \propto e^{-E(w,S)/RT}$$

- **Minimum Free-Energy (MFE):** Relevant structure = Most stable/probable
- **Partition function:** Equilibrium properties (stationary distribution)
- **Kinetics:** Finite-time dynamics of concentrations/probabilities

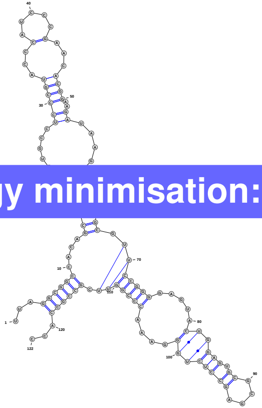
RNA sequence and structure(s)

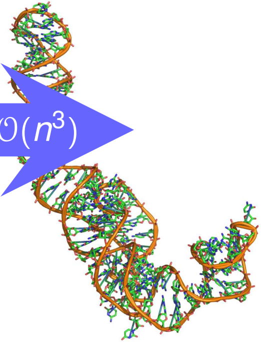
RNA = Linear Polymer = Sequence over $\{A, C, G, U\}^\star$

Energy minimisation: $\mathcal{O}(n^3)$

```
UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGGAAA
CCCGGUUCGCCGCCA

CC
```





Primary Structure

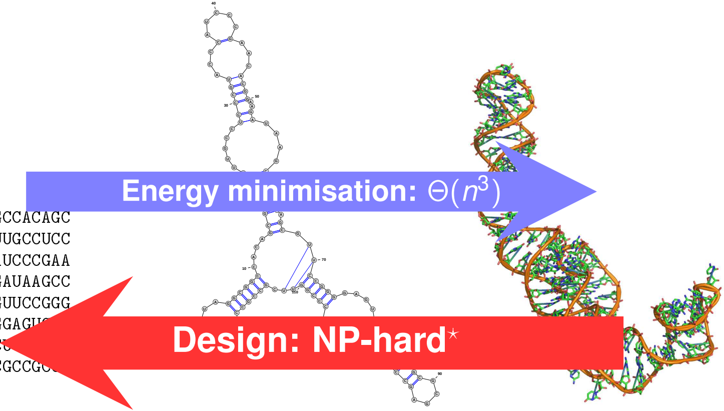
Secondary Structure

Tertiary Structure

5s rRNA 5s (PDBID: 1K73:B)

RNA sequence and structure(s)

RNA = Linear Polymer = Sequence over $\{A, C, G, U\}^{\star}$



Energy minimisation: $\Theta(n^3)$

Design: NP-hard*

UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAAGCC
CACCAGCGUUCCGGG
GAGUACUGGAGU
CGAGCCUCU
CCCGGUUCGCCGC

CC

Primary Structure Secondary Structure Tertiary Structure

5s rRNA 5s (PDBID: 1K73:B)

*Finally! **[Bonnet/Rzążewski/Sikora, RECOMB'18]**

Why would we design RNAs?

- To create building blocks for synthetic systems
  Rationally-designed RNAs increase orthogonality
- To assess the significance of observed phenomenon
  Random models should include every established characters…
  …including adoption of a single structure
- To test/push our understanding of how RNA folds
  Misfolding RNAs reveal gaps in our energy models and descriptors for the conformational spaces
- To help search for homologous sequences
  Incomplete covariance models hindered by limited training sets
  Design can be used to generalize existing alignments
- To fuel RNA-based therapeutics
  Sequence-based (siRNA, synthetic genes), but structure matters
- To perform controlled experiments

Why would we design RNAs?

- To create building blocks for synthetic systems
  Rationally-designed RNAs increase orthogonality
- To assess the significance of observed phenomenon
  Random models should include every established characters...
  ...including adoption of a single structure
- To test/push our understanding of how RNA folds
  Misfolding RNAs reveal gaps in our energy models and descriptors for the conformational spaces
- To help search for homologous sequences
  Incomplete covariance models hindered by limited training sets
  Design can be used to generalize existing alignments
- To fuel RNA-based therapeutics
  Sequence-based (siRNA, synthetic genes), but structure matters
- To perform controlled experiments

Why would we design RNAs?

- To create building blocks for synthetic systems
  Rationally-designed RNAs increase orthogonality
- To assess the significance of observed phenomenon
  Random models should include every established characters…
  …including adoption of a single structure
- To test/push our understanding of how RNA folds
  Misfolding RNAs reveal gaps in our energy models and descriptors for the conformational spaces
- To help search for homologous sequences
  Incomplete covariance models hindered by limited training sets
  Design can be used to generalize existing alignments
- To fuel RNA-based therapeutics
  Sequence-based (siRNA, synthetic genes), but structure matters
- To perform controlled experiments

Why would we design RNAs?

- To create building blocks for synthetic systems
  Rationally-designed RNAs increase orthogonality
- To assess the significance of observed phenomenon
  Random models should include every established characters. . .
  . . . including adoption of a single structure
- To test/push our understanding of how RNA folds
  Misfolding RNAs reveal gaps in our energy models and descriptors for the conformational spaces
- To help search for homologous sequences
  Incomplete covariance models hindered by limited training sets
  Design can be used to generalize existing alignments
- To fuel RNA-based therapeutics
  Sequence-based (siRNA, synthetic genes), but structure matters
- To perform controlled experiments

Why would we design RNAs?

- To create building blocks for synthetic systems
  Rationally-designed RNAs increase orthogonality
- To assess the significance of observed phenomenon
  Random models should include every established characters. . .
  . . . including adoption of a single structure
- To test/push our understanding of how RNA folds
  Misfolding RNAs reveal gaps in our energy models and descriptors for the conformational spaces
- To help search for homologous sequences
  Incomplete covariance models hindered by limited training sets
  Design can be used to generalize existing alignments
- To fuel RNA-based therapeutics
  Sequence-based (siRNA, synthetic genes), but structure matters
- To perform controlled experiments

Why would we design RNAs?

- To create building blocks for synthetic systems
  Rationally-designed RNAs increase orthogonality

- To assess the significance of observed phenomenon
  Random models should include every established characters. . .
  . . . including adoption of a single structure

- To test/push our understanding of how RNA folds
  Misfolding RNAs reveal gaps in our energy models and descriptors for the conformational spaces

- To help search for homologous sequences
  Incomplete covariance models hindered by limited training sets
  Design can be used to generalize existing alignments

- To fuel RNA-based therapeutics
  Sequence-based (siRNA, synthetic genes), but structure matters

- To perform controlled experiments

Design stories

The Nobel Prize in Physiology or Medicine 2006

Photo: L. Cicero

Andrew Z. Fire

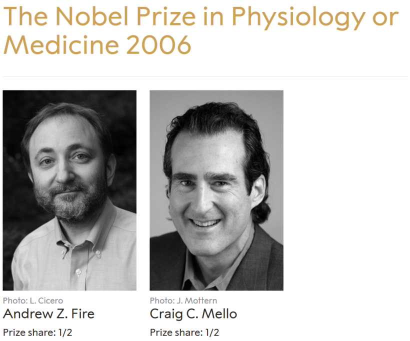
Prize share: 1/2

Photo: J. Mottern

Craig C. Mello

Prize share: 1/2

Design stories



The Nobel Prize in Physiology or Medicine 2006

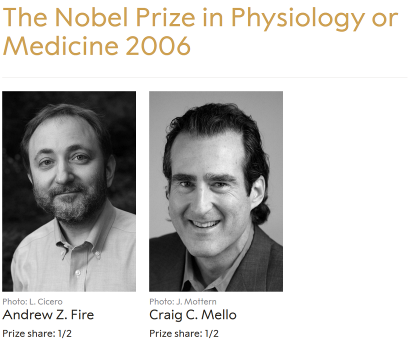
Photo: L. Cicero
Andrew Z. Fire
Prize share: 1/2

Photo: J. Mottern
Craig C. Mello
Prize share: 1/2

siRNA treatments

3 FDA-approved since 2018

Design stories



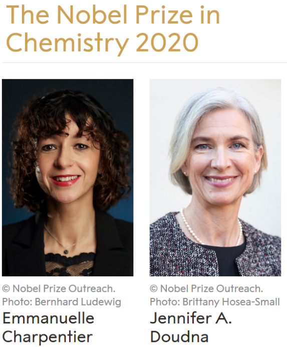
The Nobel Prize in Physiology or Medicine 2006

Photo: L. Cicero
Andrew Z. Fire
Prize share: 1/2

Photo: J. Mottern
Craig C. Mello
Prize share: 1/2

siRNA treatments

3 FDA-approved since 2018



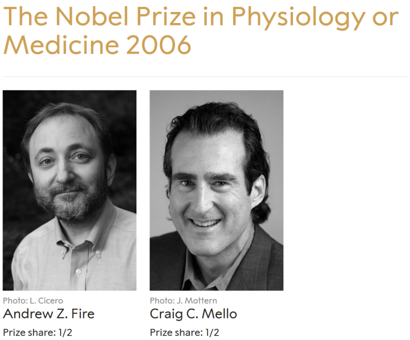
The Nobel Prize in Chemistry 2020

© Nobel Prize Outreach.
Photo: Bernhard Ludewig
Emmanuelle Charpentier

© Nobel Prize Outreach.
Photo: Brittany Hosea-Small
Jennifer A. Doudna

CRISPR/Cas9 Genome editing. . .
. . . powered by gRNAs

Design stories



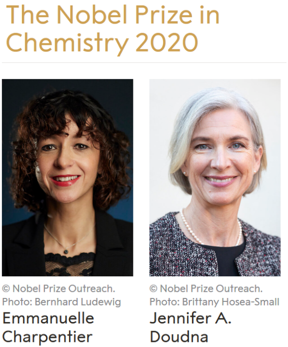
The Nobel Prize in Physiology or Medicine 2006

Photo: L. Cicero
Andrew Z. Fire
Prize share: 1/2

Photo: J. Mottern
Craig C. Mello
Prize share: 1/2

siRNA treatments

3 FDA-approved since 2018

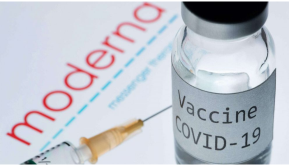


The Nobel Prize in Chemistry 2020

© Nobel Prize Outreach.
Photo: Bernhard Ludewig
Emmanuelle Charpentier

© Nobel Prize Outreach.
Photo: Brittany Hosea-Small
Jennifer A. Doudna

CRISPR/Cas9 Genome editing. . .
. . . powered by gRNAs



mRNA-based vaccines (SARS-Cov2)

Goal of design → Function

Goal: Achieve a predefined biological function (as abstracted by a model)

Goal of positive design

Compatibility with a model of function

In practice: Optimize interaction affinity or stability, constrained sequence composition. . .

Goal of negative design

To avoid unwanted functions

In practice: Avoid off-target interactions, more stable alternative structures, kinetic traps. . . (inverse combinatorial problems)

In the context of RNA:

- **Positive design:** Seq/struct comparison, composition, +/- motifs, energie(s)
  → Random generation, CSP
- **Negative design:** Target structure → Minimum Free-Energy + Boltzmann prob ↗
  → Local search, exp algorithms, black magic (heuristics, ⋆NN, crowdsourcing. . .)

Existing approaches for negative design

Based on local search. . .

- RNAInverse - TBI Vienna
- Info-RNA - Backofen@Freiburg
- RNA-SSD - Condon@UBC
- (Inca)RNAFBinv - Barash@BGU
- NUPack - Pierce@Caltech

. . . bio-inspired algorithms. . .

- FRNAKenstein - Hein@Oxford
- AntaRNA - Backofen@Freiburg
- ERD - Ganjtabesh@Tehran

. . . exact approaches. . .

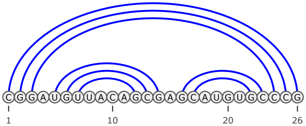
- RNAIFold - Clote@Boston College
- CO4 - Will@Leipzig

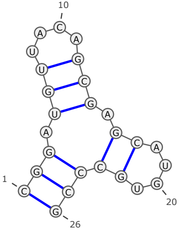
Typical issues:

- Naive initialization strategies
- Synthesized sequences do not necessarily fold properly (kinetics)
- Overly GC-rich sequences
- No negative results

$\Rightarrow$ **Combinatorial foundations!**

Energy model





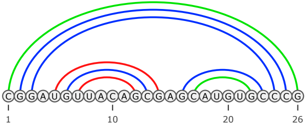
This talk: Restriction to **valid** base-pairs = $\{(A, U), (G, C), (G, U)\}$

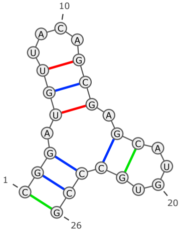
- **RNA structure** R**:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**

  Motif → Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$

  Free-energy $E(S, R)$**:** Sum of energies for motifs in R, given sequence S

Energy model

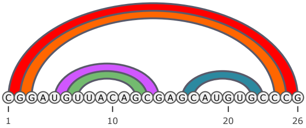


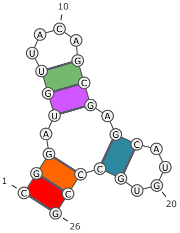


This talk: Restriction to **valid** base-pairs = $\{(A, U), (G, C), (G, U)\}$

- **RNA structure *R*:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**
  Motif → Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$
  Free-energy $E(S, R)$: Sum of energies for motifs in R, given sequence S

Energy model

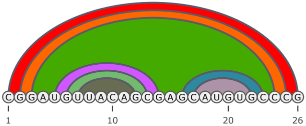


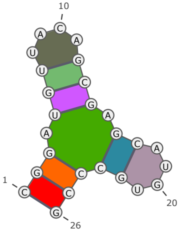


This talk: Restriction to **valid** base-pairs = $\{(A, U), (G, C), (G, U)\}$

- **RNA structure R:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**
  Motif → Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$
  Free-energy $E(S, R)$: Sum of energies for motifs in R, given sequence S

Energy model

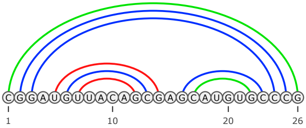


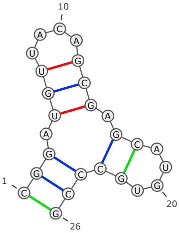


This talk: Restriction to **valid** base-pairs = $\{(A, U), (G, C), (G, U)\}$

- **RNA structure R:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**
  Motif → Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$
  Free-energy $E(S, R)$: Sum of energies for motifs in R, given sequence S

Energy model



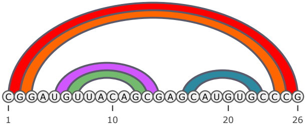


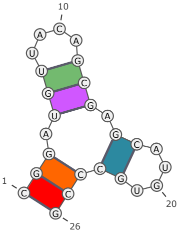
This talk: Restriction to **valid** base-pairs = $\{(A, U), (G, C), (G, U)\}$

- **RNA structure** R**:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**
  Motif $\rightarrow$ Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$
  Free-energy $E(S, R)$**:** Sum of energies for motifs in R, given sequence S

$$E_R = 2 \cdot \Delta\begin{pmatrix} \text{U} \\ | \\ \text{G} \end{pmatrix} + 4 \cdot \Delta\begin{pmatrix} \text{G} \\ | \\ \text{C} \end{pmatrix} + 2 \cdot \Delta\begin{pmatrix} \text{C} \\ | \\ \text{G} \end{pmatrix}$$

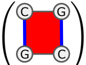
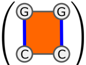
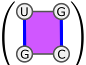
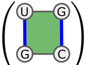
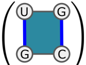
Energy model



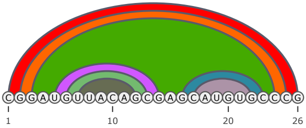


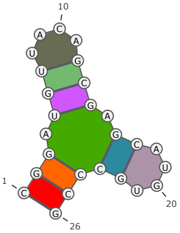
This talk: Restriction to **valid** base-pairs = $\{(A, U), (G, C), (G, U)\}$

- **RNA structure R:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**
  Motif $\rightarrow$ Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$
  Free-energy $E(S, R)$: Sum of energies for motifs in R, given sequence S

$E_R = \Delta$ () $+ \Delta$ () $+ \Delta$ () $+ \Delta$ () $+ \Delta$ ()

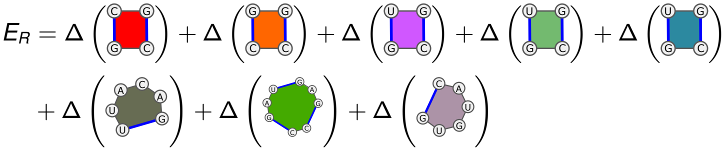
Energy model





This talk: Restriction to **valid** base-pairs = $\{(A,U),(G,C),(G,U)\}$

- **RNA structure R:** Set of base pairs (BPs)
- **Motifs:** Connected positions + content (e.g. Base Pairs, Stacking, Loops. . .)
- **Energy model:**
  Motif → Free-energy contribution $\Delta(m, a) \in \mathbb{R} \cup \{+\infty\}$, $m \subset [1, n]$, $a \in \Sigma^{|m|}$
  Free-energy $E(S, R)$: Sum of energies for motifs in R, given sequence S



RNA Inverse Folding

Definition (INVERSE-FOLDING(E) problem)

Input: Secondary structure R + Energy distance $\Delta > 0$.
Output: RNA sequence $S \in \Sigma^\star$ such that:

$$\forall R' \in \mathcal{S}_{|S|} \setminus \{R\} : \; E(S, R') \geq E(S, R) + \Delta$$

or $\varnothing$ if no such sequence exists.

Difficult problem: Probably no **obvious** DP decomposition

- NP-hard problem **[Bonnet *et al*, RECOMB'18]**... after almost 30 years!
- Existing algorithms: Heuristics or Exponential-time
- **Reason(s):** Non locality, no theoretical framework, too many parameters...

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable $\Leftrightarrow$ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable $\Rightarrow$ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]

Designability in simple BP-based energy models

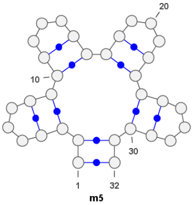
Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable ⇔ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable ⇒ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

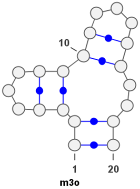
  Theorem: Similar motifs exist for any **energy model** and **design criterion**

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

[Yao *et al*, ACM-BCB'19]



m_5



$m_{3\circ}$

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable $\Leftrightarrow$ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable $\Rightarrow$ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- $\exists$ **Separated** coloring for structure $\Rightarrow$ Designable (+ $\Theta(n)$ algo.)

  Base pairs $\rightarrow$ 3 colors: ● $\rightarrow$ G · C; ○ $\rightarrow$ C · G; ◍ $\rightarrow$ A · U or U · A.

  Coloring rules: Within each loop, #● ≤ 1, #○ ≤ 1, #◍ ≤ 2 **and** #● + #○ < 2

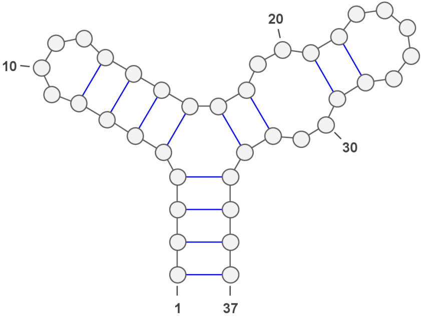
  Level of a base pair = #● − #○ on path to root.

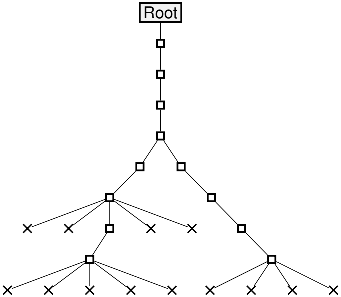
  Separated coloring = ◍ and unpaired positions occur at **different** levels

Separated Coloring (example)

Base pairs → 3 colors: ● → G · C; ○ → C · G; ◍ → A · U or U · A.

Coloring rules: Within each loop, #● ≤ 1, #○ ≤ 1, #◍ ≤ 2 **and** #● + #○ < 2

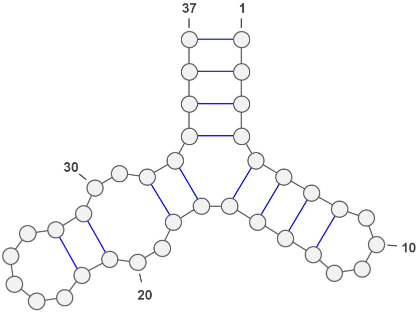


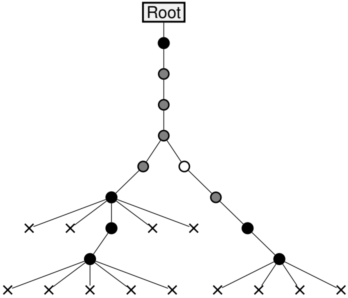


Separated Coloring (example)

Base pairs → 3 colors: ● → G · C; ○ → C · G; ◍ → A · U or U · A.

Coloring rules: Within each loop, #● ≤ 1, #○ ≤ 1, #◍ ≤ 2 **and** #● + #○ < 2

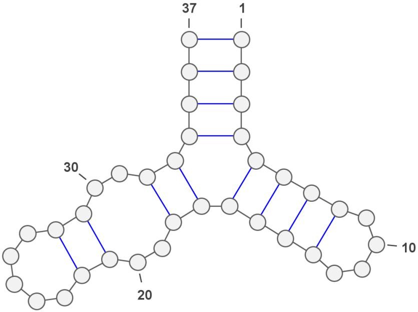


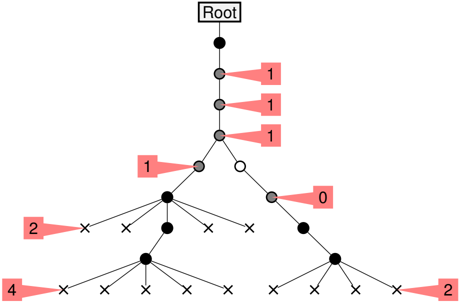


Separated Coloring (example)

Base pairs → 3 colors: ● → G · C; ○ → C · G; ◉ → A · U or U · A.

Coloring rules: Within each loop, #● ≤ 1, #○ ≤ 1, #◉ ≤ 2 **and** #● + #○ < 2

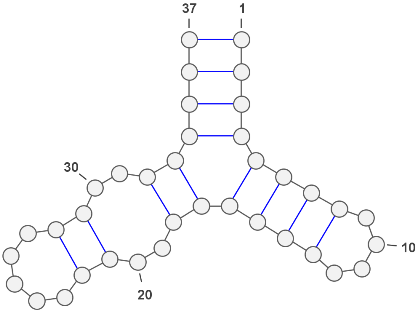


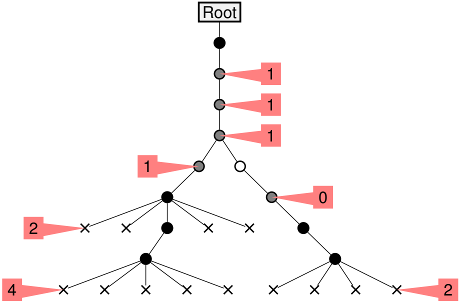


Separated Coloring (example)

Base pairs → 3 colors: ● → G · C; ○ → C · G; ◍ → A · U or U · A.

Coloring rules: Within each loop, #● ≤ 1, #○ ≤ 1, #◍ ≤ 2 **and** #● + #○ < 2



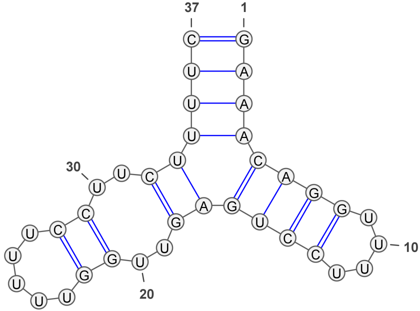


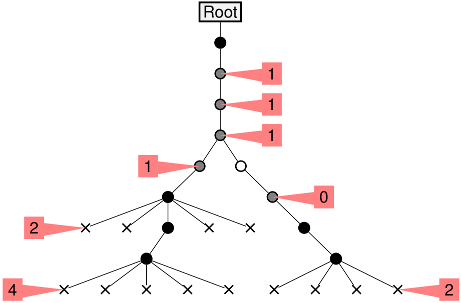
Levels of ◍: {0, 1} + Levels of unpaired/leaves: {2, 4} ⇒ Coloring is **separated**

Separated Coloring (example)

Base pairs → 3 colors: ● → G · C; ○ → C · G; ◍ → A · U or U · A.

Coloring rules: Within each loop, #● $\leq$ 1, #○ $\leq$ 1, #◍ $\leq$ 2 **and** #● + #○ $<$ 2





Levels of ◍: {0, 1} + Levels of unpaired/leaves: {2, 4} ⇒ Coloring is **separated**

Design: GAAAAGUUGGUUUUCCUUCUCAGGUUUUCCUGUUUC

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable ⇔ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable ⇒ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- ∃ **Separated** coloring for structure ⇒ Designable (+ $\Theta(n)$ algo.)

  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable ⇔ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable ⇒ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- $\exists$ **Separated** coloring for structure ⇒ Designable (+ $\Theta(n)$ algo.)

  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable ⇔ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable ⇒ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- ∃ **Separated** coloring for structure ⇒ Designable (+ $\Theta(n)$ algo.)

  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable ⇔ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable ⇒ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- $\exists$ **Separated** coloring for structure ⇒ Designable (+ $\Theta(n)$ algo.)

  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable $\Leftrightarrow$ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable $\Rightarrow$ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- $\exists$ **Separated** coloring for structure $\Rightarrow$ Designable (+ $\Theta(n)$ algo.)

  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable $\Leftrightarrow$ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable $\Rightarrow$ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- $\exists$ **Separated** coloring for structure $\Rightarrow$ Designable (+ $\Theta(n)$ algo.)

  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

Designability in simple BP-based energy models

Partial characterization of **designable** structures **[Hales *et al*, CPM'15+Algorithmica'17]**

- **Saturated structures:** Designable $\Leftrightarrow$ Degree of multiloops ≤ 4 (+ $\Theta(n)$ algo.)
- Designable $\Rightarrow$ No multiloop of *degree* ≥ 5 (m_5 motif), or *degree* ≥ 3 *with* ≥ 1 *unpaired base(s)* ($m_{3\circ}$ motif).

  Corollary: Only an **exponentially small** (on n) fraction of structs is designable

  [Yao *et al*, ACM-BCB'19]
- $\exists$ **Separated** coloring for structure $\Rightarrow$ Designable (+ $\Theta(n)$ algo.)

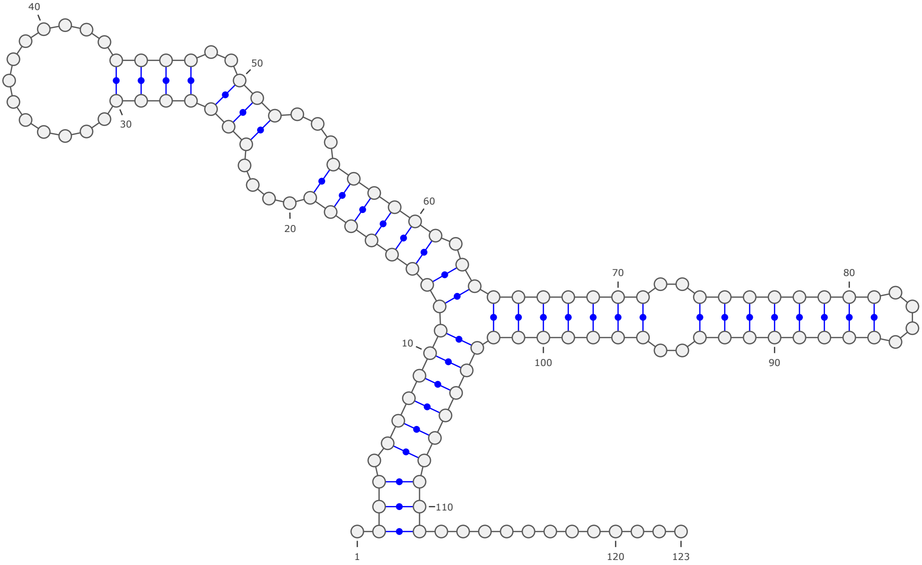
  Corollary: Approximate design for any structure avoiding m_5 and $m_{3\circ}$ in $\Theta(n)$ time

  Idea: Insert new BPs on helices to **offset** unpaired/leaves and ●

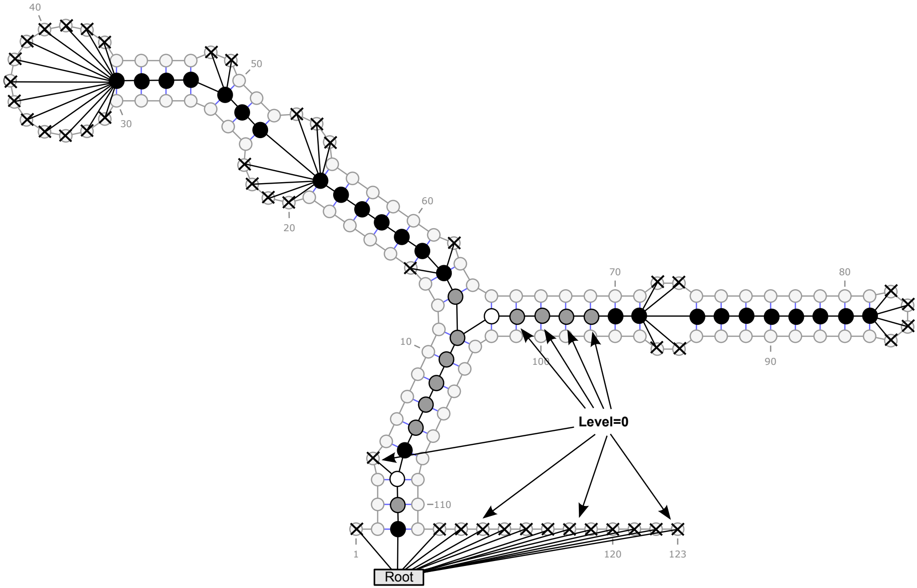
Open problems

- Algorithm/characterization of separated-colorable tree?
- Inserting min #Base pairs: Complexity? Algorithm?
- Complex color sets for more realistic energy models?
- FPT design for some (yet unknown) parameters?
- In practice? Design (approximate) backbone + local search?

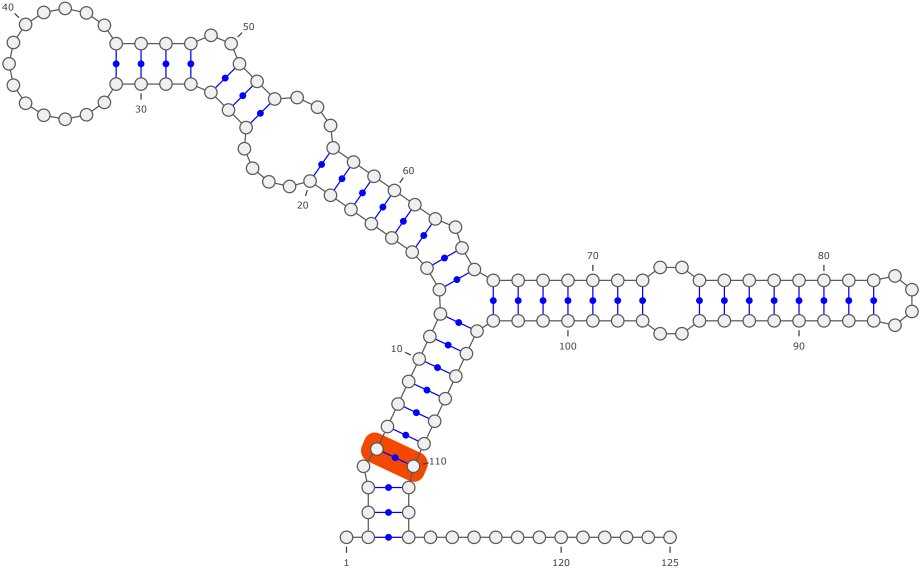
In real life. . .



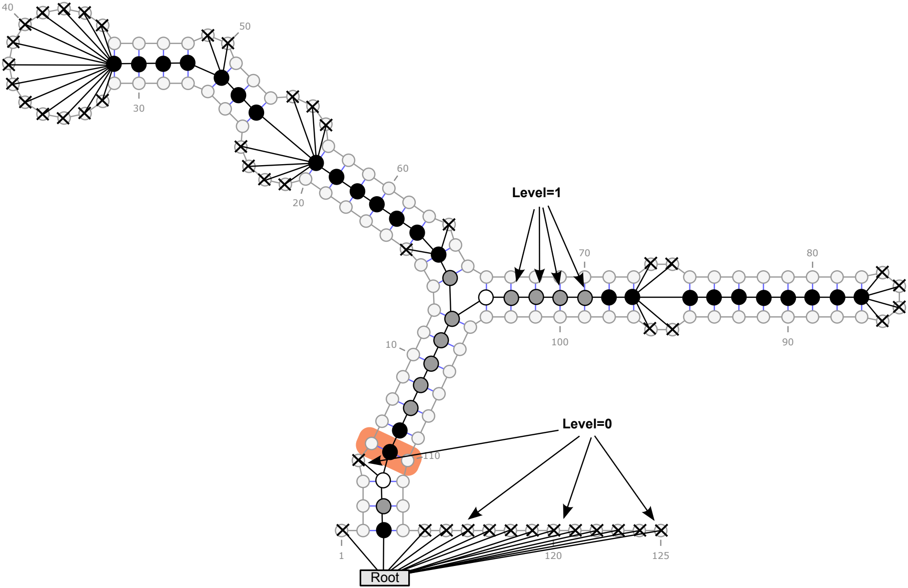
In real life. . .



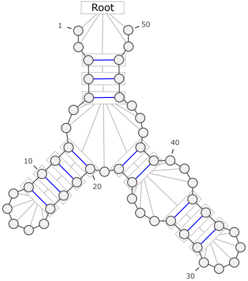
In real life…

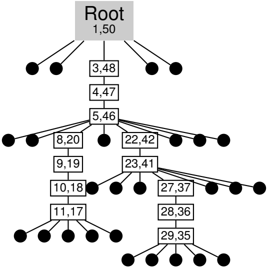


In real life…



Enumerative properties of secondary structures





In dot-bracket notation:

$$\bullet\bullet(((\bullet\bullet((((\bullet\bullet\bullet\bullet\bullet))))\bullet((\bullet\bullet\bullet(((\bullet\bullet\bullet\bullet\bullet)))\bullet\bullet\bullet))\bullet\bullet\bullet)))\bullet\bullet$$

Secondary structures generated by simple context-free grammar

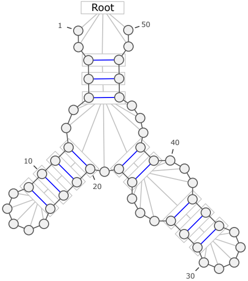
$$S \to \bullet S \mid (\,T\,)\,S \mid \varepsilon \qquad\qquad T \to \bullet S \mid (\,T\,)\,S \mid$$

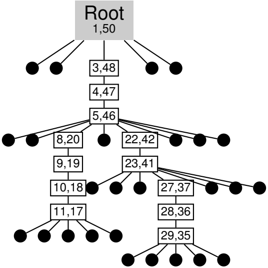
Theorem (Waterman 1978): Number s_n of secondary structures over n nucleotides asymptotically obeys

$$s_n = \frac{\kappa}{2\sqrt{\pi}} \times \frac{\rho^{-n}}{n\sqrt{n}}(1+\mathcal{O}(1/n)) \qquad \kappa := \sqrt{\frac{15+7\sqrt{5}}{2}} \qquad \frac{1}{\rho} := \frac{2}{3-\sqrt{5}} \approx 2.62$$

Techniques: Generating functions + Singularity (complex) analysis

Enumerative properties of secondary structures





In dot-bracket notation:

$$\bullet\bullet(((\bullet\bullet((((\bullet\bullet\bullet\bullet\bullet))))\bullet((\bullet\bullet\bullet(((\bullet\bullet\bullet\bullet\bullet)))\bullet\bullet\bullet))\bullet\bullet\bullet)))\bullet\bullet$$

Secondary structures generated by simple context-free grammar

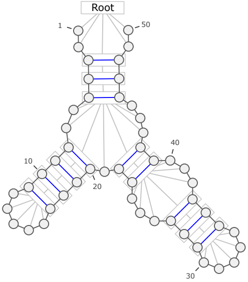
$$S \to \bullet S \mid (\,T\,)\,S \mid \varepsilon \qquad\qquad T \to \bullet S \mid (\,T\,)\,S \mid$$

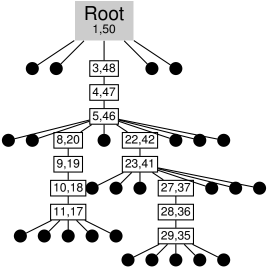
Theorem (Waterman 1978): Number s_n of secondary structures over n nucleotides asymptotically obeys

$$s_n = \frac{\kappa}{2\sqrt{\pi}} \times \frac{\rho^{-n}}{n\sqrt{n}}(1+\mathcal{O}(1/n)) \qquad \kappa := \sqrt{\frac{15+7\sqrt{5}}{2}} \qquad \frac{1}{\rho} := \frac{2}{3-\sqrt{5}} \approx 2.62$$

Techniques: Generating functions + Singularity (complex) analysis

Enumerative properties of secondary structures





In dot-bracket notation:

$$\bullet\bullet(((\bullet\bullet((((\bullet\bullet\bullet\bullet\bullet))))\bullet((\bullet\bullet\bullet(((\bullet\bullet\bullet\bullet\bullet)))\bullet\bullet\bullet))\bullet\bullet\bullet)))\bullet\bullet$$

Secondary structures generated by simple context-free grammar

$$S \to \bullet S \mid (\,T\,)\,S \mid \varepsilon \qquad\qquad T \to \bullet S \mid (\,T\,)\,S \mid$$

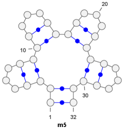
Theorem (Waterman 1978): Number s_n of secondary structures over n nucleotides asymptotically obeys

$$s_n = \frac{\kappa}{2\sqrt{\pi}} \times \frac{\rho^{-n}}{n\sqrt{n}}(1+\mathcal{O}(1/n)) \qquad \kappa := \sqrt{\frac{15+7\sqrt{5}}{2}} \qquad \frac{1}{\rho} := \frac{2}{3-\sqrt{5}} \approx 2.62$$

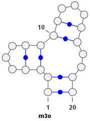
Techniques: Generating functions + Singularity (complex) analysis

Enumerative consequences of forbidden motifs

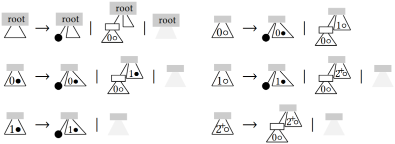
#Secondary structures of size $n \to K.\frac{2.62^n}{n\sqrt{n}}$



m_5



$m_{3\circ}$



#Secondary structures of size n **avoiding** m_5 **and** $m_{3\circ}$: $K'.\frac{\mathbf{2.35}^n}{n\sqrt{n}}$

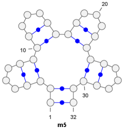
Theorem (Yao/Chauve/Régnier/P, ACM-BCB 2019)

Proportion of **designable** sec. struct. of length *n* decreases exponentially with *n*.

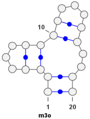
- Generalizes to any list of forbidden motifs (monkey/typewriter *paradox*)
- Forbidden motifs (aka **local obstructions**) exist for all **usual negative design** objectives (**defects**)
- …and can be *black box* computed for complex energy models

Enumerative consequences of forbidden motifs

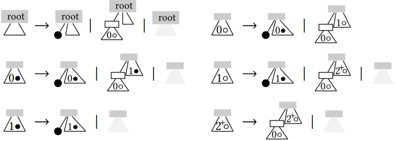
#Secondary structures of size $n \to K.\frac{2.62^n}{n\sqrt{n}}$



m_5



$m_{3\circ}$



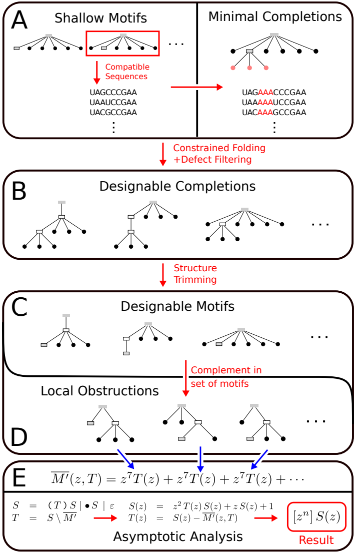
#Secondary structures of size n **avoiding** m_5 **and** $m_{3\circ}$: $K'.\frac{\mathbf{2.35}^n}{n\sqrt{n}}$

Theorem (Yao/Chauve/Régnier/P, ACM-BCB 2019)

Proportion of **designable** sec. struct. of length n decreases exponentially with n.

- Generalizes to any list of forbidden motifs (monkey/typewriter *paradox*)
- Forbidden motifs (aka **local obstructions**) exist for all **usual negative design** objectives (**defects**)
- . . . and can be *black box* computed for complex energy models

Computing local obstructions



A Shallow Motifs

Compatible Sequences

UAGCCCGAA
UAAUCCGAA
UACGCCGAA
⋮

Minimal Completions

UAGAAACCCGAA
UAAAAAUCCGAA
UACAAAGCCGAA
⋮

Constrained Folding
+Defect Filtering

B Designable Completions

Structure
Trimming

C Designable Motifs

Complement in
set of motifs

Local Obstructions

D

E $\overline{M'}(z,T) = z^7T(z) + z^7T(z) + z^7T(z) + \cdots$

$$S = (\,T\,)\,S \mid \bullet S \mid \varepsilon \qquad S(z) = z^2\,T(z)\,S(z) + z\,S(z) + 1$$

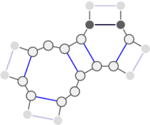
$$T = S \setminus \overline{M'} \longrightarrow T(z) = S(z) - \overline{M'}(z,T) \longrightarrow \boxed{[z^n]\,S(z)}$$

Asymptotic Analysis

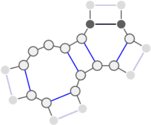
Result

Selected local obstructions in Turner energy models

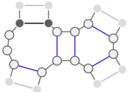
Distance to subopts $\Delta > 0$ kcal.mol^{-1} ($d^S < 1$) → 17 local obstructions.



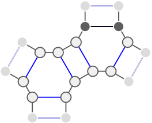
Motif #1
Forbidden motif



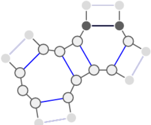
Motif #2
Forbidden motif



Motif #3
Forbidden motif

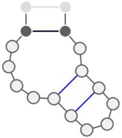


Motif #4
Forbidden motif

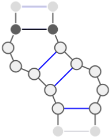


Motif #5
Forbidden motif

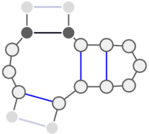
Distance to subopts $\Delta \geq 1$ kcal.mol^{-1} ($d^S < 1/e$) → 28 local obstructions.



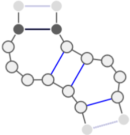
Motif #1
0.67032



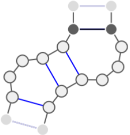
Motif #2
0.60653



Motif #3
0.49659



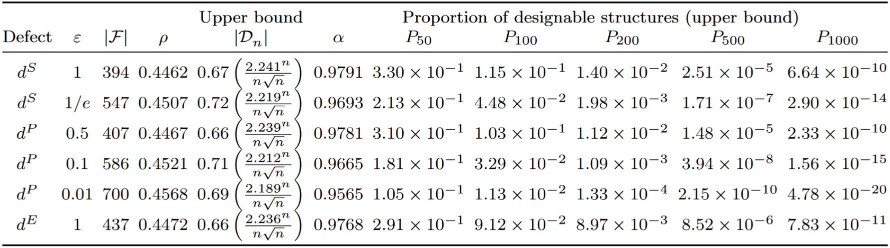
Motif #4
0.44933



Motif #5
0.44933

- Very few occurrences in experimental 3D RNA structures (PDB)
- Always seemingly stabilized by **non-canonical base pairs**

Impact of design objectives



				Upper bound		Proportion of designable structures (upper bound)				
Defect	ε	$\lvert\mathcal{F}\rvert$	ρ	$\lvert\mathcal{D}_n\rvert$	α	P_{50}	P_{100}	P_{200}	P_{500}	P_{1000}
d^S	1	394	0.4462	$0.67\left(\frac{2.241^n}{n\sqrt{n}}\right)$	0.9791	3.30×10^{-1}	1.15×10^{-1}	1.40×10^{-2}	2.51×10^{-5}	6.64×10^{-10}
d^S	$1/e$	547	0.4507	$0.72\left(\frac{2.219^n}{n\sqrt{n}}\right)$	0.9693	2.13×10^{-1}	4.48×10^{-2}	1.98×10^{-3}	1.71×10^{-7}	2.90×10^{-14}
d^P	0.5	407	0.4467	$0.66\left(\frac{2.239^n}{n\sqrt{n}}\right)$	0.9781	3.10×10^{-1}	1.03×10^{-1}	1.12×10^{-2}	1.48×10^{-5}	2.33×10^{-10}
d^P	0.1	586	0.4521	$0.71\left(\frac{2.212^n}{n\sqrt{n}}\right)$	0.9665	1.81×10^{-1}	3.29×10^{-2}	1.09×10^{-3}	3.94×10^{-8}	1.56×10^{-15}
d^P	0.01	700	0.4568	$0.69\left(\frac{2.189^n}{n\sqrt{n}}\right)$	0.9565	1.05×10^{-1}	1.13×10^{-2}	1.33×10^{-4}	2.15×10^{-10}	4.78×10^{-20}
d^E	1	437	0.4472	$0.66\left(\frac{2.236^n}{n\sqrt{n}}\right)$	0.9768	2.91×10^{-1}	9.12×10^{-2}	8.97×10^{-3}	8.52×10^{-6}	7.83×10^{-11}

Extension to bivariate analysis (ensemble defect)

Definition: Target $S^\star$, sequence w
Ensemble defect $\mathcal{D}_E(w, S^\star)$ = Expected distance to $S^\star$ within Boltzmann distribution

$$\mathcal{D}_E(w) = \sum_{S \in \mathcal{S}_w} BPDist(S, S^\star)\,\frac{e^{-E_{w,S}/kT}}{\mathcal{Z}_w}$$

Property: $\mathcal{D}_E$ is super additive over any subset of **disjoint** motifs $m_1, m_2 \ldots$ in $S^\star$

$$\min_{w} \mathcal{D}_E(w, S^\star) \geq \left(\min_{w_1} \mathcal{D}_E(w_1, m_1)\right) + \left(\min_{w_2} \mathcal{D}_E(w_2, m_2)\right) + \ldots$$

→ Additive **lower bound** for ensemble defect

Remark: Occurrences of motifs can be **marked** within sec. struct. grammar
→ **Bivariate gen. fun.** + strongly connected, aperiodic system of equations
→ **Normal distribution** for lower bound on defect **(Drmota Theorem)**
(Expectation: μn, Std dev.: $\sigma\sqrt{n}$)

Extension to bivariate analysis (ensemble defect)

Definition: Target $S^\star$, sequence w
Ensemble defect $\mathcal{D}_E(w, S^\star)$ = Expected distance to $S^\star$ within Boltzmann distribution

$$\mathcal{D}_E(w) = \sum_{S\in\mathcal{S}_w} BPDist(S, S^\star)\frac{e^{-E_{w,S}/kT}}{\mathcal{Z}_w}$$

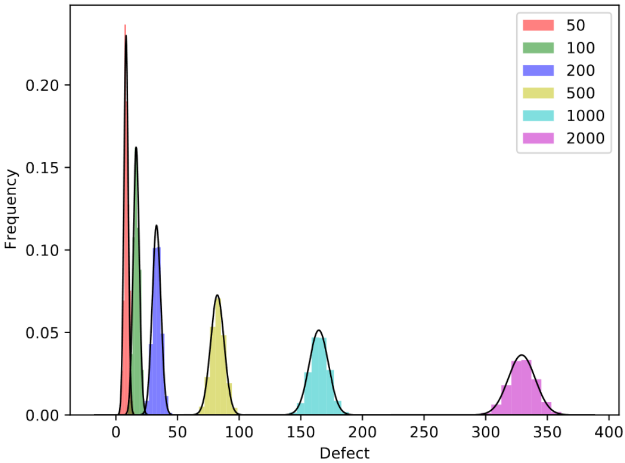
Property: $\mathcal{D}_E$ is super additive over any subset of **disjoint** motifs $m_1, m_2 \ldots$ in $S^\star$

$$\min_{w} \mathcal{D}_E(w, S^\star) \geq \left(\min_{w_1} \mathcal{D}_E(w_1, m_1)\right) + \left(\min_{w_2} \mathcal{D}_E(w_2, m_2)\right) + \ldots$$

→ Additive **lower bound** for ensemble defect

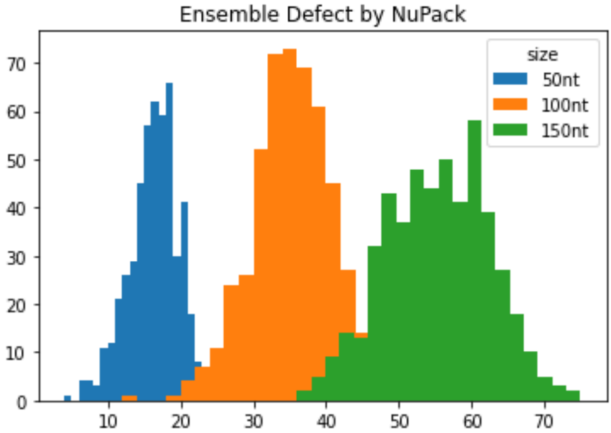
Remark: Occurrences of motifs can be **marked** within sec. struct. grammar
→ **Bivariate gen. fun.** + strongly connected, aperiodic system of equations
→ **Normal distribution** for lower bound on defect **(Drmota Theorem)**
(Expectation: μn, Std dev.: $\sigma\sqrt{n}$)

Asymptotic distribution of ensemble defect



- List of motifs restricted to ensure absence of overlap
- Motifs additive → Lower bound on real ensemble defect

Empirical distribution of ensemble defect



- NUPACK optimizes ensemble defect [Zadeh *et al*, 2011]
- Local search → Upper bound on real ensemble defect

Conclusions

- RNA design is a **timely** topic for Bio Maths/CS
- Negative design, a hard problem, poorly understood
  → Future combinatorial studies needed!
- Structure approximating design: a promising **tractable** alternative?
- Parameterized complexity of inverse folding?
- Forbidden motifs: **Ubiquitous** in DP-based inverse combinatorial optimization
- **Way** less designable structures than initially thought
- Does **Nature** find a way around undesignability?
  Or should we refine phenotype/genotype studies (neu**t**ral networks)?

Merci – Thank you

Collaborators:

Ecole Polytechnique

- S. Will, H.T. Yao
- M. Régnier, A. Héliou

Simon Fraser University

- J. Hales, J. Manuch, L. Stacho
- C. Chauve

McGill University

- J. Waldispühl

Université du Québec à Montréal

- V. Reinharz

University of Vienna

- S. Will, S. Hammer

Ben Gurion University

- D. Barash, M. Drory Retwitzer, A. Churkin

Supported by:

cnrs

Agence Nationale de la Recherche
ANR

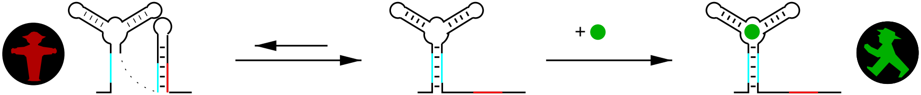
FWF

משרד המדע,
הטכנולוגיה והחלל
Ministry of Science, Technology & Space

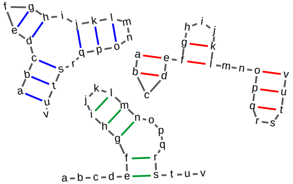
Supp. Mat. Positive design for multiple RNAs

Multiple RNA design: Motivation

Example: *Riboswitch* for translation control



Multiple target structures → ***Multiple design of RNAs***


a b c d e f g h i j k l m n o p q r s t u v
a b c d e f g h i j k l m n o p q r s t u v
a-b-c-d-e f g h i j k l m n o p q r s-t-u-v

```
abcdefghijklmnopqrstuv
(((((.)).(((..))).))).
((.))((...))..(((..)))
....(((((..)))...))...
```

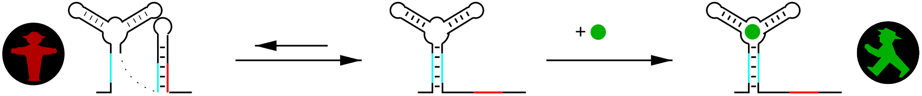
Objective: To **randomly** generate RNA sequences under constraints

1. **Validity** for targeted structures wrt base pairing nucleotides
2. **Stability** (low free-energy, comparable across structures…) of target structures
3. **Constrained composition:** (prescribed GC content), +/- motifs…

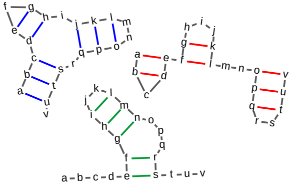
Stochastic backtrack: Pre-count and generate **valid** sequence (uniform distrib.)
+ Further refinements using **local search**

Multiple RNA design: Motivation

Example: *Riboswitch* for translation control



Multiple target structures → *Multiple design of RNAs*



```
abcdefghijklmnopqrstuv
(((((.)).(((..))).))).
((.))((...))..(((..)))
....(((((..)))...))...
```

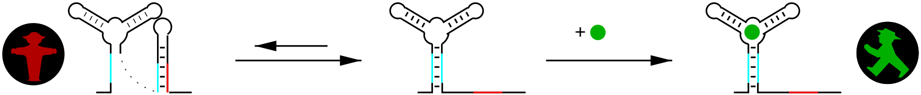
Objective: To **randomly** generate RNA sequences under constraints

1. **Validity** for targeted structures wrt base pairing nucleotides
2. **Stability** (low free-energy, comparable across structures. . .) of target structures
3. **Constrained composition:** (prescribed GC content), +/- motifs. . .

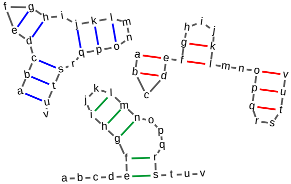
Stochastic backtrack: Pre-count and generate **valid** sequence (uniform distrib.)
+ Further refinements using **local search**

Multiple RNA design: Motivation

Example: *Riboswitch* for translation control



Multiple target structures → *Multiple design of RNAs*



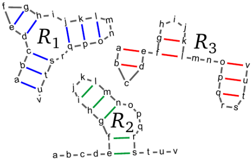
```
abcdefghijklmnopqrstuv
(((((.)).(((..))).))).
((.))((...))..(((..)))
....(((((..)))...))...
```

Objective: To **randomly** generate RNA sequences under constraints

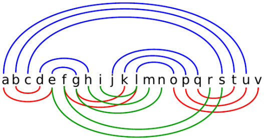
1. **Validity** for targeted structures wrt base pairing nucleotides
2. **Stability** (low free-energy, comparable across structures. . .) of target structures
3. **Constrained composition:** (prescribed GC content), +/- motifs. . .

Stochastic backtrack: Pre-count and generate **valid** sequence (uniform distrib.)
+ Further refinements using **local search**

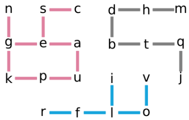
Our problem (simplified)



i) Input Structures



ii) Merged Base-Pairs



iii) Compatibility Graph

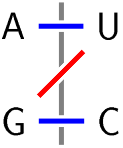
Question: How many valid sequences over $\Sigma^n := \{A, C, G, U\}^n$?

Problem (#ValidSequences)

Input: Secondary structures $\mathcal{R} = \{R_1, \ldots, R_k\}$ of length n

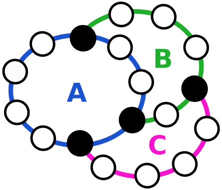
Output: Num. of valid sequences

$$|\{S \in \Sigma^n \mid \forall (i,j) \in R_\ell, (S_i, S_j) \text{ forms a valid base pair}\}|$$



Valid base pairs

State of the art



Abfalter/Flamm/Stadler 2003:

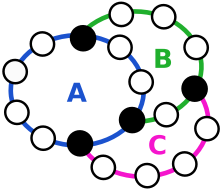
- Ear decomposition **[Whitney 1932]**
- *Peel input graph* as paths $A_1, \ldots, A_k$ such that only the ends of A_i are in $\cup_{j>i} A_j$
- **Dynamic programming:** Counting #valid paths for each component, conditioned by nucleotide chosen for its **anchors** (black nodes);
- Careful **combination** of values yields #valid sequences.

Complexity: $\Theta(n.4^{\Omega})$ where $\Omega = $ Max #anchors. Worst-case: $\Omega \in \Theta(n)$

Some comments:

- Is this optimal? Other algorithms/parameters?
- Which extensions possible? (Multidim.) Boltzmann-Gibbs distrib.
- Is this exp. really necessary? **Probably** since counting #P-hard

State of the art



Abfalter/Flamm/Stadler 2003:

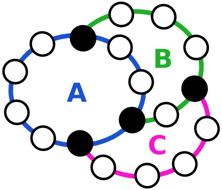
- Ear decomposition **[Whitney 1932]**
- *Peel input graph* as paths $A_1, \ldots, A_k$ such that only the ends of A_i are in $\cup_{j>i} A_j$
- **Dynamic programming:** Counting #valid paths for each component, conditioned by nucleotide chosen for its **anchors** (black nodes);
- Careful **combination** of values yields #valid sequences.

Complexity: $\Theta(n.4^{\Omega})$ where $\Omega =$ Max #anchors. Worst-case: $\Omega \in \Theta(n)$

Some comments:

- Is this optimal? Other algorithms/parameters?
- Which extensions possible? (Multidim.) Boltzmann-Gibbs distrib.
- Is this exp. really necessary? **Probably** since counting #P-hard

State of the art



Abfalter/Flamm/Stadler 2003:

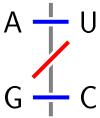
- Ear decomposition **[Whitney 1932]**
- *Peel input graph* as paths $A_1, \ldots, A_k$ such that only the ends of A_i are in $\cup_{j>i} A_j$
- **Dynamic programming:** Counting #valid paths for each component, conditioned by nucleotide chosen for its **anchors** (black nodes);
- Careful **combination** of values yields #valid sequences.

Complexity: $\Theta(n.4^{\Omega})$ where $\Omega =$ Max #anchors. Worst-case: $\Omega \in \Theta(n)$

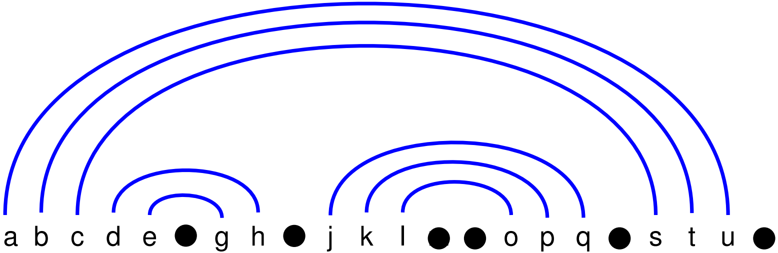
Some comments:

- Is this optimal? Other algorithms/parameters?
- Which extensions possible? (Multidim.) Boltzmann-Gibbs distrib.
- Is this exp. really necessary? **Probably** since counting #P-hard

Counting valid sequences: WC/Wobble + single structure



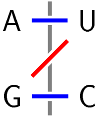
Valid base pairs (BPs) = Including **Wobble** base pairs



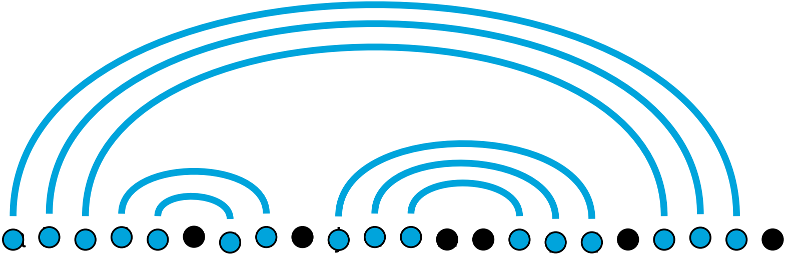
Question: How many **valid** sequences?

Answer: $4^{\#\text{Unpaired}} \times 6^{\#\text{BPs}} \rightarrow 6\,879\,707\,136$

Counting valid sequences: WC/Wobble + single structure



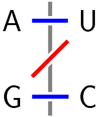
Valid base pairs (BPs) = Including **Wobble** base pairs



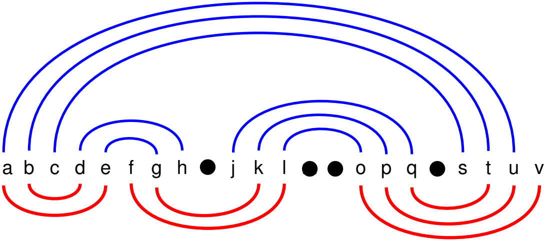
Question: How many **valid** sequences?

Answer: $4^{\#\text{Unpaired}} \times 6^{\#\text{BPs}} \rightarrow 6\,879\,707\,136$

Counting valid sequences: WC/Wobble + Two structures

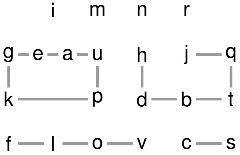


Valid base pairs (BPs) = Including **Wobble** base pairs



Dependency graph:

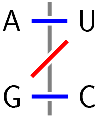
Cycles + Paths



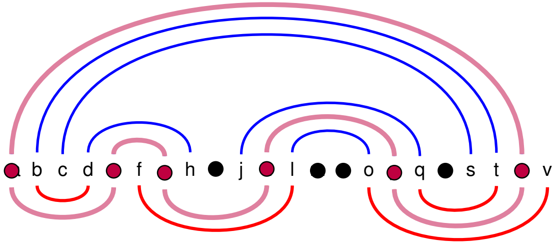
Question: How many **valid** sequences?

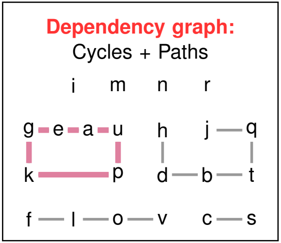
Answer: $\neq \emptyset$! (dep. graph and valid BPs both **bipartite** **[Flamm *et al*, RNA 2001]**)

Counting valid sequences: WC/Wobble + Two structures



Valid base pairs (BPs) = Including **Wobble** base pairs





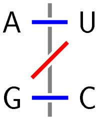
Dependency graph:

Cycles + Paths

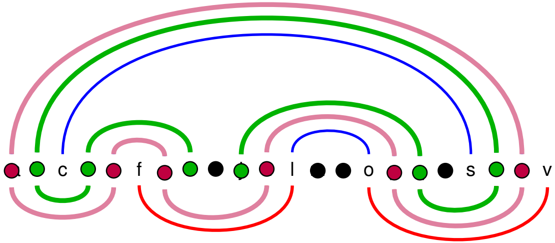
Question: How many **valid** sequences?

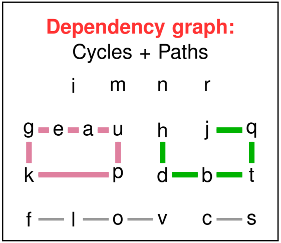
Answer: $\neq \emptyset$! (dep. graph and valid BPs both **bipartite** **[Flamm *et al*, RNA 2001]**)

Counting valid sequences: WC/Wobble + Two structures



Valid base pairs (BPs) = Including **Wobble** base pairs



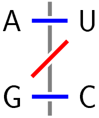

Dependency graph:

Cycles + Paths

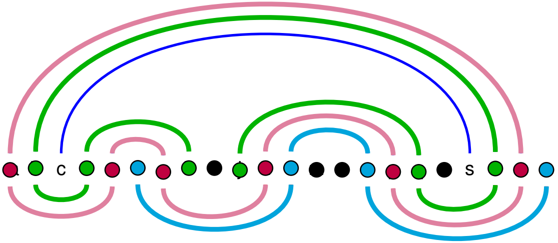
Question: How many **valid** sequences?

Answer: $\neq \emptyset$! (dep. graph and valid BPs both **bipartite** **[Flamm *et al*, RNA 2001]**)

Counting valid sequences: WC/Wobble + Two structures

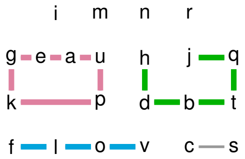


Valid base pairs (BPs) = Including **Wobble** base pairs



Dependency graph:

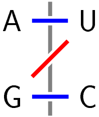
Cycles + Paths



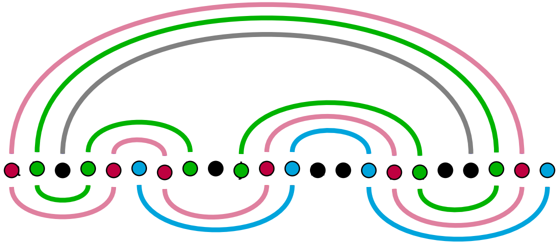
Question: How many **valid** sequences?

Answer: $\neq \emptyset$! (dep. graph and valid BPs both **bipartite** **[Flamm *et al*, RNA 2001]**)

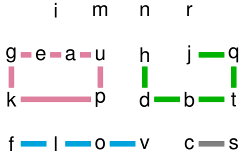
Counting valid sequences: WC/Wobble + Two structures



Valid base pairs (BPs) = Including **Wobble** base pairs



Dependency graph:
Cycles + Paths



Question: How many **valid** sequences?

Answer: $\neq \emptyset$! (dep. graph and valid BPs both **bipartite [Flamm *et al*, RNA 2001]**)

$$\#\text{Designs}(G) = \prod_{c \in CC(G)} \#\text{Designs}(cc)$$

Counting valid sequences for paths and cycles

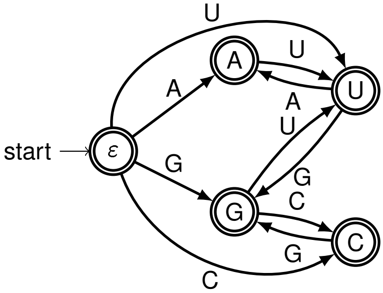
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For paths: A simple automaton. . .



Remark: A ↔ C/G ↔ U symmetry

Counting valid sequences for paths and cycles

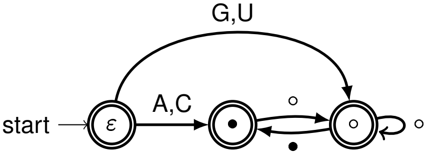
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For paths: A simple automaton. . .



Remark: A $\leftrightarrow$ C/G $\leftrightarrow$ U symmetry

Counting valid sequences for paths and cycles

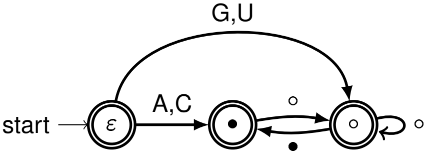
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For paths: A simple automaton. . .



Remark: A ↔ C/G ↔ U symmetry

$$m_{\bullet}(n) = m_{\circ}(n-1)$$

Counting valid sequences for paths and cycles

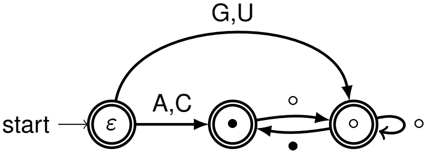
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For paths: A simple automaton. . .



Remark: A ↔ C/G ↔ U symmetry

$$\begin{aligned} m_\bullet(n) &= m_\circ(n-1) \\ m_\circ(n) &= m_\circ(n-1) + m_\bullet(n-1) \\ &= m_\circ(n-1) + m_\circ(n-2) \\ &= \mathcal{F}(n+2) \end{aligned}$$

Counting valid sequences for paths and cycles

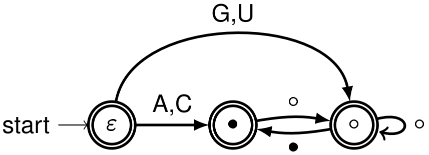
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For paths: A simple automaton. . .



Remark: A ↔ C/G ↔ U symmetry

$$\begin{aligned} m_\bullet(n) &= m_\circ(n-1) \\ m_\circ(n) &= m_\circ(n-1) + m_\bullet(n-1) \\ &= m_\circ(n-1) + m_\circ(n-2) \\ &= \mathcal{F}(n+2) \end{aligned}$$

(Since $m_\circ(0) = 1$ and $m_\circ(1) = 2$)

Counting valid sequences for paths and cycles

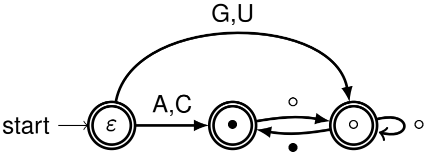
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For paths: A simple automaton…



Remark: A ↔ C/G ↔ U symmetry

$$\begin{aligned} m_\bullet(n) &= m_\circ(n-1) \\ m_\circ(n) &= m_\circ(n-1) + m_\bullet(n-1) \\ &= m_\circ(n-1) + m_\circ(n-2) \\ &= \mathcal{F}(n+2) \end{aligned}$$

(Since $m_\circ(0) = 1$ and $m_\circ(1) = 2$)

$$p(n) := m_\varepsilon(n) = 2\,m_\bullet(n-1) + 2\,m_\circ(n-1) = 2(\mathcal{F}(n) + \mathcal{F}(n+1)) = 2\mathcal{F}(n+2)$$

Counting valid sequences for paths and cycles

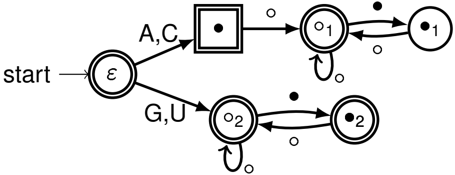
$p(n)$: #Valid sequences for **path** of length n.
$c(n)$: #Valid sequences for cycle of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

For cycles: A slightly more complex automaton. . .



Counting valid sequences for paths and cycles

$p(n)$ and $c(n)$: #Valid sequences for **paths** and cycles of length n.

Theorem (#Valid sequences for paths and cycles)

$$p(n) = 2\,\mathcal{F}_{n+2} \qquad \text{et} \qquad c(n) = 2\,\mathcal{F}_n + 4\,\mathcal{F}_{n-1}$$

where $\mathcal{F}_n$ is the n-th Fibonacci number.

G: Dependency graph, merging the two structures (max degree ≤ 2).
G uniquely decomposed in $\mathcal{P}(G)$ paths and $\mathcal{C}(G)$ cycles.

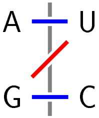
Theorem (#Valid sequences for 2-structures)

The number $\#\text{Designs}(G)$ of valid sequences for G is

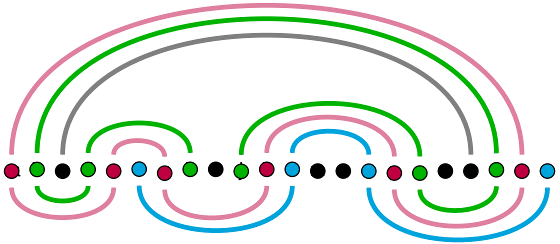
$$\#\text{Designs}(G) = \prod_{p\in\mathcal{P}(G)} 2\,\mathcal{F}_{|p|+2} \times \prod_{c\in\mathcal{C}(G)} \left(2\,\mathcal{F}_{|c|} + 4\,\mathcal{F}_{|c|-1}\right)$$

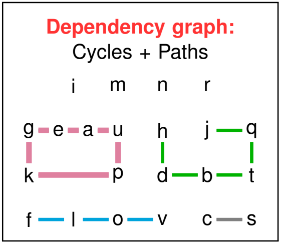
Caterpilar tree: $\frac{(2+\sqrt{3})\times(1+\sqrt{3})^n+(2-\sqrt{3})\times(1-\sqrt{3})^n}{2}$ (n nodes)
Complete binary: $2\,a_k$ (height k) $a_k = (a_{k-2}+1)^4 + 2(a_{k-1}+1)(a_{k-2}+1)^2 + (a_{k-1}+1)^2 - 1$

Counting valid sequences: WC/Wobble + Two structures



Valid base pairs (BPs) = Including **Wobble** base pairs





Dependency graph:

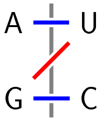
Cycles + Paths

Question: How many **valid** sequences?

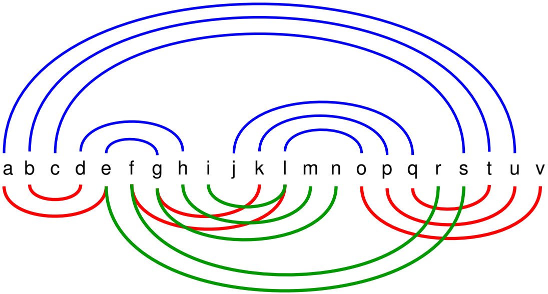
Answer :$\neq \varnothing$! (both BP and dependency graphs **bipartite**)

$$\#\text{Designs}(G) = \prod_{c\in CC(G)} \#\text{Designs}(cc) = 2\,322\,432$$

Counting valid sequences: WC/Wobble + > 2 structures

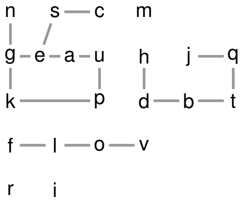


Valid base pairs (BPs) = Including **Wobble** base pairs



Dependency graph:

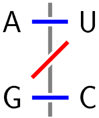
Cycles, Paths, Trees. . .



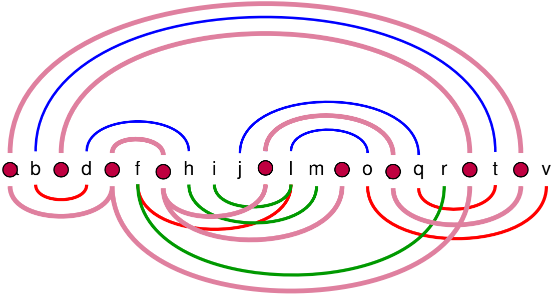
Question: How many valid **sequences**?

Answer: Non-bipartite → ∅; Bipartite → ????

Counting valid sequences: WC/Wobble + >2 structures

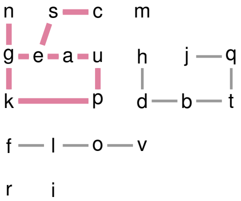


Valid base pairs (BPs) = Including **Wobble** base pairs



Dependency graph:

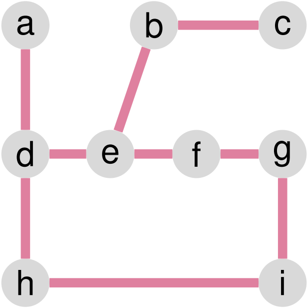
Cycles, Paths, Trees. . .

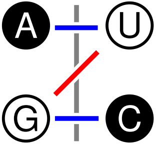


Question: How many valid **sequences**?

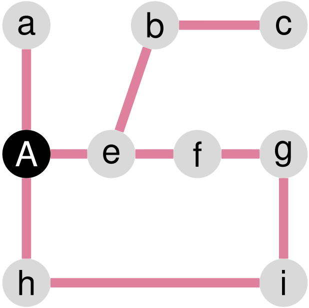
Answer: Non-bipartite → ∅; Bipartite → ????

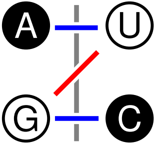
Independent sets ⇔ Valid sequences



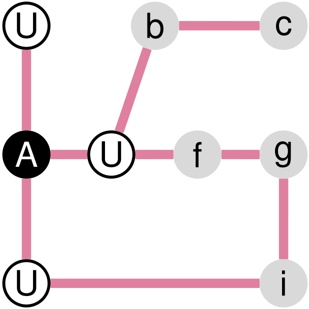


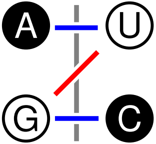
Independent sets ⇔ Valid sequences



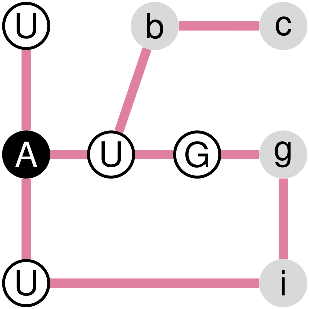


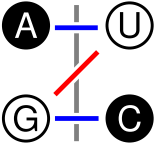
Independent sets ⇔ Valid sequences



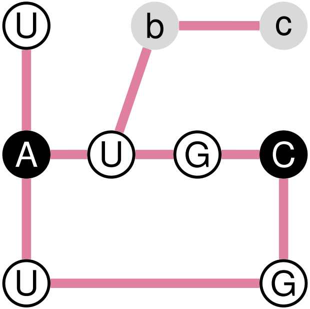


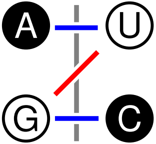
Independent sets ⇔ Valid sequences



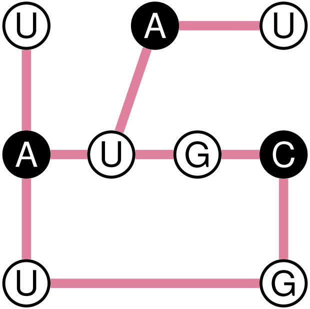


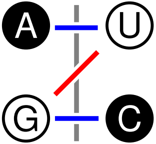
Independent sets ⇔ Valid sequences



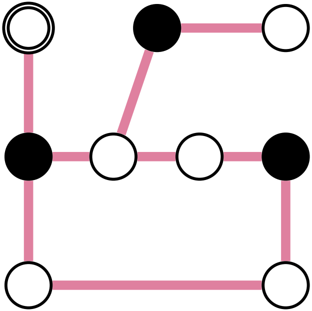


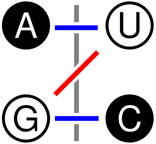
Independent sets ⇔ Valid sequences





Independent sets ⇔ Valid sequences



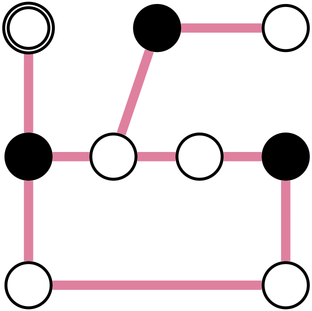


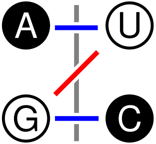
Remark: **Black circles** non-adjacent in valid sequences

Up to trivial symmetry* (*e.g.* north-western position $\in \{U, C\}$):

$$\text{Designs}^\star(\text{cc}) \subseteq \text{IndSets}(\text{cc})$$

Independent sets ⇔ Valid sequences





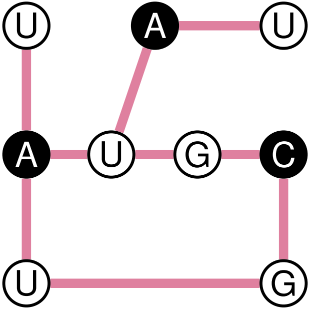
Remark: **Black circles** non-adjacent in valid sequences

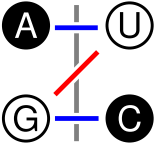
Up to trivial symmetry* (*e.g.* north-western position $\in \{U, C\}$):

$$\text{Designs}^\star(\text{cc}) \subseteq \text{IndSets}(\text{cc})$$

Independent Sets (black) + NW $\in \{U, C\} \Rightarrow$ Valid sequence

Independent sets ⇔ Valid sequences





Remark: Black circles non-adjacent in valid sequences

Up to trivial symmetry* (*e.g.* north-western position $\in \{U, C\}$):

$$\text{Designs}^\star(\text{cc}) \subseteq \text{IndSets}(\text{cc})$$

Independent Sets (black) + NW $\in \{U, C\} \Rightarrow$ Valid sequence

$\Rightarrow$ Bijection between Designs*(cc) and IndSets(cc).

Valid sequences and independent sets

Theorem (#Designs and ind. sets in connected bipartite graphs)

Let G be a **bipartite and connected** dependency graph:

$$\#\text{Designs(G)} = 2 \times \#\text{Designs}^{\star}\text{(G)} = 2 \times \#\text{IndSets(G)}$$

For **bipartite** dependency graph G, one has:

$$\#Designs(G) = \prod_{cc \in CC(G)} 2 \times \#IndSets(cc) = 2^{|CC(G)|} \times \#IndSets(G)$$

But $\#IndSets(G)$ is **#P-hard** on bipartite graphs ($\#BIS$) **[Dyer & Greenhill'00]**

(+ Any graph G is the dependency graph of some structure family)

So $\exists$ Poly-Time algorithm for $\#Designs(G) \to$ Poly-Time algorithm for $\#BIS\ldots$

Theorem

Counting #Designs is #P-hard.

No Poly-Time algorithm for #Designs(G) **unless** $\#P = FP$ ($\Rightarrow P = NP$)

Valid sequences and independent sets

Theorem (#Designs and ind. sets in connected bipartite graphs)

Let G be a **bipartite and connected** dependency graph:

$$\#\text{Designs}(\text{G}) = 2 \times \#\text{Designs}^{\star}(\text{G}) = 2 \times \#\text{IndSets}(\text{G})$$

For **bipartite** dependency graph G, one has:

$$\#Designs(G) = \prod_{cc \in CC(G)} 2 \times \#IndSets(cc) = 2^{|CC(G)|} \times \#IndSets(G)$$

But $\#IndSets(G)$ is **$\#$P-hard** on bipartite graphs ($\#BIS$) **[Dyer & Greenhill'00]**

(+ Any graph G is the dependency graph of some structure family)

So $\exists$ Poly-Time algorithm for $\#Designs(G) \rightarrow$ Poly-Time algorithm for $\#BIS\ldots$

Theorem

Counting #Designs is #P-hard.

No Poly-Time algorithm for #Designs(G) **unless** $\#P = FP$ ($\Rightarrow P = NP$)

Valid sequences and independent sets

Theorem (#Designs and ind. sets in connected bipartite graphs)

Let G be a **bipartite and connected** dependency graph:

$$\#\text{Designs(G)} = 2 \times \#\text{Designs}^\star\text{(G)} = 2 \times \#\text{IndSets(G)}$$

For **bipartite** dependency graph G, one has:

$$\#Designs(G) = \prod_{cc \in CC(G)} 2 \times \#IndSets(cc) = 2^{|CC(G)|} \times \#IndSets(G)$$

But $\#IndSets(G)$ is #P**-hard** on bipartite graphs ($\#BIS$) **[Dyer & Greenhill'00]**

(+ Any graph G is the dependency graph of some structure family)

So ∃ Poly-Time algorithm for $\#Designs(G)$ → Poly-Time algorithm for $\#BIS$...

Theorem

Counting #Designs is #P-hard.

No Poly-Time algorithm for #Designs(G) **unless** $\#P = FP$ ($\Rightarrow P = NP$)

Valid sequences and independent sets

Theorem (#Designs and ind. sets in connected bipartite graphs)

Let G be a **bipartite and connected** dependency graph:

$$\#\mathrm{Designs}(\mathrm{G}) = 2 \times \#\mathrm{Designs}^{\star}(\mathrm{G}) = 2 \times \#\mathrm{IndSets}(\mathrm{G})$$

For **bipartite** dependency graph G, one has:

$$\#Designs(G) = \prod_{cc \in CC(G)} 2 \times \#IndSets(cc) = 2^{|CC(G)|} \times \#IndSets(G)$$

But $\#IndSets(G)$ is **#P-hard** on bipartite graphs ($\#BIS$) **[Dyer & Greenhill'00]**

(+ Any graph G is the dependency graph of some structure family)

So $\nexists$ Poly-Time algorithm for $\#Designs(G) \to$ Poly-Time algorithm for $\#BIS\ldots$

Theorem

Counting #Designs is #P-hard.

No Poly-Time algorithm for #Designs(G) **unless** $\#P = FP$ ($\Rightarrow P = NP$)

Consequences

Corollary (#Approximability for ≤ 5 structures) [[Weitz'06]

For ≤ 5 structures (crossings allowed), #Design(G) can be approximated within **any ratio** in **Poly-time** (PTAS)

Corollary (#BIS-hardness for > 5 structures) [[Cai, Galanis *et al*'16]

For more than 5 structures (crossings allowed), #Design is **equally as hard** to approximate as general #BIS.

Why crossings/Pseudokots? Because any bipartite graph of max degree Δ can be **decomposed** in Δ matchings in **Poly-Time** (Vizing theorem).

Connection between **counting** and **sampling** [Jerrum/Valiant/Vazirani'86].

Conjecture (#BIS-hardness of multiple positive design)

Quasi-uniform generation as hard as approximation of general #BIS

$\Rightarrow$ **Sampling** #P hard?

Consequences

Corollary (#Approximability for ≤ 5 structures) [[Weitz’06]

For ≤ 5 structures (crossings allowed), #Design(G) can be approximated within **any ratio** in **Poly-time** (PTAS)

Corollary (#BIS-hardness for > 5 structures) [[Cai, Galanis *et al*’16]

For more than 5 structures (crossings allowed), #Design is **equally as hard** to approximate as general #BIS.

Why crossings/Pseudokots? Because any bipartite graph of max degree Δ can be **decomposed** in Δ matchings in **Poly-Time** (Vizing theorem).

Connection between **counting** and **sampling** [Jerrum/Valiant/Vazirani’86].

Conjecture (#BIS-hardness of multiple positive design)

Quasi-uniform generation as hard as approximation of general #BIS

$\Rightarrow$ **Sampling** #P hard?

Consequences

Corollary (#Approximability for ≤ 5 structures) [[Weitz'06]

For ≤ 5 structures (crossings allowed), #Design(G) can be approximated within **any ratio** in **Poly-time** (PTAS)

Corollary (#BIS-hardness for > 5 structures) [[Cai, Galanis *et al*'16]

For more than 5 structures (crossings allowed), #Design is **equally as hard** to approximate as general #BIS.

Why crossings/Pseudokots? Because any bipartite graph of max degree Δ can be **decomposed** in Δ matchings in **Poly-Time** (Vizing theorem).

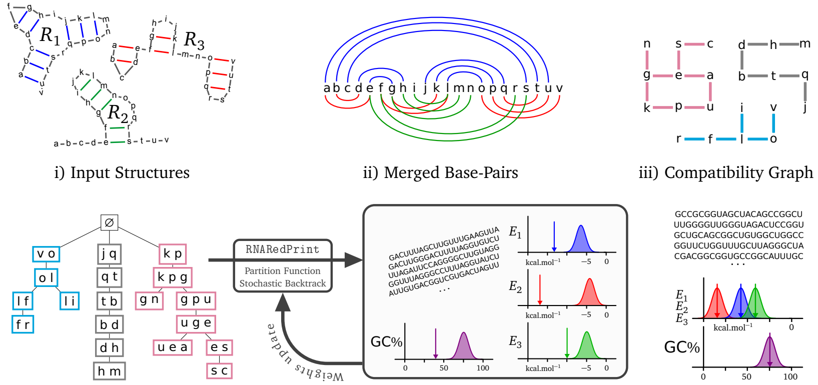
Connection between **counting** and **sampling** **[Jerrum/Valiant/Vazirani'86]**.

Conjecture (#BIS-hardness of multiple positive design)

Quasi-uniform generation as hard as approximation of general #BIS

$\Rightarrow$ **Sampling** #P hard?

Tree decomposition and Boltzmann sampling of sequences



i) Input Structures

ii) Merged Base-Pairs

iii) Compatibility Graph

iv) Tree Decomposition

v) Weight Optimization (Adaptive Sampling)

vi) Final Designs

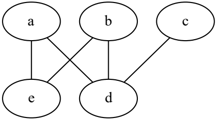
Tree decomposition and width

Tree decomposition T for a graph $G = (V, E)$:

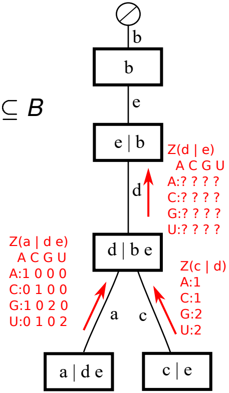
1. Nodes of T = Some subsets of V
2. **All vertices present:** $\forall v \in V, \exists b \in B$ s.t. $v \in b$
3. **All edges present:** $\forall (v, v') \in E, \exists b \in B$ s.t. $\{v, v'\} \subseteq B$
4. Nodes having $v \in V$ form a **connected** subtreee

a	b	c	d	e
(	.	.	)	.
.	(	(	)	)
(	(	.	)	)

Target structures



Dependency graph



Z(d | e)

	A	C	G	U
A:	?	?	?	?
C:	?	?	?	?
G:	?	?	?	?
U:	?	?	?	?

Z(a | d e)

	A	C	G	U
A:	1	0	0	0
C:	0	1	0	0
G:	1	0	2	0
U:	0	1	0	2

Z(c | d)

A:1
C:1
G:2
U:2

Tree decomposition

$b = \{b_1, b_2 \ldots\}$: node of D

T_b : subtree rooted at b

w : **Width** of tree decomposition D ($=\max_{b \in B} |b| - 1$)

$$\mathcal{Z}(T_b \mid b_2 \leftarrow v_2 \ldots) = \sum_{\substack{b_1 \leftarrow v_1 \\ v_1 \in \{A,C,G,U\}}} \prod_{c \text{ child of } b} \mathcal{Z}(T_c \mid b_1 \leftarrow v_1, b_2 \leftarrow v_2 \ldots)$$

Complexity: $\Theta(n\,m\,k + n\,k\,2^w)$ for **uniform generation** of m sequences (k structs)

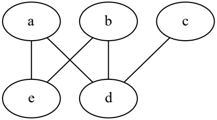
Tree decomposition and width

Tree decomposition T for a graph $G = (V, E)$:

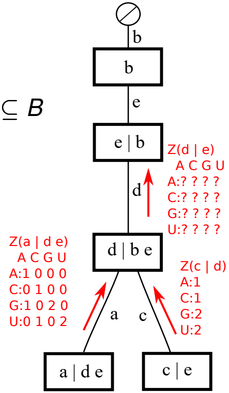
1. Nodes of T = Some subsets of V
2. **All vertices present:** $\forall v \in V, \exists b \in B$ s.t. $v \in b$
3. **All edges present:** $\forall (v, v') \in E, \exists b \in B$ s.t. $\{v, v'\} \subseteq B$
4. Nodes having $v \in V$ form a **connected** subtreee

a	b	c	d	e
(	.	.	)	.
.	(	(	)	)
(	(	.	)	)

Target structures



Dependency graph



Z(d | e)

	A	C	G	U
A:	?	?	?	?
C:	?	?	?	?
G:	?	?	?	?
U:	?	?	?	?

Z(a | d e)

	A	C	G	U
A:	1	0	0	0
C:	0	1	0	0
G:	1	0	2	0
U:	0	1	0	2

Z(c | d)

A:	1
C:	1
G:	2
U:	2

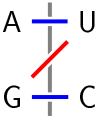
Tree decomposition

$b = \{b_1, b_2 \ldots\}$: node of D
T_b : subtree rooted at b
w : **Width** of tree decomposition D (=$\max_{b\in B} |b| - 1$)

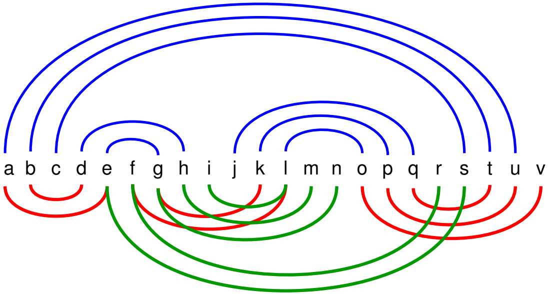
$$\mathcal{Z}(T_b \mid b_2 \leftarrow v_2 \ldots) = \sum_{\substack{b_1 \leftarrow v_1 \\ v_1 \in \{A,C,G,U\}}} \prod_{c \text{ child of } b} \mathcal{Z}(T_c \mid b_1 \leftarrow v_1, b_2 \leftarrow v_2 \ldots)$$

Complexity: $\Theta\left(n\,m\,k + n\,k\,2^w\right)$ for **uniform generation** of m sequences (k structs)

Counting valid sequences: WC/Wobble + > 2 structures

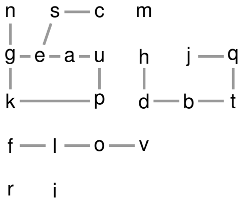


Valid base pairs (BPs) = Including **Wobble** base pairs



Dependency graph:

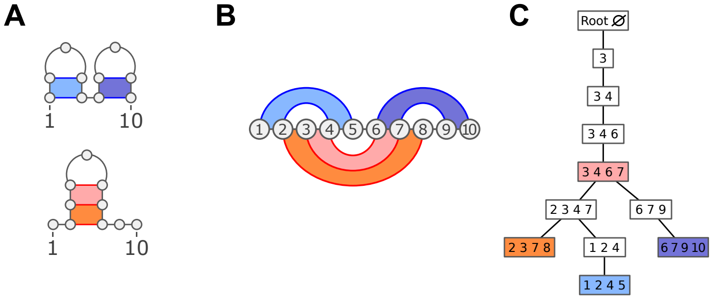
Cycles, Paths, Trees. . .



Question: How many valid **sequences**?

Answer: Non-bipartite → ∅; Bipartite → 496 672

Our problem for general free-energy models



Question: Which partition function for **valid sequences**

Problem (PFDesigns)

Input: Structures $\mathcal{R} = \{R_1, \ldots, R_k\}$ of length n + Weight $(x_1, \ldots, x_k)$

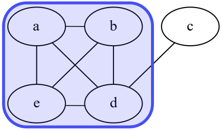
Output: Partition function

$$\mathcal{Z} = \sum_{\substack{S\in\Sigma^n \\ S \text{ valid for } \mathcal{R}}} \prod_{i=1}^{k} x_i^{E(S,R_i)}$$

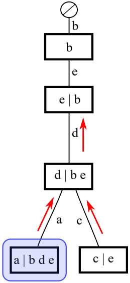
Counting/sampling, the Boltzmann-Gibbs way

a	b	c	d	e
(	.	.	)	.
.	(	(	)	)
(	(	.	)	)

Target Structures



Dependency Hypergraph



Tree Decomposition

$b = \{b_1, b_2 \ldots\}$: node of D
T_b : subtree rooted at b
w : **Width** of treedecomposition D

$$\mathcal{Z}(T_b \mid b_2 \leftarrow v_2 \ldots) = \sum_{\substack{b_1 \leftarrow v_1 \\ v_1 \in \{\mathrm{A,C,G,U}\}}} \prod_{i=1}^{k} x_i^{\sum_{E \in b} E(b, v_1, \ldots)} \prod_{c \text{ child of } b} \mathcal{Z}(T_c \mid b_1 \leftarrow v_1, \ldots)$$

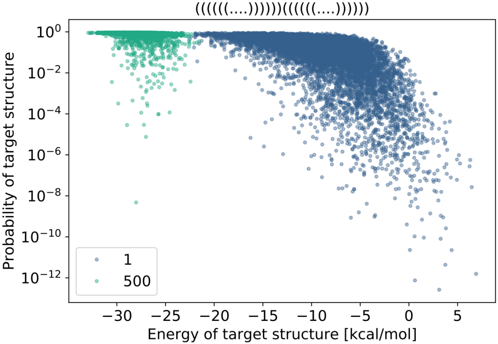
Complexity: $\Theta\left(n\,m\,k + n\,k\,2^{w+\#CC}\right)$ for sampling in Boltzmann-Gibbs distrib.

Practical impact of Boltzmann-Gibbs sampling

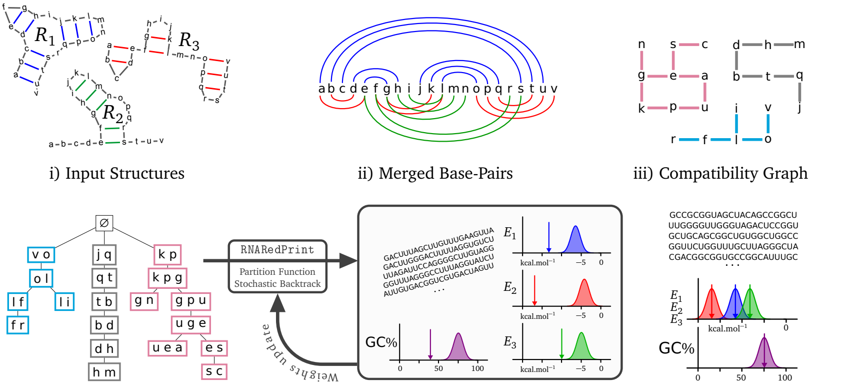
Boltzmann probability of **structure** R, pour une séquence S:

$$\mathbb{P}(R \mid S) = \frac{e^{-\frac{E(S,R)}{\beta T}}}{\mathcal{Z}_S} \quad \mathcal{Z}_S := \sum_{R'} e^{-\frac{E(S,R')}{\beta T}}$$

Objectif classique du design négatif (→ spécificité)



RNARedPrint: a flexible method for (positive) design



i) Input Structures

ii) Merged Base-Pairs

iii) Compatibility Graph

iv) Tree Decomposition

v) Weight Optimization (Adaptive Sampling)

vi) Final Designs

[Hammer/P/Wang/Will, RECOMB'18 + BMC Bioinfo 2019]

- **Fixed Parameter Tractable** algorithm based on **tree width**
- **Uniform or Boltzmann-Gibbs** sampling, to favor diversity and stability
- **Multidimensional Boltzmann sampling** for controlling free-energy, GC%. . .

`https://github.com/yannponty/RNARedPrint`

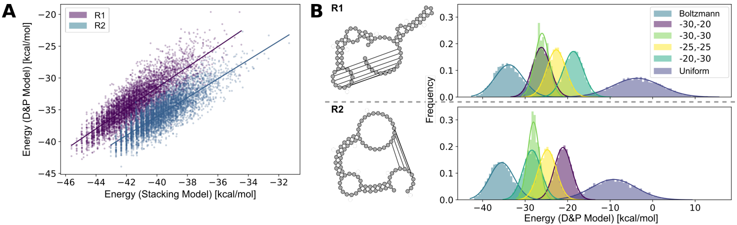
Multidimensional Boltzmann sampling

Multidimensional Boltzmann sampling [Bodini, P, DMTCS 2011]

Input: Targeted free-energies $(E^\star_\ell)_{\ell=1}^{k}$, weights $(x_\ell)_{\ell=1}^{k}$ such that $\mathbb{E}(E(w,S_\ell)) = E^\star_\ell, \forall \ell$:

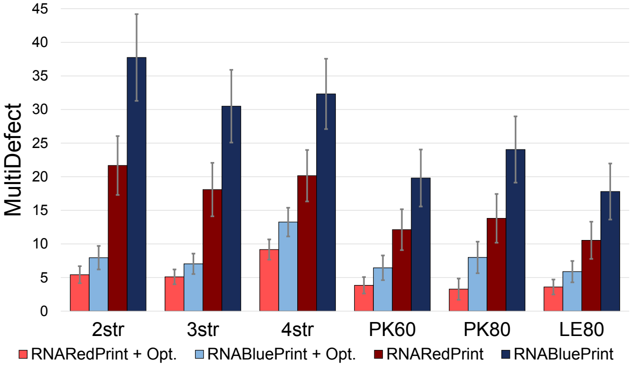
$$\mathbb{P}(w \mid x_1 \cdots x_k) \sim \prod_{\ell=1}^{k} x_\ell^{E(w,S_\ell)}$$ **+ Efficient rejection** → $\mathcal{O}(n^{k/2})$ **exact/**$\mathcal{O}(\alpha^k)$ **approx.**

Empirical efficiency for additive *concentrated* constraints (GC%, dinucleotides . . .)
→ Partial functions → Hyper-edges, *aka* cliques[1]
To Do: General framework for integer-valued constraints; Concentration tests.



[1] But tree width ↗

Strangely enough, it actually works!



$$\text{MultiDefect}(S, R_1 \cdots R_k) := \frac{\sum_{\ell=1}^{k} E(S, R_\ell) - \text{EFE}(S)}{k} + \frac{\sum\limits_{1\leq \ell < j \leq k} |E(S, R_\ell) - E(S, R_j)|}{2\binom{k}{2}}$$

where *EFE* = ensemble free-energy $EFE(S) := -\beta T \log \mathcal{Z}_S$.

Conclusion

Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)

Conclusion

Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)

Conclusion

Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)

Conclusion

Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)

Conclusion

Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)

Conclusion

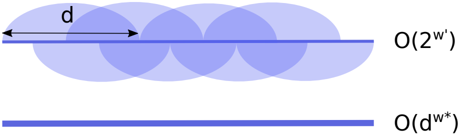
Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)

Forbidden sequences



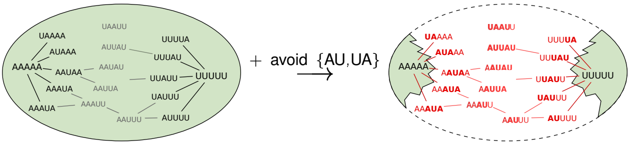
Conclusion

Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- Complexity of sequence generation for $k < 5$ structures?
- How to deal with additional sequence constraints? (DFA "product")
- How to locally navigate the space of valid sequences? (Local search)
- How to simplify dense graphs? (DCA potentials)



Conclusion

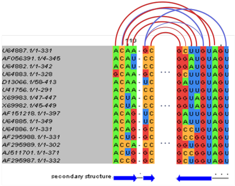
Our contribution :

- General framework for generating constrained sequences
  Ideas similar to/generalized from CTE framework (R. Dechter);
- Application to multiple RNA design, proven #P hard;
- Uses efficient rejection scheme for practical control of complex constraints;
- Practical efficiency (reasonable tree width).

Perspectives :

- To Do: Complexity of sequence generation for $k < 5$ structures?
- To Do: How to deal with additional sequence constraints? (DFA "product")
- To Do: How to locally navigate the space of valid sequences? (Local search)
- To Do: How to simplify dense graphs? (DCA potentials)

Largest vertex set given tree-width *budget*?



```
                      110
U64887.1/1-331     ACAA-GC      GCUUGUAGU
AF056391.1/4-345   ACAU-CC      GGAUGUAGU
U64882.1/1-342     ACAU-CC      GGAUGUAGU
U64883.1/1-328     GCAA-GC ...  GCUUGUAGU
D13066.1/58-413    ACAA-UC      GAUUGUAGU
U41756.1/1-291     ACAA-CC      GGUUGUAGU
X69983.1/47-447    ACUA-CC      GGUAGUAGU
X69982.1/45-449    ACUA-CC ...  GGUAGUAGU
AF151218.1/1-397   ACAG-UC      GAUUGUAGU
U64885.1/1-349     ACAG-UC      GAUUGUAGU
U64886.1/1-331     ACAG-GC      GCCUGUAGU
AF295988.1/1-331   ACUG-GC      GCCGGUAGU
AF295989.1/1-302   ACCA-CC ...  GGUGGUAGU
AJ511701.1/1-371   ACUG-GC      GCCGGUAGU
AF295987.1/1-332   ACCG-GC      GCUGGUAGU
```